université
PARIS-SACLAY

Méthodes d'analyse comparative de la variabilité intraspécifique des pangénomes procaryotes

Thèse de doctorat de l'Université Paris-Saclay

École doctorale n° 577, Structure et Dynamique des Systèmes Vivants (SDSV)
Spécialité de doctorat : Microbiologie
Graduate school : Life Science and Health
Unité de recherche : Université Paris-Saclay, Univ Évry, CNRS, CEA, Génomique métabolique, 91057, Évry-Courcouronnes, France
Référent : Université d'Évry Val d'Essonne

Thèse présentée et soutenue à Évry, le 9 février 2022, par

Adelme BAZIN

Composition du jury :

Olivier Lespinet Professeur, Université Paris-Saclay	Président
Lucie Bittner Maîtresse de conférence, Sorbonne Université	Rapportrice
Guillaume Fertin Professeur, Université de Nantes	Rapporteur
Sophie Abby Chargée de recherche, CNRS	Examinatrice
Marie Touchon Chargée de recherche, CNRS	Examinatrice
David VALLENET Directeur de recherche CEA, Genoscope CEA	Directeur de thèse
Alexandra CALTEAU Chercheure CEA, Genoscope CEA	Invitée
Claudine MÉDIGUE Directrice de recherche CNRS, Genoscope CEA	Invitée

Thèse de doctorat

NNT : 2022UPASL008

Table des matières

I Introduction **11**

1 Contexte **13**
1.1 *Bacteria*, "à quoi ça sert ?" 13
1.2 Génome et génomique 14
1.3 Objectifs de la thèse 15

II État de l'art **17**

2 Génomique microbienne **19**
2.1 Génomique, organisation et fonctions 19
2.1.1 Les génomes 19
2.1.2 Les gènes 20
2.1.3 Homologie 22
2.1.4 Organisation des génomes 23
2.1.5 Gènes et fonctions 24
2.2 Évolution des génomes 26
2.2.1 Variabilité et transfert vertical 26
2.2.1.1 Types de mutations 26
2.2.1.2 Réarrangement 27
2.2.1.3 Recombinaison 27
2.2.2 Transfert horizontal de gènes 29
2.2.2.1 Transformation naturelle 29
2.2.2.2 Conjugaison 30
2.2.2.3 Transduction 32
2.2.2.4 Autres 34
2.2.2.5 Conclusions sur le transfert horizontal 35
2.2.3 Duplication 36
2.2.4 Gènes nouveaux 37

3 Génomique comparative et pangénome **39**
3.1 Graphes en bioinformatique . . . 39
3.2 Principes et débuts de la génomique comparative . . . 42
3.2.1 Les premières comparaisons de séquences . . . 43
3.2.2 Les premières comparaisons de génomes . . . 45
3.3 Familles de gènes homologues . . . 46
3.3.1 Aspect informatique du problème . . . 46
3.3.2 Les approches bioinformatiques . . . 46
3.3.2.1 Clusters de Groupes Orthologues (ou COGs) . . . 46
3.3.2.2 CD-hit . . . 48
3.3.2.3 InParanoid . . . 49
3.3.2.4 OrthoMCL . . . 50
3.3.2.5 SEED/FigFam . . . 51
3.3.2.6 UBLAST / USEARCH / UCLUST . . . 52
3.3.2.7 kClust / MMSeqs / Linclust . . . 52
3.3.2.8 Conclusion sur la construction des familles de gènes en bioinformatique . . . 54
3.4 Des espèces chez les procaryotes ? ! . . . 54
3.4.1 Définition d'espèce . . . 54
3.4.2 Approches moléculaires pour définir une espèce . . . 55
3.4.3 Approches par comparaison de génomes . . . 56
3.4.4 Vers un consensus de l'espèce chez les procaryotes ? . . . 58
3.4.5 Homogénéisation de la taxonomie . . . 61
3.5 Pangénome . . . 62
3.5.1 Définitions et origine . . . 62
3.5.2 Modéliser les parties du pangénome . . . 64
3.5.3 Partitions du pangénome . . . 67
3.5.4 Construire un pangénome . . . 67

4 Îlots génomiques **71**
4.1 Origine et définition . . . 71
4.2 Méthodes de détection des îlots génomiques . . . 74
4.2.1 Méthodes basées sur la composition . . . 74
4.2.2 Méthodes basées sur la génomique comparative . . . 76
4.2.2.1 MOSAIC . . . 76
4.2.2.2 tRNAcc / tRIP / mobilomeFINDER . . . 77
4.2.2.3 IslandPick/IslandViewer . . . 78
4.2.2.4 xenoGI . . . 78
4.2.3 Faiblesses des approches de détections d'îlots génomiques . . . 79
4.3 Points chauds d'insertion . . . 80
4.3.1 Définition et détection . . . 80

4.3.2 Caractérisation globale des points chauds d'insertion 81

5 Modules en génomique **85**
5.1 Annotation par association 85
5.1.1 Concept 85
5.1.2 Modules et modularité 87
5.2 Identification de modules 87
5.2.1 Modules fonctionnels 88
5.2.1.1 Réseaux d'interactions protéines-protéines 88
5.2.1.2 Fusion/Fission 89
5.2.1.3 Opérons 90
5.2.1.4 Métabolisme 91
5.2.1.5 Identification de fonctions spécifiques 93
5.2.2 Modules conservés 94
5.2.2.1 Occurrences et profils phylogénétiques 94
5.2.2.2 Synténie conservée 95
5.2.2.3 Phylogénie 100
5.3 Liens entre modules fonctionnels et modules conservés 102

III Résultats **105**

6 PPanGGOLiN **107**
6.1 Graphes de pangénome et partitionnement statistique 107
6.1.1 Objectifs de PPanGGOLiN 107
6.1.2 Principe de l'approche et analyses 108
6.2 Article 1 : PPanGGOLiN : depicting microbial diversity via a partitioned pangenome graph 109
6.3 Évolution de PPanGGOLiN 137
6.3.1 Construction des familles de gènes 137
6.3.2 Identification du *persistent* dans les MAGs 138
6.4 Conclusion 139

7 panRGP **141**
7.1 Détection de régions de plasticité génomique dans un pangénome . 141
7.2 Article 2 : panRGP : a pangenome-based method to predict genomic islands and explore their diversity 142
7.3 Conclusion 151

8 panModule **153**
8.1 Détection de modules conservés dans les régions variables des pangénomes . . . 153
8.2 Article 3 : panModule : detecting conserved modules in the variable regions of a pangenome graph . . . 154
8.3 Conclusion . . . 172

9 panGBank **175**
9.1 Préambule et historique . . . 175
9.2 Introduction . . . 176
9.3 Workflow de panGBank . . . 178
9.3.1 Téléchargement des génomes . . . 178
9.3.2 Contrôle qualité . . . 178
9.3.3 Assignation taxonomique . . . 179
9.3.4 Contenu et résultats de panGBank . . . 180
9.4 Futur de panGBank . . . 182

IV Conclusions **185**

10 Conclusions et perspectives **187**
10.1 Conclusions sur ce travail de thèse . . . 187
10.2 Perspectives sur les méthodes développées . . . 189
10.2.1 PPanGGOLiN et réserves sur la méthode de partitionnement 189
10.2.2 panRGP, et l'usage d'un score arbitraire . . . 190
10.2.3 panModule, des modules conservés mais à quelle échelle ? . . . 190
10.3 Perspectives sur la génomique comparative . . . 191

Bibliographie **195**

Table des figures

2.1 Tailles des génomes (en nucléotides) de différents groupes taxonomiques d'organismes cellulaires . . . 21
2.2 Code génétique . . . 22
2.3 Illustration de différents réarrangements possibles . . . 28
2.4 Illustration de la conjugaison . . . 31

3.1 Exemple de graphe . . . 40
3.2 Chronologie de la génomique comparative . . . 44
3.3 Exemples de composantes connexes dans un graphe des COGs. . . . 48
3.4 Illustration des in-paralogues et out-paralogues . . . 50
3.5 Analyse d'ANI sur 90k génomes . . . 58
3.6 Analyse d'ANI avec 5 génomes par espèces . . . 60
3.7 Comparaison de l'homogénéité des rangs taxonomiques dans les bases de données NCBI et GTDB . . . 62
3.8 Évolution du *core* génome chez *Streptococcus agalactiae*, modélisée par une loi exponentielle . . . 64
3.9 Évolution du pangénome chez *Bacillus cereus*, modélisée par une loi de Heaps . . . 66

4.1 Schéma d'un îlot génomique . . . 72
4.2 Les différents types d'îlots génomiques . . . 73
4.3 Méthode de détection de spots . . . 82

5.1 PPI : Méthodes, types d'expériences et les espèces dans lesquelles celles-ci sont réalisées . . . 88
5.2 Organisation génomique de 4 espèces pour des gènes impliqués dans la formation d'un flagelle . . . 92

8.1 Schéma du spot *leuX* dans différentes souches de *Escherichia coli* . 172

9.1 Comparaison des génomes de GTDB avec Mash . . . 180
9.2 Nombre de pangénomes par taxons dans panGBank . . . 182

Liste des tableaux

3.1 Outils pour la construction de pangénomes 68

9.1 Plateformes web pour les pangénomes bactériens 177
9.2 Nombre de génomes restant après chaque étape de panGBank . . . 181

"What was formally recognized in physics needs now to be recognized in biology : science serves a dual function. On the one hand it is society's servant, attacking the applied problems posed by society. On the other hand, it functions as society's teacher, helping the latter to understand its world and itself. It is the latter function that is effectively missing today." Carl R. Woese, 2005

"[The relation between biology and society] is all-important ; a matter of deep concern for all biologists, philosophers of science, political leaders. And I don't see it being taken very seriously by any of them." Carl R. Woese, 2005

Première partie

Introduction

La présente partie a pour volonté d'introduire les problématiques et les sujets de la thèse sans le jargon scientifique qui facilite l'expression mais pas la compréhension par quelqu'un qui n'est pas expert en génomique. Elle a pour but de présenter brièvement les choses qui m'ont intéressé pendant ces quelques années, ainsi que leurs enjeux. Les experts me pardonneront, je l'espère, les approximations faites dans cette partie pour, je l'espère encore, la rendre accessible.

Chapitre 1

Contexte

Les génomes des bactéries sont les objets d'études de cette thèse. Toutes les affirmations ci-après sont généralement acceptées, mais pas universellement vraies, car le vivant est exceptionnel.

1.1 *Bacteria*, "à quoi ça sert ?"

Les bactéries sont des êtres microscopiques, composés d'une unique cellule et présents dans tous les environnements. On en retrouve dans la terre, dans l'eau, dans les nuages en suspension dans des gouttes d'eau. On peut également en retrouver dans les animaux, notamment dans les muqueuses ou sur la peau, ainsi que dans des environnements plus extrêmes, proches de sites actifs de volcans. On va même en retrouver dans des zones contaminées par des métaux lourds ou des hydrocarbures dans lesquels d'autres êtres vivants peinent à survivre, ou encore au cœur des fosses océaniques. Les bactéries sont partout. Elles sont remarquables par leur capacité d'adaptation qui leur permet de survivre, se transformer et prospérer lorsque leur milieu de vie change ou lorsqu'elles en colonisent de nouveaux.

Les bactéries présentent aussi une importance cruciale pour la société humaine pour deux raisons : la santé publique et l'industrie.

Au Moyen Âge, par exemple, les sociétés ont été lourdement affectées par de grandes épidémies, telles que la peste, la lèpre ou le choléra, respectivement causées par les bactéries *Yersinia pestis*, *Mycobacterium leprae*, et *Vibrio cholerae*. Certaines d'entre elles sévissent encore de nos jours dans certaines régions du monde. Néanmoins, ces épidémies ont été grandement réduites dans le monde dit "moderne" grâce à des mesures d'hygiène et aux antibiotiques, dont on attribue la découverte et la caractérisation formelle à Alexander Fleming avec la Pénicilline (Fleming, 1929). Même si la Pénicilline est en l'occurrence produite par un champignon, de nombreuses molécules antibiotiques sont produites par des bactéries.

Les actinobactéries sont à l'origine de la production de près des deux tiers des antibiotiques commercialisés ainsi que de nombreux composés anticancéreux, antiparasitaires et antifongiques (Barka et al., 2016). Au-delà des maladies qu'elles donnent et des composés qu'elles produisent, elles nous permettent aussi de vivre. En-effet, les bactéries de notre microbiote, que l'on retrouve dans notre système digestif, ont de nombreuses fonctions très bénéfiques. Elles permettent de nous protéger de certains pathogènes, de digérer certains aliments et sont impliquées dans certains mécanismes de régulations. Les bactéries sont donc importantes à étudier, car certaines causent des maladies, mais également parce que d'autres nous permettent de nous protéger.

Par ailleurs, il y a un aspect moins connu du grand public, pourtant capital, qui est l'importance des bactéries dans l'industrie de l'agroalimentaire pour la transformation des aliments à des fins de conservation ou de qualité gustative. En effet, celles-ci permettent de produire une quantité phénoménale de produits que nous consommons régulièrement comme les yaourts, le fromage ou certaines charcuteries. On peut les utiliser pour stabiliser ou réhabiliter des sols pollués, produire certaines enzymes ou certaines drogues, éliminer des parasites dans les cultures de végétaux comme le maïs ou le riz, pour ne citer que ces quelques exemples.

Elles présentent donc un intérêt capital pour notre société, puisqu'elles sont des acteurs indispensables qui permettent de faire face à de nombreuses problématiques sanitaires et écologiques. Énormément d'aspects de notre société sont liés aux bactéries, d'une manière ou d'une autre. Ainsi, pour toutes ces raisons, étudier et comprendre le fonctionnement des bactéries, comment elles évoluent et acquièrent ou perdent certaines capacités est crucial.

1.2 Génome et génomique

Un génome est l'ensemble de l'information génétique d'un être vivant. Il est fait d'ADN assemblé dans un ou plusieurs chromosomes. Le génome est la part d'un être vivant qui est transmis à ses descendants, majoritairement à l'identique. Lorsqu'on s'intéresse à un être vivant en particulier, il est intéressant d'étudier son génome, car tout ce qui le constitue y sera renseigné. Tout ce que celui-ci sera capable de faire, les composés qu'il peut produire, à quoi il ressemble ou comment celui-ci est organisé, est inscrit dans son génome. Ainsi, pour comprendre comment fonctionne une bactérie, il est particulièrement intéressant d'étudier son génome. Ce qui va faire qu'une bactérie cause une maladie, produit un antibiotique ou permet de donner un goût particulier à un fromage sera renseigné, d'une manière ou d'une autre, dans le génome de celle-ci. L'étude du génome s'appelle la génomique.

Néanmoins, un génome n'est pas écrit de manière claire et compréhensible

pour nous, humains : on va donc chercher à décoder ce qui y est inscrit. Une des approches possibles est de comparer les génomes de différents êtres vivants. Effectivement, le génome étant dépositaire de tout ce qui forme un être, on s'attend à ce que deux êtres ayant des caractéristiques communes aient des portions de génomes en commun. Par la même occasion, ce qui fait la spécificité de ces deux êtres sera renseigné dans des sections du génome qui ne leur sont pas communes. Identifier ainsi les portions communes ou uniques des génomes est la base même de ce qu'on appelle la génomique comparative.

Pour les bactéries, cela va nous intéresser particulièrement, afin d'étudier ce qui leur permet de s'adapter. Effectivement, cette capacité d'adaptation phénoménale vient du fait que celles-ci ont la capacité de récupérer de l'ADN d'autres êtres vivants, et de l'intégrer à leur génome. Ainsi, lorsqu'on étudie une population de bactéries très ressemblantes, on va pouvoir identifier les portions d'ADN qui font leurs spécificités en comparant leurs génomes et en identifiant les parties spécifiques.

Les génomes étaient à l'origine très complexe et très coûteux à étudier et les connaissances sur ceux-ci étaient très sporadiques. Ces dernières années, la génomique a connu une évolution incroyable : on peut désormais lire l'ADN de n'importe quel être vivant pour seulement quelques milliers, voir quelques centaines d'euros, par un processus qu'on appelle le séquençage. Encore plus incroyable, on est même capable de déchiffrer l'ADN de tous les êtres vivants d'un même environnement en réalisant une unique expérience de séquençage. L'étude des environnements au travers des génomes qui les composent est ce qu'on appelle la métagénomique.

Toutes ces avancées ont néanmoins soulevé d'autres problématiques : le séquençage étant désormais une chose aisée, des millions de génomes et de métagénomes ont été séquencés à travers le monde. Une source de données phénoménale, certes riche et diversifiée, mais humainement impossible à traiter. Pour analyser et comprendre cette masse de données, il faut désormais faire appel à des approches informatiques, qui permettent d'automatiser et de donner un peu de sens à ces génomes. Néanmoins, loin de ralentir ou de se stabiliser, la quantité de données génomiques générées chaque année est toujours plus importante, et il faut constamment innover et développer de nouvelles méthodes qui permettent de faire face à ce déluge, car celles-ci sont très rapidement obsolètes.

1.3 Objectifs de la thèse

Les travaux que j'ai poursuivis pendant ma thèse concernent précisément ce point. J'ai travaillé sur l'aspect informatique et algorithmique de l'étude des génomes bactériens, pour la génomique comparative sur des (dizaines de) milliers de

génomes.

Dans un premier temps, j'ai participé au développement d'un modèle de graphe de pangénome qui permet de manipuler et comparer des dizaines de milliers de génomes de bactéries entre eux et d'identifier les parties communes et les parties variables de ces bactéries. Ce modèle est utilisable au travers d'un logiciel nommé PPanGGOLiN qui sera décrit dans la première partie des résultats de cette thèse. J'ai ensuite conçu plusieurs méthodes qui utilisent le modèle de graphe de pangénome de PPanGGOLiN pour extraire des informations et apprendre de ces génomes.

Les premières méthodes ont eu pour but de détecter les portions des génomes qui variaient le plus, d'en étudier leur évolution et la manière dont elles se modifiaient. Ce sont ces régions des génomes, potentiellement liées aux capacités des bactéries, qui vont généralement intéresser les chercheurs ou les industriels. Ces méthodes ont été regroupées sous le nom de panRGP, et seront détaillées dans la seconde partie des résultats de cette thèse.

Finalement, j'ai utilisé la grande masse de génomes à laquelle nous commençons à avoir accès pour tenter de prédire les portions de génomes qui fonctionnent ensemble dans un même processus. Effectivement, comme les génomes des bactéries varient énormément et évoluent très vite, si certaines portions de leurs génomes sont souvent retrouvées conjointement, une hypothèse que l'on peut émettre est que ces portions sont mutuellement nécessaires pour apporter une fonction. Une méthode, nommée panModule, pour identifier ces portions dans les génomes sera détaillée en troisième partie des résultats cette thèse.

Deuxième partie

État de l'art

Cette partie de la thèse permet d'introduire les différents concepts utilisés. Elle introduit brievement la génomique microbienne, en décrivant les concepts clés de la génomique, tant sur l'aspect biologique que sur l'aspect informatique, puis présente les différentes mécanismes qui permettent aux génomes d'évoluer. Ensuite, trois chapitre plus spécifiques viennent décrire trois aspects clés : la génomique comparative et la pangénomique, les îlots génomiques, et les modules en génomique

Chapitre 2

Génomique microbienne

Le but de ce chapitre est de donner aux lecteurs les outils pour comprendre le contexte scientifique et les implications des travaux que j'ai pu réaliser autour des génomes bactériens de manière assez générale. Il introduit un outil clé de cette thèse, la théorie des graphes, puisqu'une grande partie des problèmes que l'on rencontre en génomique peuvent être résolus avec ces approches. Il inclut aussi une introduction sur les manières dont les génomes des procaryotes évoluent et dont la compréhension est centrale à la bioinformatique moléculaire.

Lecteur, bon courage.

2.1 Génomique, organisation et fonctions

2.1.1 Les génomes

La génomique microbienne est une science qui consiste à analyser des génomes de microbes avec des outils informatiques. Parmi ceux-ci, je ne parlerais que des bactéries, car la majorité des données de génomes sont des génomes de bactéries, et car je n'ai en très grande majorité que travaillé sur des bactéries. C'est un des nombreux domaines de la bioinformatique, ses pratiquants se considèrent généralement comme des bioinformaticiens, ou parfois comme des biologistes avec une appétence pour la bioinformatique.

Les génomes des bactéries ont plusieurs caractéristiques communes. Il est souvent circulaire et constitué de quelques centaines de milliers de paires de bases, pour les plus petits, à quelques millions pour les plus grands. Les tailles exprimées en génomique seront indiquées comme lorsqu'on compte les octets en informatique : 1 kb représente mille paires de bases, 1 Mb représente 1 million de paires de bases, 1 Gb représente un milliard de paires de bases. Les bactéries avec les génomes les plus petits sont souvent des bactéries parasites intracellulaires obliga-

toires, comme *Chlamydia trachomatis*. Son génome fait 1 Mb dans un seul chromosome, et ne dispose pas de certaines capacités métaboliques fondamentales. Elle n'a pas la capacité de survivre et de se développer autrement qu'en infectant des cellules qui vont lui permettre de récupérer ces métabolites. Les génomes les plus grands sont souvent, mais pas toujours, associés à des bactéries environnementales, comme les bactéries du sol. Un exemple est *Burkholderia cepacia* qui est un organisme trouvable dans l'environnement, mais aussi impliqué dans des infections notamment chez les patients atteints de mucoviscidose. Cette bactérie possède un génome d'environ 8 Mb, réparti dans 3 chromosomes linéaires différents.

Pour comparaison, la figure 2.1 illustre la taille des génomes de différents organismes sur une échelle logarithmique, en indiquant quelques espèces connues comme *Escherichia coli* ou *Homo sapiens* (l'humain). On notera dans cette figure que les bactéries sont, de loin, les organismes ayant les génomes les plus petits, avec l'exception notable des virus qui ne sont pas représentés et qui ont des génomes généralement encore plus petits, mais majoritairement incapable de se reproduire et de se développer seuls.

2.1.2 Les gènes

Un génome en soi ne permet rien. Il contient l'information dont un organisme a besoin pour fonctionner, mais il n'est que le récipient de cette information.

Certaines parties du génome vont être retranscrites en des petites molécules d'Acides Ribonucléiques (ARN) lors d'une étape qu'on appelle la transcription.

Certaines de ces molécules ont une fonction propre, notamment celles qui sont palindromiques et vont pouvoir former des structures grâce à cette propriété. Tel est le cas des ARN de transfert (dit tRNA, ou ARNt) qui sont essentiels pour une bactérie, car elles participent à la synthèse des protéines. D'autres molécules ARN, les ARN messagers, vont être lues par la machinerie de la bactérie pour faire des protéines dans une étape qu'on appelle la traduction.

Les protéines sont les molécules qui réalisent la majorité des fonctions dans une bactérie. Elles sont constituées d'acides aminés, pour lesquels on dénombre une centaine de structures différentes, mais dont seuls 20 d'entre eux, dits standards, sont d'ordinaire considérés en génomique microbienne.

Le passage d'une molécule d'ARN à une molécule d'acides aminés ne se fait pas aléatoirement, il suit un ordre précis qui est défini par ce qu'on appelle le code génétique. Le code génétique n'est pas universel puisqu'il en existe 33 différents, recensés au 23 juillet 2021, et potentiellement des dizaines, voir des centaines d'autres. Un récent article rend compte de plusieurs usages de codons alternatifs non référencés et présents dans des génomes de bases de données publiques (Shulgina et al., 2021).

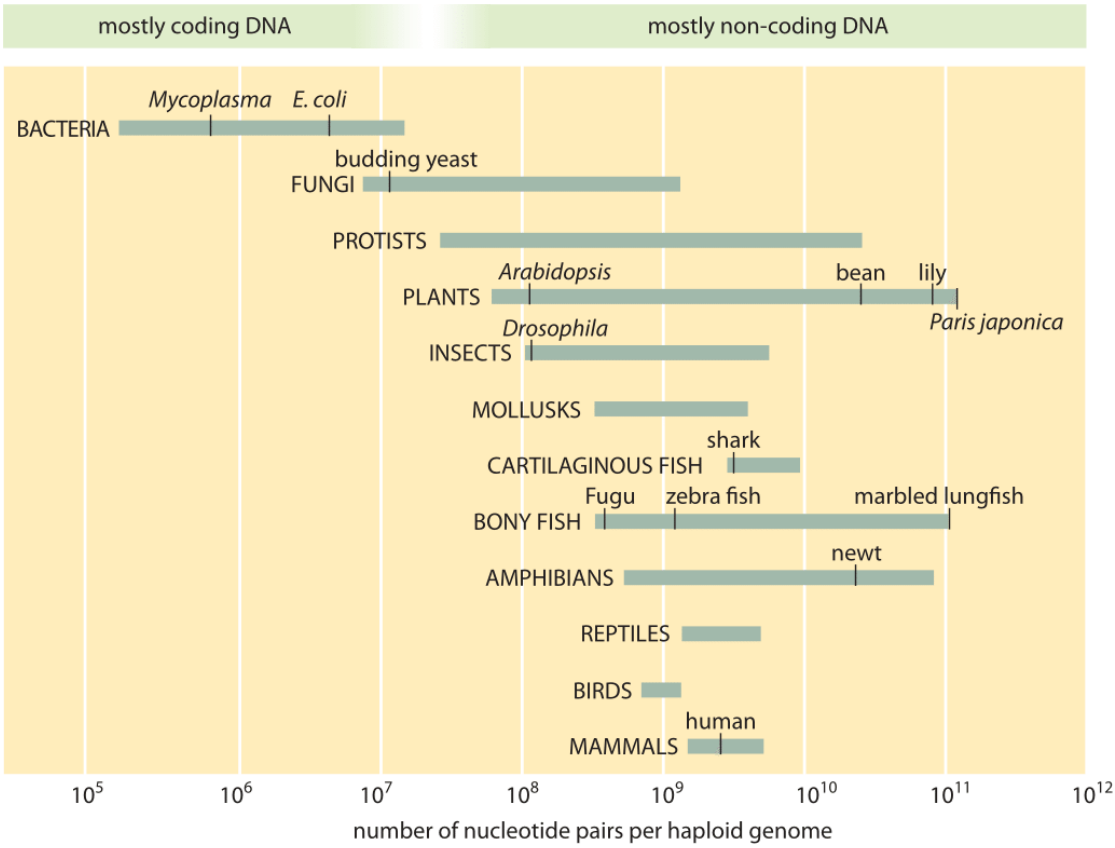


FIGURE 2.1 – Tailles des génomes (en nucléotides) de différents groupes taxonomiques d'organismes cellulaires
Cette figure représente la taille des génomes (en nucléotides) de différents groupes taxonomiques en utilisant une échelle logarithmique, avec quelques espèces connues indiquées comme références. Copié de MILO et al., 2015.

Le "code génétique standard", très tristement nommé, correspond à celui de certains eucaryotes (avec **beaucoup** d'exceptions), notamment les vertébrés. Les bactéries, Archées et plantes utilisent un code commun légèrement différent au niveau du codon d'initiation des protéines (avec, encore une fois, des exceptions).

Dans ledit code génétique, une suite de 3 bases, appelée codon, va donner un acide aminé. On génère ainsi un code avec 64 combinaisons de 3 bases, certaines étant redondantes, différents codons pouvant donner le même acide aminé. Ces combinaisons permettent 21 résultats possibles : les 20 acides aminés standards, et un codon spécial qu'on appelle "codon STOP" qui indique la fin de la protéine à la machinerie cellulaire de la bactérie, et ainsi la fin de la traduction. Pour illustrer, la figure 2.2 montre le code génétique qui est utilisé pour traduire la séquence d'un gène à l'exception du premier codon, le codon d'initiation.

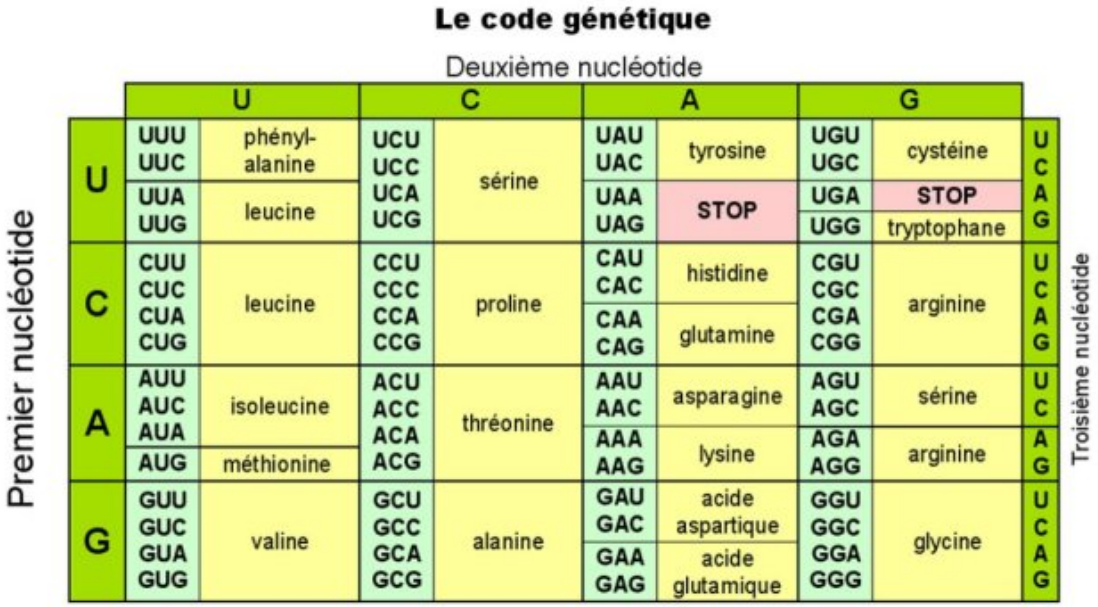


FIGURE 2.2 – Code génétique
Cette figure représente le code génétique utilisé pour faire correspondre une séquence d'ADN à une séquence d'acides aminés. Copié de https://svtfeyder.wordpress.com/le-code-genetique/.

La portion de génome qui correspond à une protéine est ce qu'on appelle un gène. Les portions de génome qui correspondent à certains ARN fonctionnels comme les ARNt sont aussi appelés gènes. Néanmoins, par abus de langage et comme les gènes codant les protéines sont des éléments importants dans ce document, on utilisera, par la suite, le terme "gène" pour désigner un gène codant des protéines. Il y a débat dans certaines sphères sur la définition de "gène", je n'ai aucune volonté de rentrer ou de répondre à celui-ci dans ce document.

Les gènes recouvrent l'immense majorité des génomes des bactéries, contrairement aux génomes eucaryotes. Une protéine bactérienne peut être constituée de quelques acides aminés à plusieurs milliers, mais en moyenne ce nombre oscille autour de 300, ce qui représente une taille approximative de 1 kb pour le gène correspondant.

2.1.3 Homologie

Le génome, et notamment celui des bactéries, évolue à chaque génération. Il change continuellement, ce qui donne une certaine diversité même aux bactéries qui

sont génétiquement proches. Les portions de génomes qui ont une origine ancestrale commune, c'est-à-dire issue de la même portion de génome originelle, sont dites homologues. On peut parler d'homologie pour n'importe quel élément du génome, et notamment des gènes. Deux gènes sont homologues s'ils ont un gène ancestral en commun.

Les évènements évolutifs qui séparent deux gènes homologues peuvent être divers et permettent de préciser le type d'homologie dont il est question. Deux gènes séparés par un évènement de duplication sont dits paralogues. Entre autres, on parlera de paralogie lorsqu'une portion de génome est dupliquée et que les deux portions résultantes évoluent par la suite de manière indépendante à l'intérieur du même génome.

Par ailleurs, deux gènes séparés exclusivement par des évènements de spéciation sont dits orthologues. Tel est le cas de deux gènes présents dans deux génomes différents, mais issus d'un même gène originel non dupliqué.

Ces notions sont importantes, car elles sont le fondement de la génomique comparative et permettent d'émettre plusieurs d'hypothèses. Premièrement, on va supposer que les gènes orthologues ont généralement les mêmes fonctions (ou en tout cas, des fonctions très proches) et sont donc impliqués dans les mêmes processus. Les gènes paralogues peuvent avoir des fonctions légèrement différentes : par exemple, des enzymes ayant des substrats différents, mais catalysant le même type de transformation chimique. Les gènes qui ont une origine commune sont similaires dans la séquence de nucléotides qui les compose. Si les séquences de nucléotides (et par extension, les séquences d'acides aminés) de deux gènes sont similaires et présents dans deux génomes différents, on va alors supposer que ces deux gènes sont homologues. Néanmoins, la discrimination entre orthologie et paralogie ne peut pas se faire sur la seule base de la similarité.

Il est important de noter que deux gènes associés à une même fonction ne sont pas nécessairement homologues. Il existe des cas de convergence évolutive, c'est-à-dire que des fonctions similaires se sont développées, mais sans avoir d'origine commune. De même, certaines régions de génomes peuvent être similaires sans être homologues : c'est possible, par hasard, pour des séquences courtes, et également pour des régions ayant une faible complexité dans certaines protéines.

2.1.4 Organisation des génomes

La majorité du génome d'une bactérie est recouvert de gènes, mais ceux-ci ne sont pas disposés aléatoirement le long d'un génome.

Dans un premier temps, certains gènes peuvent se regrouper dans ce qu'on appelle un opéron. Un opéron est un bloc d'ADN qui est transcrit en un unique ARNm et qui contient plusieurs gènes. Ces gènes sont orientés de la même manière et sont généralement impliqués dans un même processus.

De manière équivalente, même si entre les génomes de différentes espèces on retrouve des ordres différents de gènes et d'opérons, un gène va généralement être retrouvé dans un contexte similaire (Lathe III et al., 2000), c'est-à-dire que ses voisins seront souvent impliqués dans les mêmes processus. La conservation de l'ordre des gènes dans les génomes est appelée synténie conservée.

Les gènes dans un génome participent à des fonctions très diverses. Certains sont nécessaires à la survie de la bactérie, comme les gènes qui codent des protéines impliquées dans la synthèse de nucléotides. D'autres ont une utilité contextuelle, qui n'a pas forcément d'intérêt dans tous les milieux de vie, comme les protéines utilisées pour dégrader un composé chimique particulier. Ainsi, certains gènes sont retrouvés dans la majorité des bactéries, et d'autres seront beaucoup plus rares. On vérifie cela au travers de l'organisation des gènes le long du chromosome bactérien. Les gènes dits "essentiels", qui remplissent donc des fonctions nécessaires, sont retrouvés beaucoup plus souvent sur le brin direct que sur le brin indirect du chromosome (Rocha et al., 2003).

Un autre niveau d'organisation se fait au niveau de l'expressivité des gènes. Effectivement, la bactérie ne produit pas toutes les protéines de son génome en même temps, ni en même quantité. Plusieurs mécanismes, appelés mécanismes de régulation, vont rentrer en jeu pour produire les protéines dont la bactérie a besoin uniquement lorsqu'elles sont nécessaires. Cette nécessité se reflète notamment dans l'organisation du génome de la bactérie : chez les bactéries avec un taux de division élevé (celles qui se reproduisent vite), les gènes qui ont le plus besoin d'être exprimés sont retrouvés à proximité de l'origine de réplication et sont plus éloignés du terminus de réplication (Sharp et al., 1989 ; Vieira-Silva et al., 2010). Cela permet d'augmenter l'expressivité de ces gènes : chez ces mêmes bactéries, il peut y avoir plusieurs cycles de réplication simultanément (Fujisawa et al., 1973). Ainsi, les gènes proches de l'origine de réplication peuvent être présents simultanément en plusieurs copies dans une cellule, et ainsi être beaucoup plus exprimés que les gènes proches de la terminaison.

Ces différentes observations sont des tendances globales qui ont été mesurées sur les génomes bactériens, et non des règles strictes ; il y a donc bien sûr souvent des exceptions.

2.1.5 Gènes et fonctions

Certaines protéines possèdent une seule fonction et vont donc globalement évoluer seules, sans autre contrainte que la nécessité de maintenir cette fonction, si celle-ci est essentielle. Cependant, d'autres protéines vont avoir besoin d'interagir entre elles pour réaliser une fonction. C'est le cas, par exemple, de toutes les protéines qui vont former des complexes, des transporteurs ou des systèmes de sécrétions, qui sont des éléments clés permettant l'échange de certains composés entre

une cellule et son environnement mais également à l'intérieur de celle-ci. De telles protéines vont donc évoluer conjointement car le maintien de la fonction nécessite le maintien de leurs interactions : on parle alors de processus de coévolution.

Dans d'autres cas, notamment celui de la biosynthèse, l'implication de plusieurs protéines est nécessaire pour assurer la fonction. La biosynthèse correspond à la formation de substances par un organisme dans son milieu interne, généralement en modifiant un substrat initial à l'aide de protéines particulières, appelées enzymes. Chez les bactéries, plusieurs voies de biosynthèse sont essentielles, comme celles des nucléotides ou des acides aminés, car sans eux il n'y a ni acide nucléique ni protéine !

Dans le cas où plusieurs protéines sont nécessaires pour réaliser une fonction, les gènes doivent être transcrits en même temps, dans des quantités équivalentes, pour que les transcrits soit ensuite traduits simultanément afin que toutes les protéines requises soient disponibles. De fait, on va souvent retrouver ces gènes à proximité les uns des autres sur le génome, souvent dans un ou plusieurs opérons, mais il y a des exceptions.

Un certain nombre de protéines peuvent aussi intervenir dans plusieurs fonctions ou plusieurs processus : on parle alors de protéines multifonctionnelles. Certaines enzymes peuvent notamment intervenir dans plusieurs voies de biosynthèses d'acides aminés, d'autres sont capables d'utiliser plusieurs substrats, avec plus ou moins d'affinité. De plus, certaines enzymes ont plusieurs domaines fonctionnels et peuvent réaliser plusieurs réactions dans une même voie métabolique.

2.2 Évolution des génomes

Les mécanismes d'évolution des génomes sont multiples et complexes. Dans cette section, je vais tenter d'en décrire les principaux concernant les bactéries. Je ne détaillerai pas les mécanismes moléculaires, mais je donnerai une description générale de leur fonctionnement et de leur usage particulier chez certaines bactéries, le tout illustré d'exemples. Tous ces mécanismes seront décrits indépendamment bien qu'ils soient tous acteurs de l'évolution du génome au cours du temps et des générations.

2.2.1 Variabilité et transfert vertical

2.2.1.1 Types de mutations

L'héritage vertical est le fait de transmettre son génome a sa descendance. La transmission n'est néanmoins pas parfaite, et il peut y avoir des différences d'une génération à l'autre. On divise souvent les modifications en deux catégories : les SNPs et les indels.

Un SNP (Single Nucleotid Polymorphism) correspond au changement d'un nucléotide en un autre nucléotide. Cela va généralement n'avoir aucun impact sur une séquence protéique, dans ce cas-là, c'est une mutation silencieuse, aussi appelée mutation synonyme. Si la modification par contre change un acide aminé, c'est une mutation non-synonyme. La modification peut être délétère si elle est située à un endroit clé, comme le codon STOP ou le site actif, ou si un codon STOP est créé au milieu de la séquence du gène. Dans ce cas-là on parlera de mutation délétère généralement ou non-sens.

Les indels sont des variations définies par l'insertion ou la délétion d'un ou plusieurs nucléotides dans la séquence du génome. On les regroupe ensemble sous l'appellation "indel" car il est impossible de différencier les insertions des délétions sans une analyse phylogénétique. Dans ce cas, il faut déterminer l'état et la composition de la séquence ancestrale avant l'évènement.

Lorsqu'un indel se produit dans la séquence d'un gène codant une protéine, il y a deux résultats possibles.

Le premier survient lorsqu'un multiple de trois nucléotides est inséré ou délété. Dans ce cas-là, cela va rajouter ou supprimer un ou plusieurs acides aminés de la protéine. Cependant, si celui-ci n'était pas essentiel et si cela n'affecte pas l'intégrité de la protéine, il n'y aura pas forcément beaucoup de changements au niveau phénotypique.

Le second survient lorsque l'insertion ou la délétion ne concerne pas un multiple de trois nucléotides. Dans ce cas, cela va créer un changement de cadre de lecture, qui va radicalement affecter la composition en acides aminés de la protéine à

partir de l'endroit de la mutation. Ce décalage fragmente donc le gène en plusieurs morceaux, le rendant le plus souvent non fonctionnel ; ce phénomène est appelé pseudogénisation. Si la protéine est essentielle, cette mutation sera délétère pour l'organisme.

2.2.1.2 Réarrangement

La variabilité des génomes peut être également induite par un évènement de réarrangement. Un réarrangement implique l'échange de positions de certaines portions du génome entre elles. Même si un tel évènement peut sembler anecdotique au premier abord (après tout, le contenu reste le même), cela peut avoir un impact fonctionnel important sur les gènes. On a vu, dans la section 2.1.4, l'importance de l'organisation et de l'ordre des gènes dans un génome : un réarrangement va donc changer cet ordre, affectant éventuellement l'expressivité des gènes situés dans les régions où a eu lieu le réarrangement.

Il s'agit d'un processus répandu chez les procaryotes (Eisen et al., 2000) et la plupart présentent des réarrangements structuraux importants à l'échelle des espèces, observés à la fois dans des souches de laboratoire, mais aussi dans la nature (A. E. Darling et al., 2008).

Différents scénarios possibles de réarrangement du chromosome bactérien sont illustrés dans la figure 2.3.

Les réarrangements possibles ne sont pas tous équiprobables (A. E. Darling et al., 2008), car certains peuvent causer plus de problèmes structuraux que d'autres.

Les réarrangements symétriques (figure 2.3a) sont potentiellement les plus communs, car les réarrangements non symétriques (figure 2.3b) provoquent un déséquilibre dans la taille des réplicons du chromosome bactérien et sont vraisemblablement contre-sélectionnés. D'autre part, les réarrangements au sein d'un réplicon (figure 2.3c) sont généralement plus petits que les réarrangements entre réplicons, et sont aussi sujets à des réarrangements inverses qui vont restaurer l'état initial du génome.

De plus, Darling et al. ont observé que les réarrangements proches de l'origine de réplication sont plus fréquents que ceux proches du site de terminaison (A. E. Darling et al., 2008).

Au travers de ce mécanisme de réarrangement, l'ordre des gènes et leur position dans les génomes va donc varier au cours du temps, ce qui va affecter leur expressivité, mais également leur contexte fonctionnel.

2.2.1.3 Recombinaison

La recombinaison est un mécanisme très important chez tous les êtres vivants : chez les bactéries, elle intervient notamment dans les mécanismes de réparation de

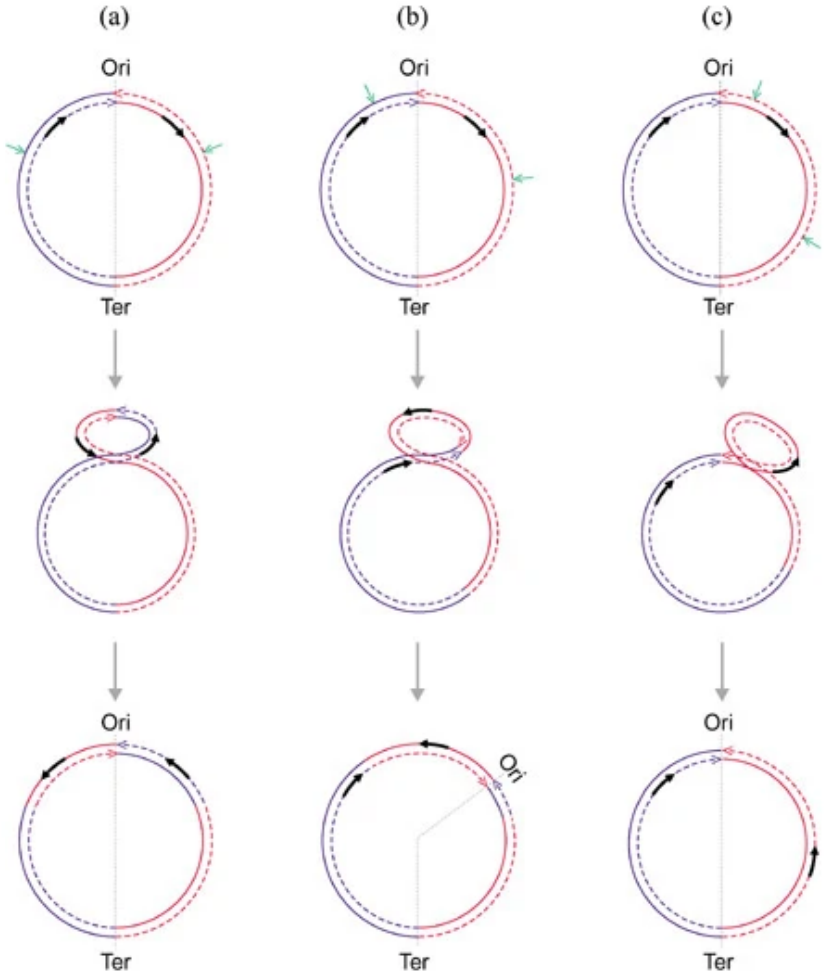


FIGURE 2.3 – Illustration de différents réarrangements possibles
Figure permettant d'illustrer différents réarrangements possibles du chromosome bactérien. (a) réarrangement symétrique, (b) réarrangement non symétrique, (c) réarrangement au sein d'un réplicon issu de MACKIEWICZ et al., 2001.

l'ADN, les événements de réarrangement et dans le transfert horizontal de gènes (EISENSTARK, 1977).

Le processus de recombinaison implique l'échange de deux portions d'ADN entre deux brins d'ADN. On définit deux types de recombinaison, à savoir la recombinaison homologue et la recombinaison non homologue. La recombinaison homologue se produit entre deux portions d'ADN homologues et aura pour effet d'échanger les deux portions entre les brins. Elle est généralement médié par une protéine très étudiée, nommée RecA, dont l'absence mène à l'incapacité de

réaliser plusieurs mécanismes et à une perte de valeur adaptative (fitness) plutôt importante pour les bactéries (Eisenstark, 1977).

À l'inverse, on appelle recombinaison non-homologue l'intégration d'éléments génomiques dans une région génomique donnée. Les mécanismes moléculaires, ici, sont très dépendants de l'origine de ces éléments génomiques. Les mécanismes moléculaires entre les deux types de recombinaison n'ont rien à voir, mais l'effet final est assez proche : le contenu du génome change, soit par remplacement, soit par ajout.

2.2.2 Transfert horizontal de gènes

Le transfert horizontal de gènes est un terme générique pour désigner l'acquisition d'ADN par des voies qui ne sont pas celles de l'héritage vertical. On parle aussi de transfert latéral, dans certains articles de la bibliographie scientifique.

De multiples mécanismes de transfert horizontal ont été décrits chez les procaryotes. Nous allons les décrire dans les sections suivantes en commençant par ceux considérés comme les trois principaux. Il existe plus de trois mécanismes, mais les autres sont *a priori* peu répandus ou, en tout cas, n'ont pas été largement observés dans l'arbre des procaryotes.

2.2.2.1 Transformation naturelle

La transformation naturelle est le premier mécanisme de transfert horizontal à avoir été identifié (Griffith, 1928). Il implique la récupération, par la cellule, d'ADN présent dans l'environnement (Dubnau et al., 2019). Toutes les bactéries ne sont pas capables de réaliser la transformation, mais la compétence est assez largement répandue dans l'arbre des procaryotes. Ce mécanisme existe en partie car il permet aussi à la cellule de récupérer de l'ADN, sans avoir à le synthétiser, pour soit le réutiliser, ce qui semble être avantageux en matière d'énergie, soit le dégrader pour en récupérer des composants comme les nucléotides.

Le processus nécessite une fixation de l'ADN à l'enveloppe de la cellule, avant que celui-ci soit intégré dans le cytoplasme, puis partiellement ou totalement intégré au génome.

Les mécanismes moléculaires qui régissent l'intégration dans la cellule, la conservation de l'intégrité de la molécule d'ADN et l'intégration dans le génome, ainsi que la proportion dans laquelle la bactérie va réaliser cette transformation varient grandement d'une espèce à l'autre (Stewart et al., 1986).

Au sujet de l'intégration dans la cellule, il existe un mécanisme différent entre les bactéries didermes et monodermes, car cela implique, dans le cas des didermes de traverser la paroi en double couche de la cellule. Néanmoins, les deux mécanismes sont moléculairement bien décrits (Dubnau et al., 2019).

Certaines bactéries récupèrent de l'ADN indépendamment de sa composition ou de son origine. D'autres, comme *Nesseria gonorrhoeae*, nécessitent la présence d'une séquence d'ADN spécifique sur la molécule pour que celle-ci soit intégrée. Ainsi, l'ADN acquis par transformation correspond, dans ce cas précis, à des molécules issues de bactéries de la même espèce ou d'espèces proches.

La transformation naturelle peut être utilisée dans le contexte de la réparation de l'ADN. Notamment, *Streptococcus pneumoniae*, qui ne dispose pas de système de réparation SOS, tue les bactéries sœurs de la même espèce pour en récupérer l'ADN. De plus, la bactérie méthyle activement l'ADN simple brin récupéré afin de mitiger sa dégradation. Contrairement à *N. gonorrhoeae*, elle est aussi capable de récupérer de l'ADN qui n'est pas issu d'une bactérie sœur.

Dans certains cas, la transformation naturelle va être favorisée entre des bactéries de même espèce mais de souches différentes et défavorisée entre des bactéries extrêmement proches (ex. de la même souche) (Lyons et al., 2016). Dans le cas de *Bacillus subtilis*, les bactéries de souches différentes vont s'entre-tuer en produisant différents antibiotiques, ce qui va déclencher des mécanismes de réponses au stress ou bien de mort cellulaire. En réponse à ces mécanismes de réponses au stress, les bactéries acquièrent une quantité plus importante d'ADN étranger lorsqu'elles sont dans des zones peuplées de bactéries de souches différentes que lorsqu'il n'y a que des bactéries de la même souche. La reconnaissance entre les souches semble être effectuée par des gènes de fonctions parfois très différentes, dont la transcription varie énormément, et qui sont peu conservés entre les différentes souches de *B. subtilis*.

La transformation naturelle est donc un mécanisme très présent dans le monde procaryote, et qui intervient dans plusieurs systèmes en fonction du mode de vie et des capacités des bactéries.

2.2.2.2 Conjugaison

La conjugaison est un mode de transfert horizontal qui utilise un système moléculaire appelé système conjugatif. Il s'agit d'un système unidirectionnel qui nécessite un contact direct entre deux bactéries, une bactérie donneuse et une bactérie receveuse. La bactérie donneuse est celle qui doit posséder le système conjugatif. Il y a deux catégories d'éléments génomiques qui disposent d'un système conjugatif : les plasmides conjugatifs, qui se répliquent indépendamment du génome et les ICE (Integrative Conjugative Elements) (Johnson et al., 2015) qui sont des éléments intégratifs conjugatifs, aussi appelés parfois des transposons conjugatifs. Ces derniers sont des éléments intégrés directement au génome, donc répliqués avec celui-ci, et qui disposent d'un système conjugatif, généralement un système de sécrétion de type IV.

Ces éléments génomiques mesurent entre 18 et 500 kb, voire plus, et sont sou-

vent des éléments clés, car porteurs de fonctions d'adaptation qui viennent s'ajouter aux gènes du système conjugatif. Ces fonctions peuvent correspondre, entre autres, à la résistance aux antibiotiques ou aux métaux lourds.

La frontière entre les deux types d'éléments (plasmides et ICE) est assez fluide, car il existe des plasmides, portant un système conjugatif, qui peuvent s'intégrer dans le génome, ainsi que des ICE capables d'exister hors du génome, sous forme d'épisomes.

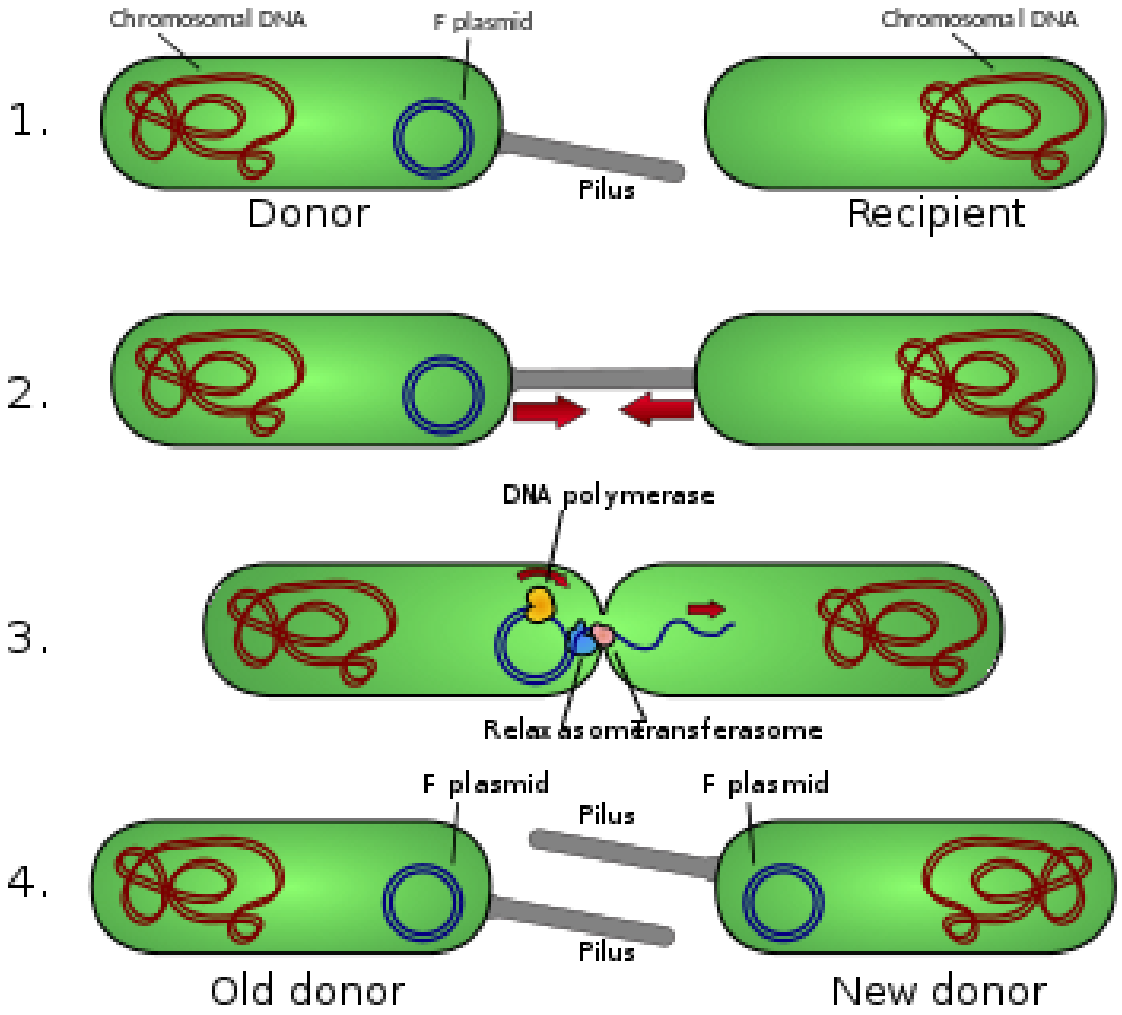


FIGURE 2.4 – Illustration de la conjugaison
Figure permettant d'illustrer le mécanisme de conjugaison dans le cas d'un plasmide conjugatif. Image venant de https://en.wikipedia.org/wiki/Bacterial_conjugation consulté le 2 août 2021.

Pendant la conjugaison, la bactérie donneuse va produire un appendice appelé

pilus conjugatif, fixé sur sa membrane (voir figure 2.4.1). Lorsqu'elle va se connecter à une autre bactérie, elle va initier le transfert d'une copie de l'élément conjugatif, en commençant par l'origine de transfert (voir figures 2.4.2 et 2.4.3). Si l'élément transféré est un ICE, celui-ci peut s'exciser du génome avant de se répliquer, puis se réintégrer au génome une fois le transfert réalisé.

Après le transfert, la bactérie receveuse sera, elle aussi, capable de produire un pilus conjugatif afin de réaliser des conjugaisons (voir figure 2.4.4).

Dans le cas des ICE, il peut parfois y avoir une excision imprécise et des gènes flanquant l'ICE seront alors transmis avec l'élément. Cela arrive notamment lorsque plusieurs séquences ressemblant au site d'intégration sont adjacentes dans le génome (Gibbons et al., 2011).

Le mécanisme de conjugaison peut être aussi utilisé pour transférer des éléments dits mobilisables, mais qui n'ont pas de système conjugatif. En effet, certains plasmides, ou d'autres éléments mobiles des génomes, peuvent être mobilisés et transmis via des pili (Valentine et al., 1988). Dans ce cas, la bactérie receveuse ne reçoit pas le système conjugatif mais uniquement l'élément mobilisable en question. On parle en l'occurrence d'IME (Integrative Mobilizable Elements) lorsque ceux-ci peuvent s'intégrer aux génomes.

Les systèmes conjugatifs sont donc des mécanismes majeurs de transmission horizontale des caractéristiques d'une bactérie à d'autres bactéries. Ce gain de fonction leur permet de s'adapter à des milieux nouveaux ou à des changements dans leur milieu de vie d'origine.

2.2.2.3 Transduction

2.2.2.3.a Les phages

La transduction est un mécanisme qui fonctionne via les phages. Les phages sont des virus de bactéries considérés comme les entités biologiques les plus abondantes sur Terre (Cobián Güemes et al., 2016). Comme la plupart des virus, leur mode de fonctionnement se repose sur l'infection d'une cellule, dont ils vont détourner la machinerie cellulaire pour se répliquer, former des milliers de copies d'eux-mêmes, avant de se relâcher dans l'environnement en tuant la cellule.

Les génomes des phages sont généralement petits, de l'ordre de quelques milliers de bases, et contiennent quelques dizaines de gènes, qui vont servir à produire une tête protéique appelée capside pour protéger le matériel génétique, généralement une queue pour infecter les cellules, et éventuellement un système de réplication.

On classifie les phages en deux larges catégories : les phages virulents et les phages tempérés.

Les phages virulents suivent un cycle lytique : ils infectent leurs hôtes, détournent la machinerie de réplication, répliquent des milliers de copies d'eux-

mêmes, puis se relâchent dans la nature en lysant la cellule (*i.e.* en brisant la paroi).

Quant aux phages tempérés, ils peuvent soit suivre le même chemin que les phages virulents, soit établir une relation un peu plus stable avec leur hôte, de manière temporaire (à l'échelle de l'évolution), qu'on appelle cycle lysogénique. Ces phages vont alors s'intégrer au génome ou rester dans la cellule à l'état d'épisome. Si le phage s'intègre au génome, il se répliquera avec le chromosome alors que s'il est sous forme d'épisome, il se répliquera de manière indépendante comme les plasmides.

Les phages intégrés aux génomes sont typiquement appelés prophages. Une fois intégrés au génome, ceux-ci vont évoluer au cours du temps au même titre que le reste du génome et peuvent ainsi devenir inopérants si certains gènes clés sont pseudogénisés. De fait, on peut parfois retrouver des restes de phages rendus inopérants, par le hasard de l'évolution, dans les génomes de bactéries. Il existe aussi des cas où des phages tempérés se sont intégrés au génome d'une bactérie et leurs gènes se sont adaptés pour former une machinerie utilisée par la bactérie elle-même : c'est le cas des systèmes de sécrétion de type VI (T6SS) qui sont homologues à la queue de phages (Veesler et al., 2011).

Par ailleurs, le prophage qui est resté fonctionnel peut entrer en cycle lytique décrit précédemment.

Le choix entre cycle lytique et cycle lysogénique pour certains phages tempérés (le groupe SPbeta) peut être régulé par des protéines produites par les phages et qui seront relâchées dans l'environnement au moment de la lyse cellulaire (Erez et al., 2017). Si la quantité de protéines est assez importante dans l'environnement, les phages auront tendance à amorcer un cycle lysogénique. L'idée derrière est que, vraisemblablement, ce mécanisme a été sélectionné pour garantir, d'une part, la présence de cellules à infecter et, d'autre part, la survie des phages sur le long terme. Effectivement, si ceux-ci infectent et tuent toutes les cellules d'un environnement, ils ne pourront plus se répliquer et les phages eux-mêmes vont finir par se dégrader. Pour préserver une source de reproduction, ces phages s'autorégulent donc entre cycle lytique et cycle lysogénique.

Les bactéries, ayant tout intérêt à ne pas se faire infecter, ont plusieurs mécanismes de défense contre les phages (Dy et al., 2014), notamment les systèmes CRISPR-Cas qui sont maintenant très connus au travers des méthodes développées pour couper l'ADN d'un organisme (Jinek et al., 2012). Il est souvent question dans la littérature d'une "course à l'armement", par analogie à certains évènements historiques récents (extrêmement, à l'échelle de l'évolution), et il existe aussi des mécanismes anti-CRISPR développés par certains phages (Stanley et al., 2018).

Aujourd'hui, les phages sont aussi utilisés comme alternatives aux antibiotiques et permettent de soigner des patients infectés par des bactéries multirésistantes

(Ferry et al., 2021).

2.2.2.3.b La transduction

La transduction survient à l'issue du cycle lytique, lorsque l'ADN va être empaqueté dans la tête du phage. Généralement, l'ADN empaqueté correspondra à l'ADN du phage, mais il arrive parfois qu'il y ait des erreurs, des exceptions entraînant l'intégration de morceaux d'ADN bactérien dans la capside. De même, l'excision du prophage peut se réaliser de manière imprécise et inclure certains gènes adjacents sur le génome.

Ces phages, porteurs de portions du génome bactérien, peuvent ensuite aller "infecter" d'autres bactéries et leur transmettre la portion du génome d'origine présente dans la capside. Celle-ci aura ensuite plusieurs devenirs possibles : catabolisée, pour en récupérer les nucléotides, recircularisée, pour redevenir un plasmide si cette portion était un plasmide, ou bien recombinée, dans certains cas, si la portion est homologue à une région du génome de la cellule infectée.

Le mécanisme de transduction a été détourné par les biologistes moléculaires et est aujourd'hui utilisé dans de nombreux contextes, par exemple en biologie moléculaire, pour intégrer ou modifier des gènes dans des génomes.

2.2.2.4 Autres

D'autres mécanismes relatifs à des transferts horizontaux de gènes existent, mais ils sont moins répandus.

2.2.2.4.a Agents de transfert de gènes

Les agents de transfert de gènes (GTA, pour Gene Transfer Agents) sont des machineries procaryotes qui ressemblent à des phages, et qui transfèrent des petites portions du génome à d'autres cellules(Lang et al., 2017). Les portions vont de 4 à 14 kb en fonction des systèmes recensés, et ce sont des portions aléatoires des génomes qui sont transférés.

Certains suggèrent que les GTA sont homologues à des phages qui auraient été "adaptés" par la bactérie, un peu comme les T6SS qui ont été mentionnés précédemment. Néanmoins, il existe plusieurs systèmes, de fonction équivalente, qui auraient dérivé de manière indépendante depuis les phages : il s'agirait donc d'un cas de convergence évolutive pour les différents GTA. On retrouve des GTA chez les bactéries comme chez les Archées, mais seulement dans certains taxons.

Contrairement aux systèmes conjugatifs qui copient les éléments génomiques avant de les transférer, les bactéries produisant les GTA vont mourir en les utilisant. Malgré cela, il semblerait que les GTA confèrent un avantage sélectif aux

bactéries qui les possèdent, notamment car l'avantage est donné non pas au niveau individuel, mais au niveau de la population (West et al., 2016). En effet, les GTA sont globalement assez bien conservés dans un certain nombre de taxons et des analyses de mutations ont montré que les protéines du système étaient sous sélection purificatrice chez les *Alphaproteobacteria* (Lang et al., 2012).

Ces systèmes restent néanmoins moins bien connus que ceux mentionnés dans les sections précédentes, notamment au niveau de leur régulation. Plusieurs chercheurs ont déjà avancé l'idée que les GTA étaient beaucoup plus répandus parmi les procaryotes, par rapport à ce qui a déjà été relevé, étant donné l'importance que ces systèmes semblent avoir pour les bactéries qui les possèdent. Néanmoins, il y a actuellement peu de taxons connus chez lesquels les GTA ont été décrits.

2.2.2.4.b Vésicules contenant de l'ADN

Les vésicules de membrane externe, ou OMV (Outer Membrane Vesicules), ont été rapportées comme pouvant aussi faciliter le transfert horizontal de gène entre différentes bactéries (Dorward et al., 1989).

Les OMV ont plusieurs autres utilités, notamment pour le transport de molécules dans la communication intercellulaire. On y retrouve aussi de l'ADN, à la surface ou à l'intérieur (Kulp et al., 2010).

Il est supposé que l'ADN intégré aux cellules via l'OMV suit plus ou moins le même devenir que l'ADN récupéré par transformation naturelle : à savoir, souvent dégradé et occasionnellement intégré au génome.

2.2.2.4.c Nanotubes et système Ced

Un mécanisme plus ou moins équivalent aux systèmes conjugatifs, consistant à former des ponts, appelés nanotubes, entre des cellules adjacentes, permet aussi des échanges de larges régions génomiques (Dubey et al., 2011). Même si ce mécanisme n'est pas sans rappeler les systèmes de conjugaison, dans leur rôle et leur action, le système moléculaire impliqué est totalement différent et permet aussi de transmettre des molécules qui ne sont pas de l'ADN, contrairement au pilus conjugatif.

Un système similaire se retrouve aussi chez certaines archées sous le nom de *Ced system* (Naor et al., 2013). Celui-ci peut se mettre en place lorsque les cellules s'agrègent ensemble : des ponts cytoplasmiques intercellulaires vont alors se former, grâce à ce système, et de l'ADN chromosomique va être échangé entre les cellules.

2.2.2.5 Conclusions sur le transfert horizontal

Les mécanismes qui régissent le transfert horizontal de gènes sont variés : certains semblent spécialisés et n'ont été vu que chez quelques procaryotes, là où

d'autres sont identifiés chez une majorité. L'usage de ces mécanismes par les bactéries donne lieu à des stratégies très intéressantes et très diverses. Ainsi, on y retrouve des comportements très altruistes de bactéries qui se sacrifient pour transférer leur génome à leurs congénères, tandis que d'autres bactéries, très proches, s'entre-tuent afin que les survivantes récupèrent l'ADN des cellules mortes. La grande diversité de mécanismes impliqués dans le transfert horizontal peut grandement complexifier les génomes des bactéries. On verra par la suite, dans la partie dédiée aux îlots génomiques, que l'histoire évolutive de certaines régions génomiques peut être très chaotique. En effet, il est parfois impossible d'identifier tous les évènements de transfert de gènes qui se sont succédé.

L'avantage que les transferts horizontaux confèrent aux bactéries est sans appel : ces dernières acquièrent alors une capacité d'adaptation phénoménale, illustrée par le fait qu'on retrouve des bactéries dans tous les milieux.

2.2.3 Duplication

Au début de l'ère de la génomique, les duplications de gènes étaient vues comme la source principale de variations dans tous les domaines de la vie. Cependant, grâce aux premières dizaines de génomes séquencés, il s'est avéré que c'était les transferts horizontaux de gènes qui étaient la plus grande source d'innovation chez les procaryotes (Treangen et al., 2011).

Cette conclusion peut être néanmoins discutée avec d'autres études, utilisant des modèles phylogénétiques, qui rapportent que le taux de duplication peut être équivalent au taux de transfert (Szöllősi et al., 2012). Selon d'autres auteurs, qui recherchent uniquement des duplications et des transferts récents dans une phylogénie, le taux de duplication est presque négligeable (Tria et al., 2021).

Les duplications sont des événements assez fréquents et des variations du nombre de copies de gènes sont observables. Néanmoins, on remarque dans les populations naturelles que les génomes contiennent très rarement des gènes dupliqués très similaires, indiquant que les duplications sont souvent éliminées, probablement à cause de la redondance.

Cependant, parfois, certains gènes dupliqués survivent pour évoluer vers de nouvelles fonctions. Le fait d'avoir deux copies d'un même gène, dont une superflue, présente le double avantage de pouvoir, d'une part, réduire la sélection purificatrice qui maintient habituellement la composition des séquences et, d'autre part, augmenter la dérive génétique pour la séquence du gène. Cela signifie que les gènes peuvent avoir plus de mutations non-synonymes, entraînant une variabilité dans la composition de la séquence du gène ; ce dernier peut être alors amené vers de nouvelles fonctions, comme métaboliser de nouveaux substrats si la protéine est une enzyme.

Une autre possibilité très contextuelle est que la duplication permet au gène

d'être plus exprimé. Dans certains cas de résistance aux antibiotiques, la résistance est due à la présence de pompes à efflux qui évacuent les antibiotiques de la cellule. De fait, dupliquer les gènes qui conduisent à la formation de cette pompe peut permettre à la cellule d'améliorer sa résistance à l'antibiotique en question, en produisant tout simplement plus de pompes.

2.2.4 Gènes nouveaux

L'émergence de gènes *de novo* correspond à l'apparition d'un gène dans une zone non codante d'un génome. Ce mécanisme est plutôt décrit chez les eucaryotes, dont l'immense majorité du génome est couvert de régions non codantes, alors que chez les procaryotes, l'écrasante majorité du génome code des protéines. Néanmoins, il existe quelques exemples de gènes ayant *a priori* émergé de régions non codantes du génome.

En effet, l'émergence potentielle de petits gènes ARN régulateurs, à partir de régions intergéniques, a été observée dans certaines bactéries (Raghavan et al., 2015). Un autre cas rapporte l'émergence de gènes dans des positions qui correspondent déjà à un gène codant, mais sur une autre phase (Delaye et al., 2008), dans un mécanisme qui a été appelé *overprinting*, ou "surimpression".

L'émergence de gènes *de novo* est donc possible, même chez les procaryotes, bien que la contribution de ce mécanisme à la diversité génomique soit plutôt faible en comparaison de tous les autres mécanismes mentionnés dans ce chapitre.

Chapitre 3

Génomique comparative et pangénome

La génomique comparative, et plus spécifiquement la pangénomique sont au cœur de cette thèse. Dans ce chapitre, j'introduis les concepts principaux de la génomique comparative. Ces concepts incluent la théorie des graphes, la comparaison de séquences, la construction de familles de gènes, la comparaison de génomes et la classification des espèces chez les procaryotes. Je détaillerai aussi plus spécifiquement les problématiques liées au concept de pangénome.

3.1 Graphes en bioinformatique

Je vais maintenant introduire le volet informatique de cette thèse portant sur la théorie des graphes.

Les graphes sont utilisés pour résoudre de nombreuses problématiques dans de nombreux domaines et notamment en bioinformatique, et du vocabulaire ainsi que des concepts de la théorie des graphes sont régulièrement employés tout au long de ce manuscrit. Le public visé étant majoritairement bioinformaticien, et beaucoup de bioinformaticiens ne s'en servant pas nécessairement tous les jours, je vais à présent expliquer la terminologie utilisée dans cette thèse. Cette terminologie est normalement la même que celle communément utilisée par les chercheurs en théorie des graphes.

La théorie des graphes est une branche des mathématiques et de l'informatique utilisée dans tous les domaines scientifiques et notamment en biologie. Un graphe est un modèle composé de deux éléments, les nœuds (appelés aussi sommets) et les arêtes (appelés aussi liens) qui vont relier les nœuds. Les graphes peuvent aussi être appelés réseaux : dans la vie courante, on parle entre autres de réseau informatique ou de réseaux sociaux qui, dans leur aspect théorique, sont des graphes. Dans

le contexte du réseau social, un nœud sera un individu et les arêtes seront les relations d'amitié. Puisqu'une figure vaut mille mots, la figure 3.1 illustre un graphe relativement simple composé de 6 nœuds et de 7 arêtes.

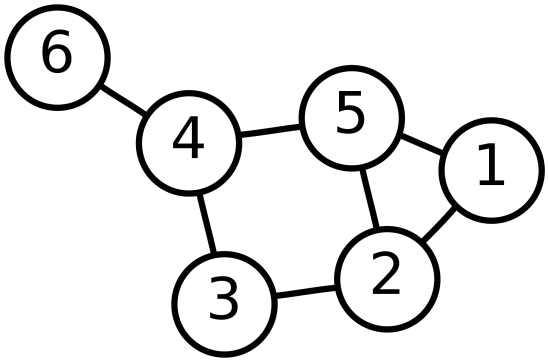


FIGURE 3.1 – Exemple de graphe
Figure illustrant un graphe. Celui-ci est composé de 6 nœuds, et de 7 arêtes. Copié de https://fr.wikipedia.org/wiki/Théorie_des_graphes consulté le 17 août 2021

Ce sont des outils très pratiques et il existe une quantité phénoménale d'algorithmes qui permettent de réaliser des opérations sur ces graphes, que ce soit pour les modifier, les explorer ou encore pour définir ou calculer des caractéristiques propres aux graphes.

De même, il existe plusieurs types de graphes, au regard du problème à résoudre, associés au vocabulaire descriptif de ces graphes. Je vais maintenant définir les termes utilisés dans ce document.

Graphe orienté

Un graphe peut être orienté, cela veut dire que les arêtes ont un sens. Soit deux nœuds u et v, il peut exister une arête qui va de u à v, mais pas forcément d'arête allant de v à u. Dans un graphe non orienté, une arête qui irait de u à v irait aussi forcément de v à u. Le graphe de la figure 3.1 est non orienté.

Dans le contexte de la bioinformatique, on utilise des graphes orientés lorsque les arêtes expriment des relations qui ne sont pas nécessairement équivalentes.

Graphe pondéré

Un graphe peut être pondéré, cela veut dire que les arêtes ou les nœuds ont un poids, une valeur qui leur est attribuée. Par exemple, en bioinformatique, un graphe dans lequel les nœuds correspondent à des gènes peut être pondéré, entre autres, lorsque les arêtes représentent la similarité entre ces gènes.

Graphe étiqueté

En bioinformatique, un graphe est généralement étiqueté, cela veut dire que les nœuds et/ou les arêtes ont des annotations, de n'importe quel type, qui permettent de les définir. Celles-ci peuvent être uniques, comme un identifiant ou descriptives,

comme une taille, un poids, mais qui donnent toujours des informations sur ce que le nœud ou l'arête représente. Un usage courant en bioinformatique est de parler de "graphe coloré" plutôt que de graphe étiqueté, notamment lorsqu'on construit des graphes de pangénomes. Néanmoins, la coloration de graphe est un champ de recherche dont le concept n'a rien à voir avec l'étiquetage et l'usage du terme "graphe coloré" en bioinformatique : le but de la coloration de graphe consiste à attribuer une couleur à chaque sommet pour que chaque paire de sommets reliés par une arête soient de couleur différente.

Sous-graphe

Un sous-graphe est une fraction d'un graphe. C'est un sous-ensemble constitué de certains nœuds, qui inclut également les arêtes reliant ces nœuds les uns aux autres.

Clique

Une clique est un ensemble de nœuds qui sont tous connectés entre eux. La clique maximale d'un graphe sera la clique avec le plus grand ensemble possible de nœuds. Dans la figure 3.1, les nœuds 1, 2 et 5 forment une clique correspondant à la clique maximale du graphe, car on ne peut pas trouver de clique avec plus de nœuds.

Chemin

Un chemin dans un graphe est un ensemble d'arêtes qui permettent de relier deux nœuds entre eux. Dans la figure 3.1, l'arête qui relie le nœud 6 et le nœud 4, ainsi que celle reliant le nœud 4 avec le nœud 3 forment un chemin entre le nœud 6 et le nœud 3 : il s'agit du chemin le plus court pour relier ces deux nœuds.

Composante connexe

Une composante connexe est un ensemble de nœuds pour lesquels il existe un chemin qui relie chaque paire de nœud de l'ensemble. S'il n'existe pas de chemin entre deux nœuds, cela veut dire qu'ils sont dans des composantes connexes différentes. La figure 3.1 est une composante connexe unique, car il existe au moins un chemin entre chaque paire de nœud.

Graphe complet

Un graphe complet est un type de graphe spécifique dans lequel tous les nœuds sont connectés entre eux. La clique maximale d'un graphe complet est l'ensemble de ses nœuds.

Densité

La densité d'un graphe est le rapport entre le nombre d'arêtes existantes et le nombre d'arêtes possibles si le graphe était un graphe complet.

Centralité

La centralité est une métrique associée à un nœud. Elle est généralement indicatrice de l'importance du nœud dans un graphe et définie par une fonction qui permet de souligner cette importance. Une fonction courante est de calculer le

nombre de plus courts chemins (parmi tous les chemins les plus courts entre tous les nœuds du graphe) qui passe par le nœud.

Fermeture transitive

Une fermeture transitive est une opération applicable sur un graphe. On parle généralement d'une fermeture transitive de taille n, ou l'opération consisté à ajouter une arête entre tous les nœuds qui sont reliés par un chemin de taille n ou moins dans le graphe original.

Clustering, ou partitionnement

Partitionner un graphe est le fait de le diviser en plusieurs parties. On va généralement chercher à classer les nœuds du graphe dans plusieurs sous-ensembles selon des règles qui vont dépendre des objectifs du partitionnement. On va parler de partitionnement à plusieurs reprises dans ce document. Dans certains cas, on va essayer de créer des parties s'établissant grâce à la topologie du graphe, en regroupant les nœuds d'une même zone dense, et en minimisant le nombre d'arêtes entre ces groupes. La taille, le nombre de parties et la définition d'une "zone dense" sont des variations possibles entre les différents algorithmes utilisés en bioinformatique. Dans d'autres cas, on utilise un système de règles qui peut se baser sur la topologie et sur les informations étiquetées sur les nœuds du graphe.

Par habitude, on va généralement parler de cluster ou de clustering et non de partitionnement, car l'anglicisme est plutôt répandu dans la terminologie bioinformatique. Il existe des dizaines, voire des centaines d'algorithmes de clustering différents. Certains répondent à des problématiques spécifiques, d'autres sont plus génériques, mais ont généralement pour but de minimiser le nombre d'arêtes entre les groupes et de maximiser la densité des groupes formés. Ces approches sont aussi appelées parfois "recherche de communautés".

Plusieurs algorithmes issus de la théorie des graphes sont régulièrement utilisés en bioinformatique. Deux d'entre eux seront mentionnés plus loin :

— MCL, pour Markov Cluster Algorithm, qui repose sur la simulation de flux dans un graphe (Van Dongen, 2000).

— La méthode dit "de Louvain" (Blondel et al., 2008) qui cherche à optimiser une fonction, la modularité, qui est une mesure de la structuration d'un graphe et qui peut être calculée de plusieurs manières par rapport au type de graphe étudié.

3.2 Principes et débuts de la génomique comparative

Le principe de base de la génomique comparative est le suivant : plus l'origine commune de deux éléments génomiques homologues est proche dans le temps, plus

leurs séquences se ressembleront. La plupart du temps, celle-ci correspondra à une séquence d'ADN, mais on pourra aussi considérer des acides aminés dans le cas des protéines, ou encore de l'ARN pour certains virus. La génomique comparative se base donc essentiellement sur la comparaison de séquences. Néanmoins, évaluer la proximité de plusieurs séquences, sachant les mécanismes qui régissent leur évolution, est loin d'être trivial.

La comparaison de séquences peut se faire soit deux à deux uniquement, soit en comparant plus de deux séquences en même temps. Comparer des séquences deux à deux est fondamentalement plus simple que de comparer plusieurs séquences simultanément. Généralement, un bioinformaticien cherchera, dans un premier temps, à identifier toutes les séquences qui sont proches les unes des autres en utilisant des comparaisons deux à deux. Ensuite, il formera des groupes de séquences similaires, plus communément appelés clusters, avant de réaliser, si nécessaire, des comparaisons multiples entre les séquences d'un même cluster.

La génomique comparative a évolué conjointement avec l'augmentation du nombre de génomes séquencés. Au cours des dernières décennies, les technologies d'acquisition de séquences, protéiques ou génomiques, ainsi que les méthodes pour les comparer ont évolué conjointement pour enrichir les bases de données existantes. De fait, à l'été 2021, on dénombre, dans les banques de données comme NCBI GenBank, un million de génomes uniques séquencés et plusieurs millions d'autres regroupés dans des métagénomes, qui sont tous comparables dans une certaine mesure. La figure 3.2 représente l'évolution des technologies de séquençages, du nombre de génomes disponible dans GenBank, et des différentes analyses et outils que je vais mentionner dans ce chapitre, ou qui ont pu être importants dans le cadre de cette thèse.

3.2.1 Les premières comparaisons de séquences

On peut dater les origines et la formalisation de la génomique comparative aux débuts des années 1960. À ce moment-là, certains chercheurs commencent à se demander comment réussir à déterminer si la ressemblance entre deux séquences d'acides aminés est due au hasard, ou non. Puisque ces questions étaient relatives à des protéines ayant des fonctions relativement proches, il semblerait qu'il était d'usage d'opérer une comparaison visuelle afin d'identifier si celles-ci étaient réellement très similaires ou non (Needleman et al., 1970 ; WATSON et al., 1961). Cette comparaison visuelle étant plutôt fastidieuse, et la significativité n'étant pas visuellement mesurable, certains chercheurs commencèrent à suggérer des approches méthodiques, basées sur des statistiques, qui permettraient de comparer et mesurer les différences entre des séquences.

Ainsi, Fitch (Fitch, 1966) suggère de mesurer la différence entre deux protéines en calculant le nombre de mutations nécessaires dans la séquence de nucléotides

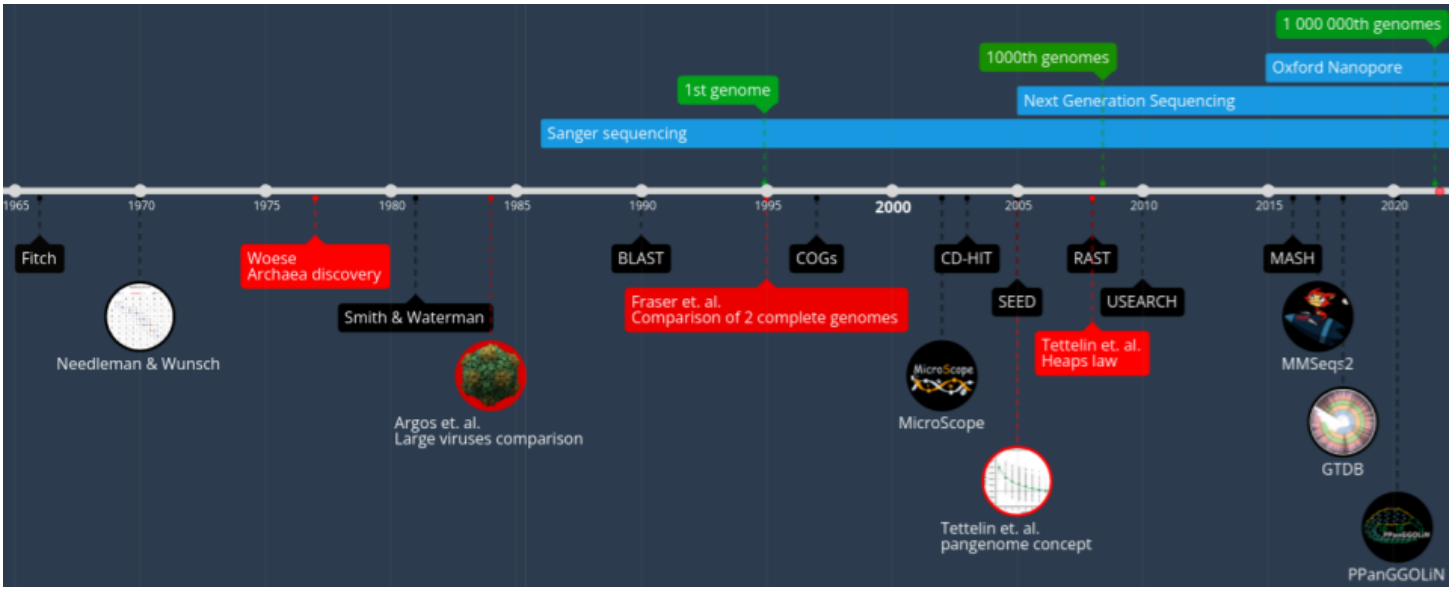


FIGURE 3.2 – Chronologie de la génomique comparative
Figure illustrant plusieurs éléments de la génomique comparative qui ont pu avoir une importance dans le contexte de cette thèse. Les éléments bleus représentent la commercialisation des différentes technologies de séquençages. Les éléments verts représentent des étapes notables dans le nombre de génomes disponibles dans GenBank. Les éléments noirs représentent des algorithmes ou des logiciels. Les éléments rouges représentent des articles d'analyses importants. Figure réalisée avec la version gratuite de TimeGraphics.

pour passer d'une séquence à l'autre ; cette approche est illustrée par le cas de l'α- et de la β-hémoglobine, deux protéines pour lesquelles l'homologie était déjà établie.

À cette époque, un article fondateur pour la génomique comparative et la bioinformatique en général est publié : celui de Needleman et Wunsch (NEEDLEMAN et al., 1970). Le but de leur algorithme est de déterminer le nombre maximal d'acides aminés alignables entre deux séquences, en considérant toutes les interruptions possibles, mais sans en énumérer toutes les possibilités ; ce qui permet d'obtenir efficacement une mesure de similarité précise entre deux protéines. Cet algorithme reste toujours largement utilisé afin de garantir un alignement optimal lorsqu'on compare deux séquences entre elles. Celui-ci sera modifié par Smith et Waterman en 1981 (T. F. SMITH et al., 1981), dans le but de rechercher entre deux séquences la sous-séquence commune la plus similaire, en autorisant certaines parties des séquences à ne pas s'aligner.

Malgré les nombreuses méthodes et leurs optimisations, le nombre de séquences que les chercheurs voudront comparer va graduellement augmenter jusqu'à ce qu'il

soit impossible d'utiliser des algorithmes exacts pour comparer une séquence aux bases de données toujours plus grandes.

La publication de BLAST en 1990 (Altschul et al., 1990) marque le début des outils de comparaisons de séquences qui utilisent des heuristiques plutôt que de réaliser toutes les comparaisons. Le principe de ces outils est d'estimer rapidement quelles séquences peuvent être similaires, pour ensuite ne réaliser des alignements que sur un sous-ensemble de séquences plutôt que sur une base de données complète. De très nombreux outils chercheront à faire exactement la même chose que BLAST, mais plus rapidement, ou pour des données ayant des caractéristiques particulières. LASTAL (Kiełbasa et al., 2011) ou Diamond (Buchfink et al., 2021 ; 2015) sont deux exemples d'alternatives à BLAST beaucoup plus rapide et très utilisés.

3.2.2 Les premières comparaisons de génomes

Avec l'obtention des premiers génomes complets d'organismes, les premières comparaisons de génomes sont réalisées. En 1984, Argos *et al.* comparent le virus CPMV (CowPea Mosaic Virus), infectant certaines légumineuses, avec des picornavirus, actifs sur des vertébrés (Argos et al., 1984). Les auteurs identifient ainsi des portions communes, à savoir des gènes dont la fonction ainsi que l'ordre semble être similaire entre les deux virus. Ils supposent alors, parmi d'autres hypothèses, que ces deux virus partageraient une histoire évolutive commune.

En 1995, Fraser *et al.* publient le génome d'un organisme procaryote, *Mycoplasma genitalium*, et le comparent avec le premier génome procaryote séquencé, *Haemophilus influenzae*. Les différences entre les génomes sont alors associées aux différences de physiologie et de métabolisme observées chez les deux bactéries (Fraser et al., 1995). Les auteurs identifient des groupes de gènes avec un ordre conservé, notamment deux groupes codant des protéines ribosomales. Pour comparer ces deux génomes, les auteurs utilisent une approche appelée BLAZE (Brutlag et al., 1993), basée sur l'algorithme de Smith et Waterman (T. F. Smith et al., 1981). Parallélisé sur un ordinateur, cet algorithme permet d'identifier, pour chaque gène, les portions les plus ressemblantes entre chaque génome.

Le problème de cette approche est que le nombre de comparaisons à réaliser est une combinatoire du nombre d'éléments. Par conséquent, on atteint très rapidement des quantités d'opérations informatiques irréalisables lorsqu'on souhaite comparer plus de deux génomes en même temps. Pour résoudre cela, d'autres approches, fréquemment utilisées de nos jours, permettent de former au préalable des groupes — ou clusters — de gènes similaires au moyen d'algorithmes conçus pour regrouper des gènes homologues en familles de gènes homologues.

3.3 Familles de gènes homologues

3.3.1 Aspect informatique du problème

Une méthode de construction de familles de gènes se découpera en deux grandes étapes : tout d'abord, un calcul des liens existant entre les gènes, généralement associé à des alignements, suivi d'un partitionnement, ou clustering, du graphe formé par ces liens.

Au travers de cette approche, on cherche à regrouper les séquences homologues. Parfois, en fonction des applications, on cherchera à avoir des groupes plus stringents : on utilisera alors les notions d'orthologie et de paralogie et on essayera de ne regrouper que les orthologues. Dans ce cas-là, les approches utilisées seront très différentes, notamment au niveau de la construction des alignements.

Ces méthodes sont de plus en plus utilisées, car elles permettent de réaliser des analyses variées : de la phylogénie, d'autres analyses de génomique comparative, des constructions de pangénomes, comme on le verra plus loin, ou encore, de l'annotation fonctionnelle. Dans ce dernier cas, on va supposer que les gènes d'une même famille ont la même annotation fonctionnelle.

3.3.2 Les approches bioinformatiques

Les approches bioinformatiques utilisent la théorie des graphes mais il n'est pas simple d'évaluer laquelle ou lesquelles sont les plus adaptées à la problématique de la construction de familles de gènes. Bien souvent, le choix de l'approche dépendra de la diversité des génomes étudiés, mais également des questions auxquelles on souhaite répondre avec ces génomes.

Ces approches seront mises à disposition soit au travers de bases de données, en utilisant des génomes publics, soit sous la forme de logiciels, pour que les chercheurs puissent les appliquer sur leurs propres génomes. Je vais maintenant mentionner les principales.

3.3.2.1 Clusters de Groupes Orthologues (ou COGs)

Une des premières méthodes de construction de familles avec l'ensemble des gènes de plusieurs génomes, et ce, de manière méthodique et généralisée, concerne l'approche originale des COGs (de l'anglais, Clusters of Orthologous Groups), publiée en 1997 (Tatusov et al., 1997).

Pour construire les COGs, les auteurs ont utilisé les sept génomes complets disponibles à l'époque, à savoir cinq bactéries (*Haemophilus influenzae, Escherichia coli, Mycoplasma Pneumoniae, Cyanobacterium synechocystis et Mycoplasma genitalium*), une archée (*Methanococcus jannaschii*) et un eucaryote unicellulaire

(*Saccharomyces cerevisiae*). L'ensemble de ces organismes représente un total de 17,967 protéines.

Ils ont ensuite comparé deux à deux toutes les protéines de chacun des génomes en utilisant une version modifiée de l'algorithme de BLAST (Altschul et al., 1990), et ont noté, s'il y en a un, le meilleur alignement (best hit en anglais) de chaque gène dans chaque génome. Lorsque le meilleur alignement est réciproque, on utilise alors les termes Bidirectional Best Hit (BBH) ou Best Reciprocal Hit (BRH).

Ces best hits sont ensuite utilisés pour construire un graphe dans lequel les nœuds et les arêtes représentent les protéines et les "best hits", respectivement. De plus, il s'agit d'un graphe orienté car un best hit n'est pas nécessairement réciproque : cette notion est illustrée, sur des données réelles, dans la figure 3.3 tirée de l'article originel décrivant les COGs.

Ensuite, les auteurs ont considéré les gènes formant graphiquement des triangles, autrement dit des cliques de taille 3, comme de probables orthologues. Effectivement, si le gène d'un génome a un best hit avec deux autres gènes provenant de deux autres génomes, alors il semble très peu probable que ceux-ci soient réciproquement des best hits à moins d'être, eux-mêmes, orthologues. Par ailleurs, l'approche COG présente l'avantage de ne pas dépendre de la valeur de similarité de l'alignement et de permettre, ainsi, de regrouper des gènes orthologues qui évoluent très vite comme des gènes qui évolueraient plus lentement. De plus, les auteurs réunissent, dans le même COG, tous les triangles qui ont un côté en commun. Néanmoins, certains des best hits, formant des triangles, peuvent être des paralogues, notamment lorsqu'il existe, au sein d'une famille, un gène unique issu d'un génome A et plusieurs gènes homologues issus d'un génome B. Dans ce cas précis, tous les gènes du génome B auront pour best hit le gène du génome A.

Finalement, les auteurs obtiennent 710 COGs, à savoir des groupes d'au moins trois gènes généralement orthologues. Pour compléter, ils alignent ensemble toutes les protéines non classifiées et forment un nouveau graphe dans lequel les nœuds sont des protéines et les arêtes des alignements. Ensuite, ils récupèrent toutes les composantes connexes de taille 3 ou plus, ajoutant 10 groupes supplémentaires. Ils obtiennent au final 720 groupes à partir de 6 814 protéines. Dans cette première analyse, les auteurs présument que des protéines appartenant au même COG ont la même fonction. Ainsi, les COGs deviennent un outil de génomique comparative pour l'annotation fonctionnelle, en plus de son utilité pour la génomique évolutive.

La base de données associée est très utilisée et régulièrement mise à jour (Galperin et al., 2021 ; Tatusov et al., 2003), avec des améliorations tant au niveau de l'alignement qu'au niveau de l'approche de clustering (Kristensen et al., 2010 ; Wolf et al., 2012). Il existe également plusieurs bases de données annexes dédiées à des groupes taxonomiques particuliers comme, entre autres, les phages

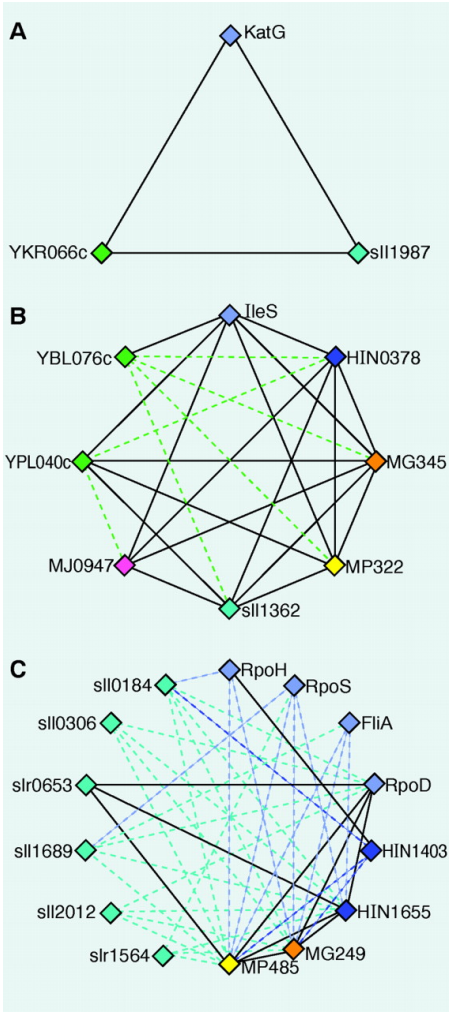


FIGURE 3.3 – Exemples de composantes connexes dans un graphe des COGs. Figure illustrant les best hits, réciproques ou non, dans un graphe des COGs. Les best hits réciproques sont en lignes pleines et ceux non réciproques en lignes pointillées. Copié de TATUSOV et al., 1997.

(KRISTENSEN et al., 2011), les archées (MAKAROVA et al., 2007), etc.

D'autres bases de données, comme EggNOG, ont aussi réutilisé le même algorithme sur d'autres données (Lars Juhl JENSEN et al., 2007).

3.3.2.2 CD-hit

L'algorithme de CD-hit se différencie principalement des autres dans la manière de construire les clusters (W. LI et al., 2001). Il existe plusieurs publications associées à CD-hit (FU et al., 2012 ; HUANG et al., 2010 ; W. LI et al., 2006) décri-

vant différentes améliorations de l'algorithme ; ce dernier a même été réimplémenté dans d'autres logiciels (Steinegger et al., 2017).

Son principe de base est le suivant : les séquence de protéines sont ordonnées de la plus longue à la plus courte et sélectionnées dans cet ordre. Elles sont ensuite comparées aux protéines "représentantes" des clusters, existantes au moment de la comparaison (la première protéine forme nécessairement un nouveau cluster). Si ces comparaisons donnent un résultat passant le ou les filtres choisis (identité, couverture, score de l'alignement...), alors la protéine est assignée au cluster ayant le meilleur alignement. En revanche, si aucune des représentantes n'est assez proche, alors la protéine initialise un nouveau cluster, dont elle sera la représentante.

Ainsi, la protéine la plus longue de chaque cluster en est la représentante. Utiliser une protéine pour représenter un cluster permet, en pratique, de réaliser moins de comparaisons que si toutes les protéines étaient comparées.

Par ailleurs, l'algorithme de CD-hit ne considérant pas à quelles espèces appartiennent les protéines traitées, il est exclusivement utilisé afin de construire des familles d'homologues plutôt que des familles d'orthologues.

3.3.2.3 InParanoid

L'objectif de inParanoid est de séparer distinctement les orthologues de certains types de paralogues (Remm et al., 2001). Dans certains cas, un gène peut être dupliqué après un évènement de spéciation. Le cas échéant, les deux gènes d'une espèce sont paralogues entre eux, mais sont aussi orthologues de l'unique gène d'une autre espèce. Au contraire, si la duplication a lieu avant la spéciation, chaque gène aura un orthologue et un paralogue dans l'autre espèce, et sera paralogue avec le second gène de sa propre espèce.

Les auteurs introduisent alors deux nouveaux termes pour différencier les types de paralogies : ainsi, les in-paralogues et les out-paralogues définissent des gènes dont l'évènement de duplication a eu lieu, respectivement, avant et après la spéciation. La figure 3.4 illustre ces concepts avec un arbre de gènes.

En effet, les méthodes décrites jusqu'alors ne distinguent pas spécifiquement les orthologues des paralogues, et c'est pourquoi l'on retrouve souvent des paralogues dans le même cluster que ce soit en utilisant les algorithmes de COG ou de CD-hit.

Dans leur approche, les auteurs commencent par aligner les protéines de chaque génome entre elles ; ces derniers utilisent BLAST dans l'article original, mais soulignent que n'importe quel logiciel d'alignement de séquence peut faire l'affaire. Considérons maintenant deux génomes A et B. Tout d'abord, des alignements croisés sont réalisés, autrement dit toutes les protéines du génome A sont alignées sur celles du génome B, et inversement. De même, des auto-alignements sont réalisés en alignant les protéines de chaque génome, A et B, sur elles-mêmes.

Ensuite, les auteurs cherchent toutes les paires de BBH entre A et B, alors

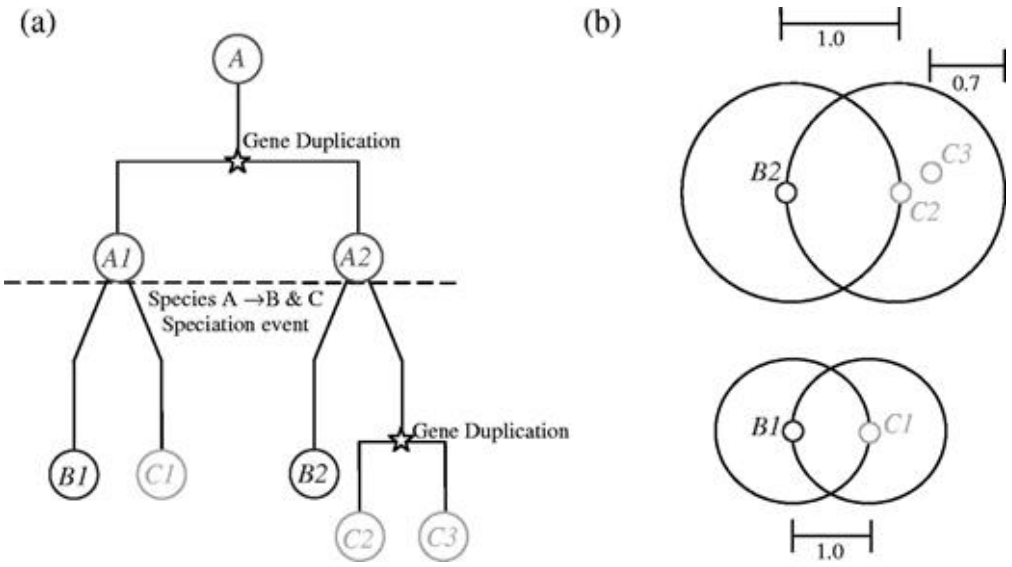


FIGURE 3.4 – Illustration des in-paralogues et out-paralogues
Figure montrant un arbre de gènes illustrant l'in-paralogie et l'out-paralogie. La protéine A de l'ancêtre "A" subit une duplication donnant deux lignées, qui mènent aux espèces "B" et "C". Dans le génome C, les gènes C2 et C3 sont des 'in-paralogues' car leur duplication a eu lieu après spéciation, et ils sont co-orthologues avec B2. Copié de O'BRIEN et al., 2005.

marquées comme orthologues et regroupées au sein du même cluster, avant de chercher à associer les in-paralogues présents dans différents clusters. L'hypothèse initiale énonce que si deux gènes de la même espèce A sont plus similaires entre eux qu'à celui ou ceux de l'espèce B, alors les deux gènes de A sont probablement des in-paralogues. Plusieurs règles vont alors être appliquées, en fonction de la topologie du graphe, et permettre de fusionner, ou non, certains des clusters précédemment formés.

Comme pour ses prédécesseurs, l'algorithme d'InParanoid a été plusieurs fois mis à jour et amélioré (ÖSTLUND et al., 2010 ; SONNHAMMER et al., 2015). Celui-ci se spécialise principalement dans le clustering de gènes eucaryotes, car différencier les orthologues des paralogues est plus important dans l'étude de ces organismes dont beaucoup ont subi des événements de duplication complète de leur génome.

3.3.2.4 OrthoMCL

Tandis qu'inParanoid est un algorithme conçu pour comparer deux génomes, orthoMCL est une solution proposée pour travailler avec un nombre indéfini de génomes (L. LI et al., 2003). La différence principale avec les autres se situe dans l'algorithme de clustering utilisé.

OrthoMCL commence par réaliser un alignement de toutes les séquences entre elles avec BLAST. Il présume ensuite que les BBH sont des orthologues et cherche, pour chaque gène, les paralogues récents probables en identifiant les gènes du même génome qui sont plus ressemblants entre eux que n'importe quel gène d'un autre génome. Ensuite, il construit un graphe représentant les séquences protéiques selon des noeuds et les relations, évoquées précédemment, selon des arêtes. Le graphe est pondéré par la p-valeur des alignements, eux-mêmes normalisés par le poids moyen des relations entre les orthologues des deux espèces des gènes considérés.

Enfin, l'algorithme de clustering MCL (Van Dongen, 2000), basé sur la théorie des flux, est appliqué sur le graphe pour former des communautés. Il s'agit probablement du premier outil bioinformatique à utiliser un algorithme de clustering, plutôt qu'un ensemble de règles, pour construire les familles de gènes.

3.3.2.5 SEED/FigFam

Le principe de SEED est assez unique dans le paysage de la construction de familles de gènes (Overbeek et al., 2005). Le projet a commencé en 2003, avec pour objectif de fournir un outil précis et fiable dans le but d'annoter les mille premiers génomes microbiens séquencés ; limite que les auteurs s'attendaient, en 2005, à dépasser aux alentours des années 2007-2008. L'aspect unique de SEED repose sur l'intervention d'experts humains pour valider les clusters obtenus en regardant leur cohérence au travers d'analyses de processus biologiques appelés sous-systèmes.

La particularité de ce projet était de faire appel à des experts de systèmes moléculaires donnés plutôt que d'annotateurs de génomes. Ainsi, la validation peut se faire sur plusieurs gènes à la fois, par des personnes connaissant bien le système moléculaire, plutôt que par des annotateurs ayant des connaissances sur l'organisme étudié, mais pas forcément sur tous les systèmes moléculaires dudit organisme. Une fois l'annotation d'un système validée, celui-ci pouvait être utilisé pour annoter automatiquement les nouveaux génomes, fournissant une source d'information *a priori* très fiable.

Dans SEED, un sous-système correspond à un ensemble de rôles fonctionnels qui réalisent un processus biologique ou interviennent dans un complexe protéique. Chaque rôle est associé à une ou plusieurs familles de protéines. Le rôle des experts était de valider l'existence ou non d'un sous-système dans les génomes étudiés en examinant l'occurrence des familles afin de les redéfinir en ajoutant/supprimant des gènes mais, également, en définissant des variants dudit système qui correspondaient à des ensembles de rôles spécifiques à certains organismes.

Depuis ces systèmes est extrait un ensemble de familles, aujourd'hui incluses dans les PATRICFams (anciennement FigFams), les familles de gènes calculées au sein de RAST (Aziz et al., 2008), un serveur web dédié à l'annotation des génomes

bactériens qui utilisait SEED à l'origine.

SEED n'est néanmoins plus maintenu en tant que tel depuis 2010.

3.3.2.6 UBLAST / USEARCH / UCLUST

Les méthodes UBLAST, USEARCH et UCLUST sont des approches permettant d'aligner des séquences proches, en les regroupant en clusters de manière plus efficace que ses prédécesseurs, notamment CD-hit (Edgar, 2010). Il s'agit possiblement d'une des premières approches dont le but est clairement de fournir d'aussi bons, voire de meilleurs résultats tout en répondant à la même problématique.

Ces approches sont dix à cent fois plus rapides que BLAST. Son principe repose sur le fait de chercher un ou plusieurs meilleurs alignements plutôt que toutes les séquences homologues d'un gène donné. Pour une séquence donnée, la base de données de séquences est triée par nombre de k-mers ('mots' de taille k) en commun, sachant que des séquences similaires tendent à partager plus de k-mers. Par la suite, en examinant cette base de données, si un alignement convenable existe, alors il est plus probable de le trouver parmi les premiers candidats. De plus, la probabilité de trouver un alignement convenable décroît au fur et à mesure que les tentatives d'alignements échouent. Ainsi, la recherche peut être arrêtée après un certain nombre de comparaisons avant même d'avoir parcouru toute la base de séquences.

UCLUST, le logiciel de clustering dédié, utilise le même algorithme que CD-hit pour définir des clusters, mais en utilisant l'approche précédemment décrite pour comparer les nouvelles séquences aux séquences représentatives des clusters. Dans l'article original, l'auteur montre que UCLUST est beaucoup plus rapide de plusieurs ordres de grandeur grâce à cette optimisation.

3.3.2.7 kClust / MMSeqs / Linclust

Originellement publiée en tant que kClust (Hauser et al., 2013), les approches de MMSeqs (Hauser et al., 2016 ; Steinegger et al., 2017) et de Linclust (Steinegger et al., 2018) en sont aujourd'hui les héritières. Le but de ces approches est de répondre à la même problématique que CD-hit et UCLUST, c'est-à-dire faire des groupes de séquences le plus rapidement possible.

kClust utilise la même stratégie de clustering que CD-hit et UCLUST, à la différence de la génération des alignements de séquences. En effet, CD-hit et UCLUST utilisent des comparaisons exactes de k-mers alors que kClust, lui, cherche des k-mers similaires tout en utilisant des k-mers de taille plus grande que ses concurrents. Le fait d'utiliser des k-mers plus longs et de chercher des correspondances similaires plutôt qu'exactes, permet à kClust d'avoir une meilleure sensibilité, tout

en gardant la vitesse accordée par l'usage des k-mers avant de réaliser l'étape d'alignement en elle-même, qui est beaucoup plus coûteuse.

De plus, l'outil utilise des k-mers dits "espacés" plutôt que des k-mers consécutifs. En pratique, cela veut dire que les k-mers, générés pour représenter la séquence, seront des ensembles ordonnés d'acides aminés sans être nécessairement consécutifs. Ceci permet une distribution plus uniforme des k-mers le long de la séquence qu'ils représentent, et cela réduit la probabilité d'avoir, par chance, un nombre conséquent de k-mers qui correspondent entre deux séquences non homologues.

MMSeqs (Hauser et al., 2016), l'héritier de kClust, utilise exactement le même algorithme pour pré-filtrer les alignements, mais cherche à être beaucoup plus sensible, afin de regrouper les homologues très distants. Pour réaliser cela, il utilise plusieurs optimisations informatiques, notamment des instructions SIMD (Single-instruction-Multiple-Data) disponibles dans les processeurs relativement récents, ou encore la possibilité de paralléliser, facilement et massivement, les calculs sur des fermes de calculs plutôt que de passer par une machine unique. MMseqs apporte plusieurs alternatives à l'algorithme incrémental utilisé par CD-hit ou UCLUST concernant l'approche de clustering. En effet, deux algorithmes alternatifs sont proposés : le premier permet de rechercher un ensemble recouvrant, à savoir un ensemble de nœuds tels que tous les autres nœuds du graphe sont soit adjacents à au moins un d'entre eux, soit font partie de l'ensemble. MMseqs utilise une approche dite gloutonne qui consiste à itérer sur les nœuds qui ont le plus de voisins pour former les clusters, jusqu'à ce que tous les nœuds du graphe soit couverts. Cependant, il n'y a aucune garantie que l'ensemble ainsi produit soit constitué d'un nombre minimal de clusters, mais cela permet de produire rapidement un ensemble recouvrant. Le second algorithme, quant à lui, extrait les composantes connexes du graphe, considérant alors que chaque composante connexe est un cluster.

MMseqs2 (Steinegger et al., 2017) introduit une nouvelle optimisation supposant que si deux séquences ont des k-mers en commun, et que ces derniers sont séparés d'exactement le même nombre d'acides aminés dans les deux séquences, alors la zone entre les deux k-mers s'aligne très probablement. Cela permet d'identifier un ensemble de portions "alignables" entre deux séquences sans calculer les alignements en eux-mêmes. MMseqs2 utilise cela comme filtre pour sélectionner uniquement les paires de séquences avec des portions alignables suffisamment longues avant de passer à l'étape d'alignement, beaucoup plus coûteuse.

Linclust (Steinegger et al., 2018) est une approche dédiée au clustering, qui se réalise sans devoir aligner toutes les séquences entre elles, contrairement aux autres approches mentionnées précédemment. Son principe est de représenter toutes les séquences par des k-mers, comparer les k-mers entre eux pour définir des "groupes de k-mers", correspondant ainsi à des regroupements de séquences ayant

des k-mers en communs. Ensuite, la plus longue séquence de chaque "groupe de k-mers" est sélectionnée pour représenter le groupe et comparée à toutes les autres séquences du groupe, avec les différentes optimisations introduites précédemment. Un lien sera placé dans un graphe entre la séquence représentante choisie et les autres séquences du groupe si la comparaison passe un certain seuil. À la fin, les séquences sont regroupées avec l'approche gloutonne de calcul d'un ensemble recouvrant. Finalement, cette optimisation permet de ne pas comparer toutes les séquences entre elles, mais uniquement celles qui font partie du même "groupe de k-mers". Par conséquent, Linclust réalise des clustering de séquences avec une complexité linéaire par rapport au nombre de séquences, contrairement à ses prédécesseurs, sans perdre beaucoup en sensibilité.

3.3.2.8 Conclusion sur la construction des familles de gènes en bioinformatique

La liste de méthodes présentées ci-dessus est loin d'être exhaustive et leur description n'est pas forcément complète car j'ai voulu mettre en avant les points qui, selon moi, sont les plus importants chez chacune d'elles. L'évolution rapide des approches de construction de familles de gènes est également à noter : celles que j'ai listées ici font partie de celles conçues exclusivement pour construire ces familles, avec parfois l'objectif de les rendre utilisables pour réaliser des annotations fonctionnelles. Néanmoins, avec l'apparition du concept de pangénome, de nouvelles approches de construction de familles de gènes, dites de "reconstruction de pangénomes", ont été développées dans le but d'être utilisées à l'échelle d'une espèce ou d'un groupe de génomes phylogénétiquement proches. La reconstruction de pangénomes étant une partie centrale de cette thèse, les outils associés seront présentés plus loin. Avant cela, une notion importante se doit d'être discutée : qu'est-ce qu'une espèce chez les procaryotes ?

3.4 Des espèces chez les procaryotes ? !

3.4.1 Définition d'espèce

Définir le terme "espèce" peut s'avérer difficile tant sa définition est souvent associée à une notion de classification. Le but est alors de répondre à la question "qu'est-ce qui se ressemble suffisamment pour être nommé de la même manière ?". Dans l'imaginaire collectif, la définition la plus communément acceptée est celle d'Ernst Mayr : "*Les espèces sont des groupes de populations naturelles, effectivement ou potentiellement interfécondes, qui sont génétiquement isolées d'autres groupes similaires*".

Chez les eucaryotes se reproduisant de manière sexuée, cette définition est raisonnablement acceptée jusqu'à une certaine limite. En effet, certains individus sont capables de se reproduire entre eux alors qu'ils appartiennent à des espèces proches, mais bien distinctes dans cette définition, car les deux populations ne sont pas interfécondes. Ces reproductions forment des hybrides parfois capables de se reproduire avec l'une des deux espèces parentes, permettant ainsi la transmission de matériel génétique d'une espèce à l'autre.

Néanmoins, cette même définition est très peu adaptée aux procaryotes qui ne se reproduisent pas entre eux ; en théorie, une cellule n'a besoin que d'elle-même pour assurer sa reproduction. Chez les procaryotes, la classification s'établit, en premier lieu, sur la base de critères phénotypiques, comme pour les eucaryotes initialement, en comparant leur morphologie, leurs capacités métaboliques ou éventuellement leur(s) hôte(s), s'ils sont infectieux. Le "Bergey's Manual of Systematic Bacteriology" (Bergey et al., 1923) est pour cela la série d'ouvrages de références sur le sujet, et se veut un guide complet permettant d'identifier et de caractériser les organismes procaryotes en se fondant sur ces divers critères phénotypiques. Malgré ses 98 ans, le projet existe toujours (la dernière édition inclut des articles de 2015), et pour des milliers d'espèces procaryotes, archées ou bactéries, il existe une description phénotypique très détaillée.

Ces premières classifications ont ensuite été successivement repensées grâce à la biologie moléculaire, aux premières séquences d'ARN ribosomiques, et enfin au séquençage de génomes bactériens complets, donnant naissance à une multitude de classifications qui peuvent se contredire ou s'accorder au regard des critères utilisés.

Par ailleurs, la notion d'espèce était déjà discutée, il y a plus de 50 ans, lorsque les outils de génomique moléculaire ou de génomique comparative sont devenus imaginables (Marmur et al., 1963). Cependant, une classification reste arbitraire, tout comme la définition d'une espèce, jusqu'à ce que les ressources taxonomiques soient suffisantes pour permettre de définir une théorie décrivant distinctement le mécanisme de spéciation des procaryotes (Mandel, 1969).

3.4.2 Approches moléculaires pour définir une espèce

Plusieurs approches ont été développées, ces dernières décennies, afin de définir si deux bactéries appartiennent à la même espèce.

En 1987, face à l'émergence de l'informatique dans le traitement des problèmes de taxonomie, un comité proposa des recommandations pour définir formellement une espèce (Wayne et al., 1987). Les auteurs affirment que seule l'espèce peut être définie en terme phylogénétique à l'échelle de la taxonomie, et que, même si la séquence complète d'ADN d'un organisme est la référence idéale pour déterminer sa phylogénie, seule l'hybridation d'ADN est un critère raisonnable justifiant sa

classification.

Ainsi, deux recommandations sont avancées :

— Une espèce doit inclure des souches dont l'ADN s'hybride à plus de 70 % dans une expérience d'hybridation ADN-ADN, et dont le ΔT_m (température à partir de laquelle la moitié des brins d'ADN sont dénaturés) varie de 5 degrés ou moins.

— Des espèces définies par la génomique, qui ne peuvent pas être phénotypiquement différenciées d'une autre espèce, ne doivent pas être nommées jusqu'à ce qu'elles soient différenciées sur la base d'une propriété phénotypique.

Néanmoins, réaliser des expériences d'hybridation ADN-ADN est techniquement difficile, et d'autres approches, qui se veulent équivalentes, gagnent alors en popularité, notamment l'usage de gènes marqueurs (EaBMG Stackebrandt et al., 1994). En particulier, le gène de l'ARNr 16S, présent chez tous les procaryotes, est un candidat intéressant car il est très aisé d'en obtenir la séquence. L'usage de l'ARNr 16S dans la reconstruction des phylogénies n'est pas original : cette méthode a été utilisée pour la première fois afin d'identifier le domaine des Archées (Carl R Woese et al., 1977). Le seuil suggéré d'utilisation de l'ARNr 16S est de 97 % d'identité, révisée à 98.7 % quelques années après (Erko Stackebrandt, 2006) ; il est important de souligner qu'il existait déjà des contre-exemples présentant certains génomes avec des taux d'hybridation ADN-ADN très faibles, mais un taux d'identité extrêmement élevé (Martinez-Murcia et al., 1992).

3.4.3 Approches par comparaison de génomes

Les premiers génomes complets soulèvent la possibilité, inexploitée mais auparavant très discutée, d'utiliser la séquence complète comme marqueur phylogénétique. La volonté de comparer ces génomes, afin de réaliser des phylogénies, soulève une question technique non triviale : comment comparer tous ces génomes en un temps raisonnable ? La définition d'un "temps raisonnable" est assez variable en fonction du besoin de chacun, et de nombreuses approches ont été développées, certaines plus précises ou plus rapides que d'autres.

La première approche à large échelle est le calcul d'ANI basé sur la comparaison de tous les gènes communs à deux génomes, introduite par Konstantinidis et al. (Konstantinidis et al., 2005). Leur approche compare tous les génomes deux à deux en alignant tous les gènes des deux génomes ensemble. Toutes les paires de gènes qui dépassent un certain seuil d'identité et de couverture sont conservées. Ensuite, une moyenne de l'identité en nucléotides est calculée entre les deux génomes, grâce à ces paires de gènes. Cela permet de donner une métrique relativement simple qui indique la ressemblance génétique entre deux génomes et qui, selon les auteurs, ne devrait pas être trop affectée par le transfert horizontal

de gènes, lorsque des génomes proches sont considérés. Les auteurs montrent que la métrique de l'ANI corrèle avec les mesures d'hybridation ADN-ADN utilisées dans les recommandations de l'article de Wayne *et al.* (Wayne et al., 1987) Les auteurs de cet article concluent qu'il existe des frontières claires entre des groupes de souches de même espèce, et que leur approche permet d'inclure l'écologie des souches en plus de la distance évolutive. Néanmoins, le problème de cette approche est que le nombre de comparaisons à réaliser est une combinatoire du nombre de génomes considérés. Ainsi, au fur et à mesure que le nombre de génomes augmente, cette approche va devenir irréalisable.

En 2016, le problème de la comparaison de milliers de génomes, en un temps raisonnable, fût partiellement résolu par la méthode Mash (Ondov et al., 2016). Mash utilise la technique du MinHash pour transformer de larges séquences en une représentation réduite et compressée du génome. Ensuite, Mash utilise ces représentations réduites, pouvant être plus petites de plusieurs ordres de grandeurs, pour réaliser les comparaisons entre les génomes et obtenir des distances dont l'erreur, par rapport à la "vraie" distance, sera limitée uniquement par la taille de la représentation utilisée. Les représentations utilisées par Mash sont encore une fois des k-mers : quelques milliers seront suffisants pour représenter un génome procaryote avec assez de sensibilité. Pour calculer les distances entre deux représentations, l'opération consiste à réaliser une distance de Jaccard entre les deux ensembles de k-mers ; les auteurs montrent que cette distance de Jaccard corrèle avec la valeur d'ANI. Par ailleurs, Mash permet de comparer 54,118 génomes en 33h sur un unique CPU, là où il faudrait peut-être plusieurs mois de calculs avec l'approche de l'article original sur l'ANI.

Depuis Mash, plusieurs approches ont été mises au point en utilisant le même concept de base, à savoir des représentations de génomes impliquant quelques milliers k-mers ou d'autres types de représentations réduites des séquences : ces nouvelles méthodes réduisent encore grandement le temps de calcul. On peut citer, entre autres, Dashing (Baker et al., 2019), Sourmash (Brown et al., 2016) ou BinDash (XiaoFei Zhao, 2019), qui permettent aujourd'hui des comparaisons encore plus rapides : par exemple, Dashing annonce pouvoir comparer 87K génomes en 6 minutes. Ces approches permettent, pour l'instant, de résoudre le problème de la comparaison de milliers de génomes. Néanmoins, celles-ci ne fonctionnent que sur les génomes proches : plus les génomes seront éloignés, moins les distances estimées seront fiables. D'autres approches existent, un peu plus lentes mais sensiblement plus rapides que l'originale, comme fastANI (Jain et al., 2018) dont les distances corrèlent mieux avec l'ANI que les distances Mash lorsque les génomes sont partiellement fragmentés ou trop distants.

Ces outils permettant la comparaison de milliers de génomes donnent alors la possibilité de se demander si, effectivement, des espèces procaryotes existent

bel et bien ou non. Une autre question subsiste alors : existe-t-il des mécanismes de spéciation, chez les procaryotes, suffisamment marqués pour permettre de les discriminer ?

3.4.4 Vers un consensus de l'espèce chez les procaryotes ?

Plusieurs articles avancent des arguments pour ou contre la notion d'espèces chez les procaryotes. Certains penchent en faveur d'un continuum génétique, contre l'existence de clusters distincts (Luo et al., 2011). D'autres affirment que les transferts horizontaux ne sont pas suffisants pour 'tordre' les barrières des espèces (Shapiro et al., 2012). Néanmoins, toutes ces études avaient le défaut de ne considérer qu'un faible nombre de génomes, incluant peu de taxons, et généralement uniquement des souches isolées en laboratoire, ne représentant pas adéquatement la diversité présente dans la nature.

Lors de la publication de fastANI (Jain et al., 2018), les auteurs ont tenté d'exploiter la masse de données que leur outil était capable de traiter afin de répondre à cette question : "*Existe-t-il effectivement des limites claires entre les espèces chez les procaryotes ?*" Dans leur article, les auteurs considèrent l'ensemble des génomes disponibles, à l'époque, dans la base de donnée du NCBI, à savoir 91,761 génomes. Ces derniers illustrent notamment les résultats de ces comparaisons au travers de la figure 3.5.

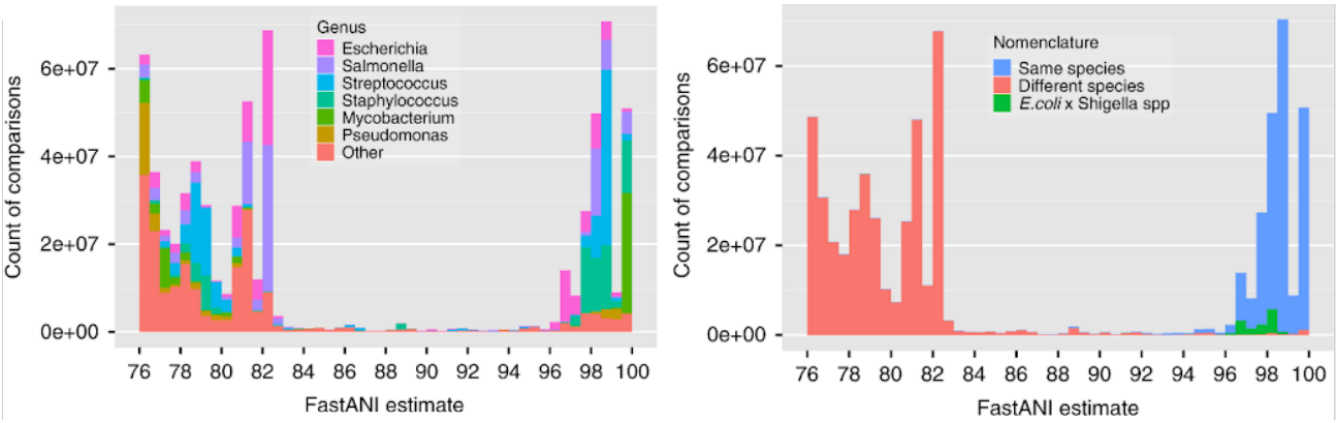


FIGURE 3.5 – Analyse d'ANI sur 90k génomes

Figure illustrant, selon les auteurs, une démarcation entre les espèces procaryotes. Résultats issus de la comparaison de 90k génomes réalisée avec fastANI.Le premier graphique est un histogramme montrant la distribution des valeurs d'ANI parmi les 90k génomes, pour les valeurs de 76% à 100%. Le second graphique montre la distribution des valeurs d'ANI en fonction de la nomenclature des génomes comparés. Copié de (Jain et al., 2018)

Cette figure illustre assez nettement une séparation entre des génomes déjà classés dans la même espèce et des génomes classés dans des espèces différentes ; avec une exception notable pour *Escherichia coli* et les espèces du genre *Shigella* qui sont, en réalité, des *E. coli* dotées de capacités infectieuses particulières (Van den Beld et al., 2012). Se basant sur ces résultats, les auteurs de fastANI soutiennent que le critère de 95 % d'ANI, déjà utilisé par le passé pour définir des espèces, semble être un bon critère pour discriminer les génomes d'espèces différentes.

Cependant, leurs conclusions et leur approche sont critiquables par plusieurs aspects :

1. La majorité des génomes du NCBI appartiennent à une poignée d'espèces très étudiées, donc l'écrasante majorité des comparaisons sont en réalité des comparaisons entre quelques-unes des espèces les plus connues uniquement.
2. Les génomes des espèces très étudiées ont beaucoup de redondance : les souches issues d'analyses épidémiologiques sont très clonales et les laboratoires de recherche se partagent des souches qui ont été séquencées plusieurs fois à l'occasion de différents projets.
3. Les auteurs se basent sur une taxonomie (celle du NCBI) dont la variabilité génétique entre les groupes de même rang taxonomique a été soulignée (Parks et al., 2018).

Dans leur discussion, les auteurs mentionnent une tentative de résoudre les problèmes 1 et 2 précédemment mentionnés : ils ont ainsi échantillonné cinq génomes, parmi toutes les espèces ayant plus de cinq génomes, sur lesquels ils ont reproduit leur analyse. La délimitation à 95% d'ANI est alors beaucoup moins marquée (voir la figure 3.6 de Jain et al. (Jain et al., 2018)).

Néanmoins, cette approche présente un biais : celui de la taxonomie utilisée, avec des espèces qui sont en réalité déjà définies. Les auteurs concluent que l'origine de cette limite peut être liée soit à la réduction très importante de la fréquence de recombinaison lorsque les séquences ont moins de 95% d'identité, soit à la sélection naturelle due à une forte compétition en milieu naturel, soit à des processus stochastiques neutres.

Plusieurs autres équipes de recherche ont tenté de répondre aux différentes problématiques, introduites (ou revisitées) par Jain et al. (Jain et al., 2018), pour essayer de conclure sur l'existence d'espèces chez les procaryotes. Une approche tentant de résoudre le problème potentiellement lié à la base de données a été d'utiliser exclusivement des génomes reconstruits à partir de métagénomes (Olm et al., 2020). Les auteurs ont utilisé 5,203 génomes issus de 1,457 échantillons métagénomiques différents pour tester l'existence de clusters et évaluer les métriques potentielles pouvant les discriminer. En premier lieu, les auteurs identifient également, dans leur jeu de données, une limite autour de la valeur de 95 % d'ANI.

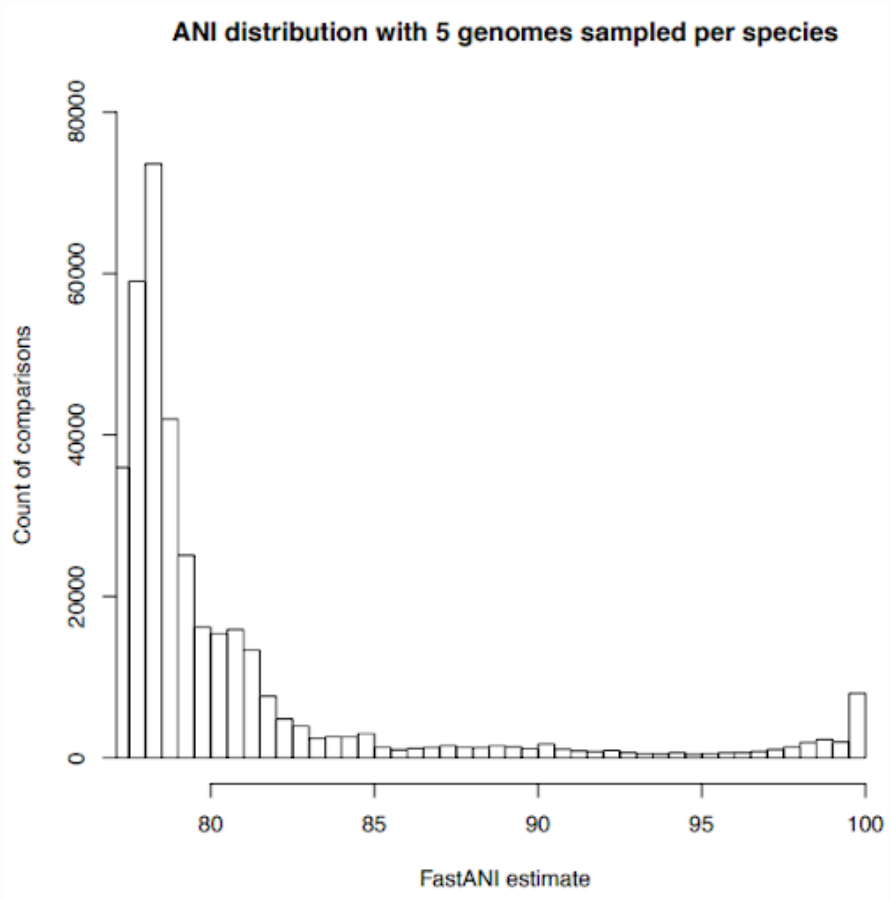


FIGURE 3.6 – Analyse d'ANI avec 5 génomes par espèces
Figure illustrant toujours, selon les auteurs, une démarcation entre les espèces procaryotes, même si celle-ci est moins marquée. Résultats issus de la comparaison de 3,750 génomes de 750 espèces du NCBI réalisée avec fastANI. Copié des données supplémentaires de (JAIN et al., 2018).

Ceux-ci ont aussi tenté d'identifier la force évolutive à l'origine de cette limite distincte en estimant, au travers de deux méthodes, les taux de recombinaison homologue en fonction de l'ANI des génomes et en calculant le ratio des mutations non-synonymes sur les mutations synonymes à l'échelle des différents génomes. Ils ont ainsi déterminé un déclin très important dans la recombinaison homologue de 100 à 95 % d'ANI, jusqu'à être négligeable à 95 %. Pour les auteurs, ces données indiquent qu'il existe effectivement des espèces dans les génomes procaryotes, et que cette délimitation est provoquée par les recombinaisons homologues entre les génomes capables de les réaliser.

Les différentes approches que j'ai décrites ainsi que leurs conclusions sont tou-

jours régulièrement discutées (C. S. Murray et al., 2021). Certains chercheurs pensent que les travaux autour de l'existence et de la délimitation des espèces ne doivent plus se centrer sur l'existence même des espèces, qui pour eux est désormais actée (Rodriguez-R et al., 2021), mais principalement sur les génomes dont l'ANI se situe dans les limites de 90-95%. D'après ces mêmes chercheurs, le but est désormais d'essayer de comprendre les raisons de ces exceptions qui peuvent être liées, par exemple, à des spécificités écologiques ou fonctionnelles (Rodriguez-R et al., 2021).

3.4.5 Homogénéisation de la taxonomie

Comme mentionné précédemment à plusieurs reprises, les taxonomies actuelles ont pour principal défaut d'être historiquement basées sur les phénotypes et complémentées par un mélange de différentes méthodes, formant des groupes à l'homogénéité très diverse dans l'arbre du vivant. En 2018, un groupe de recherche a proposé une taxonomie sur l'ensemble des procaryotes en se basant exclusivement sur des données génomiques : GTDB (en anglais, Genome Taxonomy DataBase) (Parks et al., 2018).

La figure 3.7 illustre la problématique liée à la taxonomie du NCBI, largement utilisée, et la compare à la solution proposée par les auteurs.

Leur approche utilise 120 familles de protéines ubiquitaires, parmi les procaryotes, qui ont été évaluées précédemment comme phylogénétiquement informatives. Les auteurs réalisent des alignements multiples pour toutes les familles, concatènent les alignements des différentes familles pour chaque génome puis génèrent des arbres phylogénétiques décrivant les relations entre les différents génomes. Ensuite, les auteurs tentent de mesurer les limites entre les différents génomes en calculant des mesures de divergence évolutive et en établissant des seuils pour les différents rangs taxonomiques. Finalement, ils tentent de nommer de manière systématique les différents rangs taxonomiques dans l'arbre, en respectant les sémantiques d'usage, et en essayant de réutiliser, si possible, des noms d'espèces équivalents à ceux en vigueur. Chaque espèce est associée à un génome de référence qui doit être le plus complet, le plus fiable et le plus représentatif de l'ensemble des autres génomes appartenant à cette espèce. Une grande partie de leur travail consistait à harmoniser ces noms ainsi que le choix des génomes de référence, pour que ceux-ci collent le plus possible aux noms et génomes de la littérature.

L'existence d'une taxonomie de ce type est une petite révolution à l'échelle de la génomique comparative, car cela donne une homogénéité à la notion d'espèce, inexistante jusqu'à présent à l'échelle des procaryotes. "Like cigars, a good species and a good classification is one which satisfies." (Mandel, 1969)

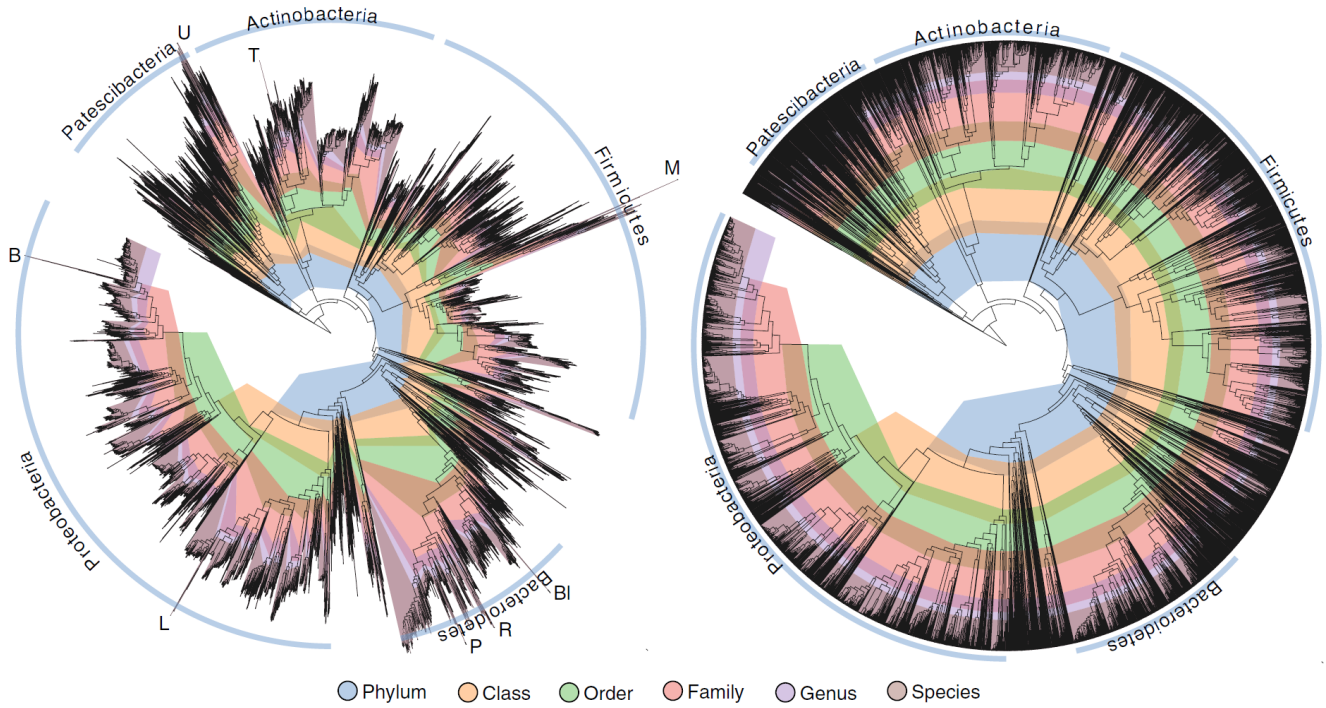


FIGURE 3.7 – Comparaison de l'homogénéité des rangs taxonomiques dans les bases de données NCBI et GTDB
Figure illustrant les rangs taxonomiques du NCBI, projetés sur l'arbre calculé par les auteurs (à gauche), en comparaison de l'actualisation des rangs taxonomiques proposés par les auteurs (à droite). Copié de (GAUTREAU, 2020) et adapté de (PARKS et al., 2018)

3.5 Pangénome

3.5.1 Définitions et origine

Un pangénome est l'union de toutes les séquences présentes dans un ensemble de génomes. Dans cette thèse seront considérés uniquement les pangénomes construits à partir des gènes. Il existe aussi des pangénomes construits directement depuis les séquences, notamment pour les eucaryotes chez lesquels la diversité dans le contenu en gènes est moindre. Tel est le cas également pour certaines applications chez les procaryotes, notamment pour différencier les souches d'une même espèce ou encore pour réaliser de la génomique comparative à l'échelle nucléotidique. Le concept de pangénome est intrinsèquement lié au concept d'espèce, car on va bien souvent reconstruire les pangénomes à l'échelle d'un groupe taxonomique dont le niveau considéré sera, généralement, celui de l'espèce.

En génomique microbienne, le terme pangénome a été introduit dans l'article de Tettelin *et al.* (Tettelin et al., 2005), présentant le premier pangénome reconstruit, ainsi que dans celui de Medini *et al.*, discutant du concept de pangénome (Medini et al., 2005). Dans leur article, Tettelin *et al.* reconstruisent le pangénome de *Streptococcus agalactiae* en utilisant, à l'époque, les 6 génomes disponibles de cette espèce. Pour reconstruire le pangénome, ils comparent les protéines de chaque paire de génomes au moyen de trois méthodes : 1) alignement de toutes les protéines selon l'algorithme développé par Smith and Waterman (T. F. Smith et al., 1981), 2) alignement des ORF d'un génome contre la séquence de l'autre génome, et 3) alignement des protéines prédites d'un génome contre la séquence d'ADN traduite de l'autre génome. Ensuite, ils considèrent qu'un gène est conservé entre deux souches si au moins l'une de ces trois méthodes produit un alignement avec au moins 50% d'identité et 50% de couverture. Les auteurs introduisent alors les concepts de génome cœur ou *core*, qui correspond à l'ensemble des gènes partagés par la totalité des génomes considérés, et de génome accessoire (en anglais, accessory ou dispensable), qui correspond à tous les autres gènes du pangénome. Ils définissent aussi une approche pour identifier les îlots génomiques qui représentent, selon eux, les portions d'ADN de plus de 5kb dont les gènes ne sont partagés par aucune autre souche.

Les auteurs notent que plus il y a de souches considérées, plus le *core* génome est petit. Ils modélisent alors son évolution à l'aide d'un modèle de décroissance exponentielle. Par projection, ils établissent que le *core* génome atteint un minimum de 1806 gènes, et estiment que celui-ci devrait rester relativement constant au fur et à mesure que de nouveaux génomes sont ajoutés. Une modélisation de leur projection est illustrée dans la figure 3.8.

De plus, ils modélisent, par une exponentielle décroissante, le nombre de nouveaux gènes introduits par chaque nouveau génome et estiment que chacun d'entre eux devrait apporter en moyenne 33 nouveaux gènes et, par extension, que la taille du pangénome est infinie. Les auteurs enrichissent alors le concept de pangénome : un pangénome sera dit "ouvert" si sa taille augmente toujours avec l'apport de nouvelles souches et sera dit "fermé" si celle-ci n'augmente plus après l'ajout d'une quantité finie de souches. De fait, ils qualifient le pangénome de *Bacillus anthracis* de "fermé", car l'ajout d'une quatrième souche à ce pangénome n'ajoute aucun nouveau gène.

Cependant, cette approche présente plusieurs défauts et sera, par la suite, revue, corrigée et améliorée par de nombreuses méthodes, modélisations et définitions alternatives ; néanmoins, leur approche étant la première analyse de pangénomique, celle-ci pose les bases du domaine.

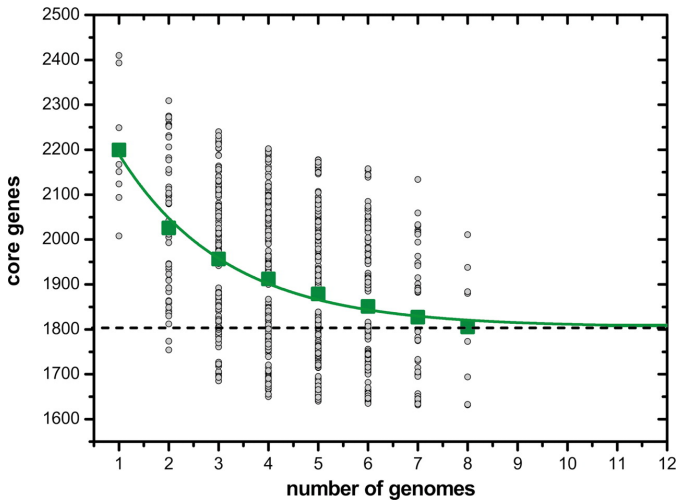


FIGURE 3.8 – Évolution du *core* génome chez *Streptococcus agalactiae*, modélisée par une loi exponentielle
Figure illustrant l'évolution du *core* génome en fonction du nombre de génomes dans le pangénome de *Streptococcus agalactiae* lorsque celle-ci est modélisée par un modèle exponentiel. Copié de TETTELIN et al., 2005

3.5.2 Modéliser les parties du pangénome

Dans l'article original de Tettelin et al. (TETTELIN et al., 2005), les auteurs modélisent la taille du *core* génome et le nombre de gènes spécifiques par des équations exponentielles décroissantes. L'hypothèse sous-jacente au fait d'utiliser ces équations est très forte puisqu'elle suppose que, pour un nombre suffisant de génomes, l'apport de nouveaux gènes devient constant, impliquant que la taille du pangénome est infinie et que le *core* génome est nécessairement stable. L'hypothèse du *core* génome strict est également très forte, car dès qu'un gène est absent d'un des génomes celui-ci est exclu du *core*. Or, il suffit d'une erreur d'assemblage, d'annotation du génome, de reconstitution du pangénome, ou encore de travailler avec des génomes partiels pour qu'un gène soit manquant. Le premier pangénome réalisé sur un nombre de génomes plus important viendra mettre au défi cette modélisation en ce qui concerne le *core* génome.

Dans leur article, Hogg *et al.* construisent un pangénome avec treize génomes de *Haemophilus influenzae* et proposent une alternative à la modélisation selon une loi exponentielle décroissante en questionnant notamment le fait que le modèle original présume que le nombre de nouveaux gènes est infini (HOGG et al., 2007). Ils proposent un modèle alternatif dans lequel chaque gène correspond à une variable aléatoire de Bernoulli ayant une certaine probabilité de succès, ici la fréquence de chaque gène dans la population. Par ailleurs, les auteurs supposent que

chaque gène est indépendant, ce qui est faux, mais permet de réduire radicalement la complexité du modèle. De plus, ils indiquent considérer cette hypothèse raisonnable dans la plupart des cas. Puisque la vraie fréquence des gènes est inconnue, les auteurs ont choisi d'associer chaque gène à k classes discrètes de gènes dont chacune a une fréquence théorique dans la population. Dans leur modèle, chaque gène est associé à une des classes de manière indépendante selon une distribution de probabilité. Dans leur étude, les auteurs ont fait le choix de fixer k=7, ainsi que les fréquences théoriques de chacune des classes entre 0.01 et 1, de manière homogène ; la classe ayant une fréquence théorique de 1 correspond au *core genome*. Les auteurs ont cherché à appliquer leur modèle sur huit génomes et à l'évaluer sur l'ensemble des treize génomes. La prédiction du modèle sur la taille du *core* génome et sur le nombre de nouveaux génomes ajoutés par chaque souche leur a semblé satisfaisante. Avec les treize souches, les auteurs obtiennent un pangénome de 2800 gènes et leur modèle indique que le pangénome de l'espèce contient un peu plus de 5 000 gènes mais que cela demanderait l'usage de centaines de souches pour qu'il soit complet. Leur modèle sera amélioré quelques années après par Snipen *et al.* qui intègreront une variabilité dans le nombre de classes et dans la fréquence théorique par classe, dont la valeur idéale sera estimée (Snipen et al., 2009).

De leur côté, Willenbrock *et al.* reconstruisent un pangénome avec 32 génomes de *E. coli* et de plusieurs espèces de *Shigella* (Willenbrock et al., 2007). Leur analyse du pangénome les amènent à réfuter l'hypothèse selon laquelle le *core* génome ne semble plus décroître, mais reste à confirmer, toujours de manière empirique, la théorie selon laquelle l'ajout de nouveaux gènes est constant. Les auteurs proposent une formule alternative en ajoutant une double racine carrée du nombre de gènes pour modéliser l'évolution du *core* génome, mais cette alternative ne sera pas vraiment reprise par la suite.

Un article de Tettelin *et al.* viendra proposer une méthode alternative pour modéliser le nombre de gènes dans les parties du pangénome (Tettelin et al., 2008). Les auteurs proposent d'utiliser une loi de Heaps, déjà employée de manière équivalente dans le contexte de la linguistique pour estimer le nombre de mots qu'une langue contient à partir de documents écrits dans cette langue. Une illustration de la projection de cette loi est visible dans la figure 3.9

L'usage de cette loi permet indifféremment d'autoriser conjointement la croissance ou la décroissance (ou la stabilité) des tailles des différentes parties du pangénome, ainsi que la réduction graduelle du nombre de nouveaux éléments gagnés par génome ajouté, sans nécessairement le forcer. La loi de Heaps est la suivante :

$$p(N) = \kappa N^{\gamma}$$

N est le nombre de génomes. Le paramètre κ est une constante de proportionnalité tandis que γ reflète la tendance de la fonction.

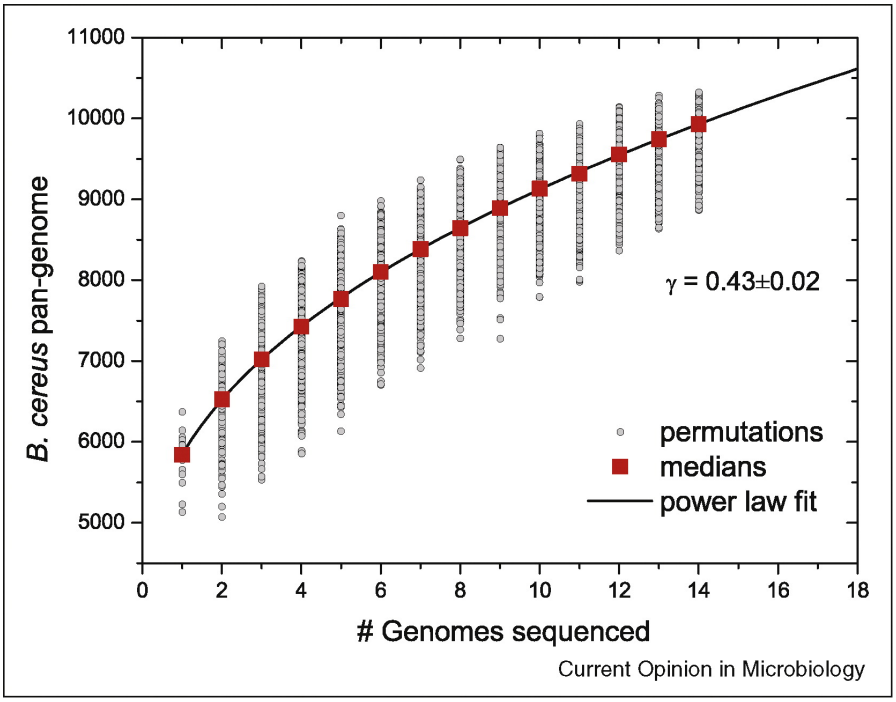


FIGURE 3.9 – Évolution du pangénome chez *Bacillus cereus*, modélisée par une loi de Heaps
La figure illustre l'évolution du pangénome en nombre de familles de gènes compte tenu du nombre de génomes de *Bacillus cereus* lorsque celle-ci est modélisée par une loi de Heaps. Copié de TETTELIN et al., 2008.

— $\gamma > 0$: donnera une croissance.

— $\gamma = 0$: donnera une tendance stable.

— $\gamma < 0$: donnera une décroissance.

Par ailleurs, lorsque la loi de Heaps est appliquée à la taille (en nombre de familles de gènes) du pangénome, elle permet d'indiquer si le pangénome est fermé ou ouvert. Lorsque $\gamma > 0$, le pangénome est dit ouvert, sinon il sera considéré comme fermé. Les auteurs ont testé leur approche sur le pangénome de différentes espèces et soulignent que l'ouverture est probablement indicatrice de la diversité des environnements dans lesquels l'espèce est présente, là où au contraire un pangénome fermé indiquerait un flux très limité de nouveaux gènes ou une population très clonale.

3.5.3 Partitions du pangénome

Tettelin *et al.* partitionnaient le pangénome en 2 parties : le *core* pour les gènes présents dans tous les génomes, et les gènes accessoires, à savoir tous les autres gènes (Tettelin et al., 2005). Par la suite, le modèle développé par Hogg *et al.* suggérait qu'il pouvait y avoir plus de deux partitions et en ont utilisé sept dans leur analyse (Hogg et al., 2007). Un avantage de leur approche est qu'elle est probablement la première à pouvoir autoriser des gènes du *core* à ne pas être dans tous les génomes, mais ça n'était *a priori* pas leur intention. Snipen *et al.* améliorèrent ce modèle en donnant la possibilité d'avoir un nombre variable de partitions dans le pangénome (Snipen et al., 2009).

En parallèle, Koonin *et al.*, dans une analyse globale des génomes procaryotes (Koonin et al., 2008), disponibles à l'époque au travers de eggNOG (Lars Juhl Jensen et al., 2007) (338 génomes), ainsi que Makarova *et al.*, dans une analyse spécifiquement faite sur des génomes d'Archées (Makarova et al., 2007), considèrent un partitionnement en 3 parties, lié au fait qu'ils pouvaient modéliser la courbe des fréquences des familles de gènes d'un pangénome avec 3 modèles exponentiels. Selon ce découpage, on parle alors de *core*, renommé *persistent* par certains groupes, pour considérer les gènes présents dans la majorité des génomes, de *shell* pour ceux présents dans un nombre intermédiaire de génomes et enfin, de *cloud* pour les gènes rares présents dans peu de génomes.

L'explosion de génomes disponibles dans les bases de données augmentera grandement la popularité de ce type d'approche, ce qui amènera l'émergence d'un nombre conséquent de logiciels permettant de reconstruire les pangénomes et d'en calculer les parties. La majorité des logiciels utiliseront ainsi les différentes manières de partitionner et de modéliser le pangénome, au regard des choix, affiliations et préférences de leurs auteurs.

3.5.4 Construire un pangénome

En premier lieu, construire un pangénome revient généralement à en calculer les familles de gènes. On a vu précédemment quelques approches dédiées au calcul des familles de gènes, de manière générique, dans un ensemble de génomes. Les méthodes qui vont permettre de reconstruire un pangénome sont plus spécifiques, dans le sens où elles sont conçues pour calculer des familles de gènes parmi des génomes qui appartiennent généralement à la même espèce, ou qui sont phylogénétiquement proches.

Il existe aussi des outils de pangénomique qui utilisent uniquement les séquences des génomes, et qui sont généralement plus dédiés aux eucaryotes, même si certains sont aussi utilisés pour certaines applications très spécifiques chez les procaryotes.

Une liste non exhaustive d'outils de reconstruction de pangénome, utilisables

en ligne de commande, est disponible dans la table 3.5.4, indiquant la méthode de reconstruction de familles, le type de partition utilisé et la scalabilité en nombre de génomes indiqué dans la littérature.

Ces outils se distinguent généralement au niveau des seuils utilisés, des méthodes de construction de familles, ainsi que des opérations "annexes" qu'ils peuvent réaliser comme des figures ou des sorties qui permettent d'autres applications.

Logiciel	Construction de familles	Modèles	Scalabilité	Citations
panOCT	Orthologues positionnels	None	300	165
PGAP	InParanoid et MCL	core, accessoire	10	332
GET_HOMOLOGUES	BBH, COG ou OrthoMCL	core, accessoire	100s	534
ITEP	OrthoMCL	core, accessoire	200	87
LS-BSR	USEARCH	core, accessoire	1000	198
Roary	CD-hit et OrthoMCL	core, shell, cloud	>1000s	2007
micropan	clustering hiérarchique	k partitions	>300	90
PIRATE	CD-hit et MCL	core, accessoire	>1000s	41
Panaroo	CD-hit	core, accessoire	>1000s	52
PanACoTA	MMseqs2	Multiple	>1000s	8

TABLE 3.1 – Outils pour la construction de pangénomes
Table listant plusieurs outils de pangénomique, récents ou populaires, avec leurs caractéristiques vis-à-vis de la construction des familles et des modèles utilisés. La table est ordonnée sur l'année de sortie de l'article correspondant. On y retrouve panOCT (FOUTS et al., 2012), PGAP (Y. ZHAO et al., 2012), GET_HOMOLOGUES (CONTRERAS-MOREIRA et al., 2013), ITEP (BENEDICT et al., 2014), LS-BSR (SAHL et al., 2014), Roary (PAGE et al., 2015), micropan (SNIPEN et al., 2015), PIRATE (BAYLISS et al., 2019), panaroo TONKIN-HILL et al., 2020 et PanACoTA (PERRIN et al., 2021)

D'autres outils de pangénomique, non centrés autour des gènes mais effectuant des partitions de pangénomes, existent : panseq (LAING et al., 2010) qui détermine des régions génomiques *core* ou accessoire, ou encore Piggy (THORPE et al., 2018) qui construit un pangénome uniquement avec les régions intergéniques chez les bactéries.

D'autres encore sont centrés sur la visualisation du pangénome, comme panTetris (HENNIG et al., 2015), panViz (PEDERSEN et al., 2017) ou encore Panache (DURANT et al., 2021). Ces outils se différencient par les visualisations qu'ils réalisent et le type de données auxquelles ils sont adaptés.

Certains outils sont dédiés à l'analyse de pangénome sur des données métagénomiques, comme MSPMiner(PLAZA OÑATE et al., 2019), PanPhlAn (SCHOLZ et al., 2016) ou mOTUpan (BUCK et al., 2021). Leurs différences, ici, se résument

aux données qu'ils traitent (des lectures, des MAGs ou des bins) ou encore aux modèles utilisés pour le partitionnement.

Enfin, il existe une multitude de serveurs web qui mettent à disposition des pangénomes pré-calculés, ou qui permettent de calculer des pangénomes en ligne, notamment EDGAR (Blom et al., 2009), PanWeb (Pantoja et al., 2017), PGA-web (X. Chen et al., 2018), ou bien MicroScope (Vallenet et al., 2020). Tous ces outils en ligne diffèrent, entre autres, par les données que l'utilisateur peut utiliser, la possibilité pour l'utilisateur de sélectionner les génomes à ajouter dans son pangénome, la taille des pangénomes réalisables et la diversité des figures générées.

Chapitre 4

Îlots génomiques

Les îlots génomiques sont le cœur de la variabilité des génomes procaryotes et récipient de ce qui fait leur capacité d'adaptation phénoménale : un point d'arrivée pour les gènes acquis par transfert horizontal. Ce chapitre introduit le concept d'îlots génomiques et présente les méthodes utilisables pour les détecter. Il introduit aussi les quelques méthodes qui permettent d'étudier l'évolution des îlots génomiques dans plusieurs génomes.

4.1 Origine et définition

Le terme "îlot génomique" est apparu pour la première fois dans un article de revue datant de l'an 2000 et concernant les îlots de pathogénicité, à savoir des régions du génome incluant des facteurs de virulence chez les bactéries pathogènes (Hacker et al., 2000).

Les îlots génomiques (GI, pour Genomic Islands en anglais) sont des éléments variables dans les génomes d'une espèce qui peuvent conférer des avantages sélectifs aux bactéries qui les possèdent. Par exemple, ils peuvent faciliter les infections lors de la colonisation d'un hôte, leur permettre d'échapper à la réponse immunitaire, ou bien d'entrer et sortir des cellules d'un hôte dans le cas d'une bactérie intracellulaire.

Globalement, les GI sont des régions larges mesurant, pour la plupart, entre 5 et 200 kb, mais certaines peuvent être beaucoup plus grandes et faire partie de plasmides ou de prophages. Ces régions présentent souvent des différences dans leur taux de GC, ainsi que dans l'usage des codons, par rapport au reste du génome ; ce qui illustre l'acquisition de ces éléments par transfert horizontal. La différence en taux de GC n'est pas une règle absolue, notamment si la bactérie d'où provient une partie des éléments transférés est phylogénétiquement proche de la bactérie hôte. Les GI sont souvent flanqués de séquences répétées comme des intégrases,

des transposases et des séquences d'insertions. Les GI sont aussi régulièrement associées à des ARN de transfert (ARNt) : cette association est potentiellement liée au fait que certains phages, ainsi que des ICE, utilisent les ARNt comme site d'intégration dans les génomes, ou encore au fait que certains phages peuvent posséder des ARNt dans leur propre génome (Bailly-Bechet et al., 2007). La figure 4.1 représente différentes caractéristiques retrouvées dans un îlot génomique. On y voit notamment l'association à un ARNt (en vert), la présence d'une intégrase (en rouge), de deux séquences d'insertions(en bleu), ainsi que quelques gènes (en jaune). Dans cette figure, l'ensemble de la région a un taux de GC uniforme, mais différent du reste du génome ; cela n'est pas toujours le cas.

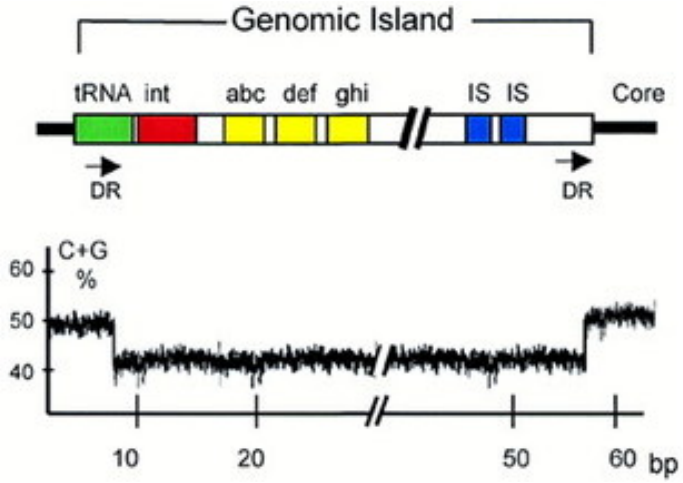


FIGURE 4.1 – Schéma d'un îlot génomique
Cette figure représente les différentes caractéristiques typiques des îlots génomiques. Copiée de (Hacker et al., 2001)

Globalement, les GI ne sont pas des régions homogènes : ils semblent être une mosaïque d'éléments initialement intégrés puis partiellement éliminés au fur et à mesure du temps (Lescat et al., 2009 ; Touchon et al., 2009). Ces régions sont très instables et rarement identiques d'un génome à l'autre, même entre des bactéries relativement proches. Cette composition morcelée complexifie la reconstruction de l'histoire des régions et leurs analyses fonctionnelles deviennent plus ardues. La proximité chromosomique des éléments qui composent ces régions n'est pas forcément indicatrice d'une quelconque proximité fonctionnelle, ni même d'une histoire évolutive commune, en tout cas à l'échelle d'un génome unique.

Les GI sont globalement retrouvés chez tous les types de bactéries : les bactéries pathogènes, qu'elles soient associées à des humains, animaux ou végétaux, mais également les bactéries symbiotiques, celles de l'environnement, ou bien celles présentes dans les microbiotes. De manière générale, toutes les bactéries qui ne sont

pas isolées dans des milieux sans flux de gènes vont être porteuses d'îlots génomiques.

Ces éléments sont aussi très mobiles (d'où une appellation commune : Mobile Genetic Elements, ou MGE) : à la fois à l'intérieur du génome, où certains éléments semblent capables de passer d'un site d'intégration à l'autre (Buchrieser et al., 1998), mais également entre les bactéries d'un même milieu, puisque certains GI sont mobilisables par transduction (Karaolis et al., 1999).

Hacker *et al.* ont introduit le concept de GI et suggèrent une classification basée sur la fonction adaptative de la région, illustrée notamment par la figure 4.2 extraite de leur publication (Hacker et al., 2001).

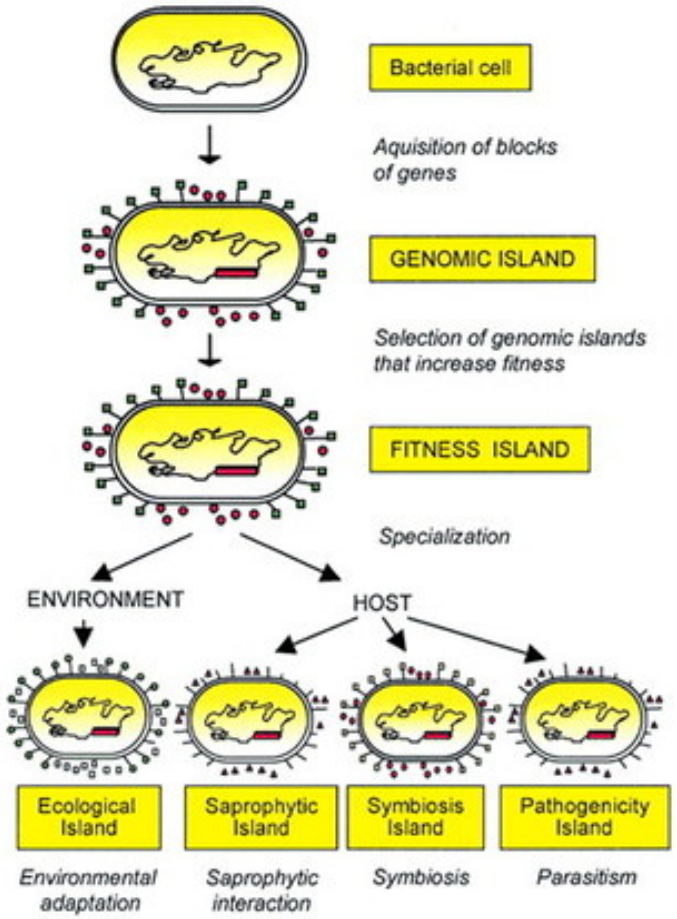


Figure 4.2 – Les différents types d'îlots génomiques

Cette figure représente les différents types d'îlots génomiques. Ils sont tous considérés comme des îlots d'adaptation (fitness en anglais) car ils apportent un avantage à la bactérie qui les possède tout en ayant des fonctions très différentes. Copiée de (Hacker et al., 2001)

Les îlots de "pathogénicité" (PAI) sont ainsi décrits comme des îlots génomiques ayant un ou plusieurs facteurs de virulence. Le concept de PAI a également été introduit par Hacker *et al.* dans un article soulignant que la délétion de deux larges régions d'ADN réduit la virulence de la souche pathogène *Escherichia coli* 536 chez des souris (Hacker et al., 1990). Ces régions incluent notamment des gènes responsables de la production d'hémolysine, qui cause la lyse des globules rouges, et de *fimbriae*, un type de pilus particulier qui permet aux bactéries de s'accrocher sur des surfaces ou des cellules.

Dans certains cas, des facteurs de virulence peuvent être retrouvés chez des bactéries non pathogènes, en lien avec leur survie et leur réplication, notamment dans certaines niches écologiques où ces bactéries n'auront pas d'activités pathogènes. Dans ces cas-là, on parlera alors d'îlots écologiques ou d'adaptation plutôt que d'îlots de pathogénicité ; d'où l'usage du terme plus général d'îlot génomique.

Éventuellement, certains éléments, devenus importants voire essentiels pour la bactérie, peuvent se condenser de plus en plus afin d'inclure uniquement les gènes nécessaires pour ladite fonction clé (Kaper et al., 1999). Un exemple bien étudié concerne un ancien îlot de *Salmonella enterica* : intégré initialement chez l'ancêtre commun à *S. enterica* et *S. bongori*, cet îlot s'est adapté à l'usage des codons et au taux de GC des deux organismes et fait désormais partie du *core* génome (Groisman et al., 1999). Cet îlot inclut des gènes de virulence, notamment pour intégrer les cellules de l'hôte lors de l'infection.

4.2 Méthodes de détection des îlots génomiques

L'intérêt grandissant des chercheurs pour les GI, qui semblent très importants dans l'histoire évolutive des bactéries, a conduit au développement de nombreuses méthodes visant à les identifier. Il existe deux grandes catégories d'approches, à savoir celles basées sur la composition et celles basées sur la génomique comparative. À présent, je vais sommairement expliciter leurs principes généraux en citant une liste non exhaustive d'outils ; concernant une liste exhaustive, il est possible de se référer à l'excellent article de Bertelli *et al.* (Bertelli et al., 2019).

4.2.1 Méthodes basées sur la composition

Les premières méthodes formalisées pour identifier les îlots génomiques utilisent la composition en nucléotides des génomes. En effet, il n'existait, à l'époque, que très peu de génomes auxquels se comparer et utiliser exclusivement la séquence du génome étudié présentait donc beaucoup d'avantages.

Il existe plusieurs manières de calculer un biais de composition (Karlin, 2001). Dans un premier temps, on peut travailler sur le taux de GC du génome : pour

cela, on va définir des blocs dans le génome. Ceux-ci peuvent être de taille fixe, ou alors défini grâce à des algorithmes conçus pour détecter des ruptures de signal, ici le taux de GC servant de signal (Lio et al., 2000). Ensuite, on compare le taux moyen de GC de chaque bloc par rapport au taux moyen de GC du génome, par exemple avec un test de χ^2 ou alors en séparant les blocs en fonction de la déviation standard.

Une autre approche utilise des signatures génomiques. Concrètement, une signature génomique peut correspondre aux abondances relatives des dinucléotides d'un génome, c'est-à-dire un rapport entre la fréquence d'un dinucléotide sur la fréquence de chacun des nucléotides qui le compose (Karlin, 1998). Ce sont concrètement des fréquences de 2-mers. Il est possible d'étendre un peu plus largement ce concept et de travailler sur des proportions de k-mers : par exemple, l'outil de prédiction AlienHunter utilise cette stratégie sur des k-mers de taille 8 (Vernikos et al., 2006).

Dans un troisième temps, le biais dans l'usage des codons est aussi une approche possible, car il s'agit d'une indication forte de l'acquisition des gènes par transfert horizontal (Médigue et al., 1991) ; cette méthode est, entre autres, implémentée dans l'une des déclinaisons de l'outil SIGI (Merkl, 2004 ; Waack et al., 2006), ou encore dans le plus récent Zisland Explorer (Wei et al., 2017).

Par ailleurs, plusieurs outils ont été développés en combinant certaines de ces méthodes plutôt que d'en utiliser qu'une seule. Ainsi, un logiciel comme IslandPath (W. Hsiao et al., 2003) va combiner taux de GC et biais en dinucléotides, sur une fenêtre glissante de 6 gènes, afin d'identifier les GI.

D'autres vont combiner le fait d'utiliser le biais de composition avec la présence d'ARNt ou de gènes de mobilité, qui sont souvent associés aux GI : c'est le cas d'Islander (Mantri et al., 2004) qui détecte les îlots potentiels associés à des ARNt, des ARNtm ou des intégrases. De plus, certains auteurs ont cherché à travailler sur l'ensemble de ces caractéristiques avec des méthodes d'apprentissage, à l'exemple du modèle élaboré par Vernikos *et al.* qui apprend les caractéristiques des GI décrits dans la littérature, comme les différentiels en composition ou les associations aux ARNt ou intégrases (Vernikos et al., 2008).

Parallèlement, certains auteurs ont étudié quelles étaient les métriques les plus discriminantes pour la détection de GI (W. W. L. Hsiao et al., 2005). Ces derniers se basent sur des GI identifiés et analysés, dans des articles précédemment publiés, afin de rechercher les facteurs les plus intéressants pour les prédire, en regardant lesquels pouvaient être prédits par chaque facteur. Cela a mené cette équipe à constituer deux jeux de données de référence qui ont ensuite été largement utilisés par les outils de détection de GI (Langille et al., 2008). L'un de ces deux jeux de données est constitué de génomes complets dont l'ensemble des GI a été expertisé, tandis que le second jeu de données a été construit grâce à une approche de

génomique comparative nommée IslandPick.

Finalement, il existe de très nombreux outils combinant différentes approches basées sur les biais de composition et des nouveaux sont régulièrement publiés encore aujourd'hui. Ces logiciels se différencient par leur méthode d'apprentissage, par les données sur lesquelles ils apprennent, ou encore par les éléments associés aux GI qu'ils vont utiliser. Parmi les exemples récents, on peut notamment citer ZislandExplorer (Wei et al., 2017) ou encore la méthode non nommée de Onesime *et al.* (Onesime et al., 2021). Précédemment, j'ai mentionné les ARNt, les ARNtm ou les intégrases, mais certains utilisent aussi les familles de gènes régulièrement associées à des GI afin de guider la détection ; c'est le cas de l'outil de prédiction IslandCafe (Jani et al., 2019).

4.2.2 Méthodes basées sur la génomique comparative

Les approches dites de "génomique comparative" concernent toutes les méthodes qui utilisent, de manière générale, la comparaison de plusieurs génomes. Elles se différencient par la manière dont les génomes sont comparés, et notamment par le nombre de génomes considérés et quelles informations sont utilisées lors de la comparaison.

4.2.2.1 MOSAIC

Les premières approches se sont basées sur les alignements de génomes, comme MOSAIC (Chiapello et al., 2005 ; 2008), dont le but était d'étudier la segmentation des génomes microbiens pour toutes les souches disponibles de plusieurs espèces. Cela inclut notamment les îlots génomiques car ce sont des éléments souvent uniques à l'échelle de l'espèce lorsqu'on considère moins d'une dizaine de génomes. MOSAIC cherche à déterminer le "backbone" et les "loops", ce qui correspond au *core* et à l'accessoire par définition, mais au niveau des séquences des génomes plutôt que des familles de gènes. Graphiquement, cela forme un cœur de séquences présentes dans tous les génomes, depuis lequel ressortent des boucles correspondant à des portions de séquences présentes uniquement dans certains génomes. Le problème de ce type d'approche, qui est d'ailleurs souligné dans l'article, est que plus il y a de génomes, plus le backbone est petit et plus les loops sont nombreuses. Les auteurs pensent que le backbone finira par être stable lorsqu'un nombre suffisant de génomes seront ajoutés, représentant alors le génome "minimal" signature de l'espèce étudiée. On sait aujourd'hui que ça n'est pas le cas, surtout avec les assemblages des génomes qu'on manipule : en effet, le *core* (l'équivalent pangénomique du backbone) décroit mécaniquement au fur et à mesure que l'on ajoute des génomes (Lukjancenko et al., 2010).

Dans le contexte de la détection d'îlots génomiques, la méthode MOSAIC identifie probablement des îlots, mais va aussi identifier toutes les variations qui ne sont pas des îlots génomiques, comme les délétions ponctuelles ou les insertions de portions d'ADN. Il est aussi très probable que cette approche ne puisse pas aller au-delà de quelques dizaines de souches.

4.2.2.2 tRNAcc / tRIP / mobilomeFINDER

Les auteurs de tRNAcc (Ou et al., 2006) utilisent le même vocabulaire que MOSAIC pour parler du *core* génome, et présument que les transferts horizontaux de gènes (HGT) permettent l'intégration d'îlots génomiques au sein de ce backbone, et que ces intégrations se font dans des points chauds d'intégration (hotspots) dont les plus communs sont des sites d'ARNt et d'ARNtm.

Les auteurs se basent sur MAUVE (A. C. Darling et al., 2004) pour réaliser des comparaisons entre les génomes. Dans leur étude, ils considèrent qu'un GI est un segment anormal situé entre le 3' d'un ARNt et le 5' de la région flanquante dont la taille maximale a été définie à 250kb. Les ARNt avec un potentiel GI sont dits "occupés" alors que les ARNt sans GI sont dits "vides". Leur approche informatique est suivie par une grande part de curation, notamment la suppression des GI trop petits et la fusion des GI assignés plusieurs fois à différents ARNt parce que ceux-ci sont colocalisés ; les auteurs choisissent alors d'assigner de tels GI à un unique ARNt.

L'approche offre l'avantage de se concentrer sur des zones où les GI sont ordinairement retrouvés, ce qui limite la quantité de faux positifs par rapport à ce que MOSAIC pourrait produire. Cependant, elle présente l'inconvénient de ne détecter que les GI proches des ARNt et ARNtm, et de nécessiter l'intervention humaine pour nettoyer les résultats. Un autre défaut important, souligné par les auteurs, est que le nombre de comparaisons à réaliser devient trop important au-delà de quelques génomes (le nombre de 5 est donné dans l'article) puisque leur approche se base sur des comparaisons multiples. Pour pallier ce problème, les auteurs suggèrent de ne comparer que quelques-uns des génomes.

Une deuxième partie de leur article propose de tester si certaines régions d'un génome contenant des GI contiennent aussi des GI dans d'autres génomes. Les auteurs définissent les régions par deux séquences de 2kb, avant et après le GI : ces deux régions sont typiquement des séquences flanquantes du GI qui vont définir un site d'insertion (spot) dans une ou plusieurs espèces. Avec leur outil tRIP, les auteurs réalisent ce qu'ils appellent une e-PCR, ou PCR *in silico*. L'objectif est d'identifier la taille de la région entre les deux "amorces". Si la taille de la région ainsi identifiée dans un autre génome est supérieure à 1kb, ils la définissent comme "occupée" et considèrent que son contenu est aussi un GI. Les auteurs de tRNAcc et tRIP ont couplé leurs outils avec d'autres outils d'analyses pour les calculs de

biais de composition et d'annotation fonctionnelle dans un serveur web appelé mobilomeFINDER (Ou et al., 2007).

4.2.2.3 IslandPick/IslandViewer

IslandPick est une approche originale dans sa publication car elle a été construite initialement pour valider les prédictions faites par les méthodes basées sur les biais de composition grâce à une approche de génomique comparative (Langille et al., 2008). Un problème jusqu'alors assez récurrent dans les approches de génomique comparative est le changement d'échelle : dès qu'il y a une dizaine de génomes ou plus, les outils deviennent beaucoup trop lents à cause de la quantité de comparaisons à réaliser. Une alternative est tout simplement de renoncer à inclure tous les génomes.

IslandPick propose une approche automatique et flexible de sélection de génomes, en travaillant avec une fonction de distance génomique entre les génomes. Cela permet de sélectionner des génomes d'intérêt, parmi les plus appropriés, afin d'y détecter des GI. Pour IslandPick, un ensemble de génomes appropriés doit contenir entre six et douze génomes, doit inclure un outgroup, à savoir un ou plusieurs génomes qui ne sont pas de l'espèce étudiée mais d'une espèce proche, et plusieurs de ces génomes doivent avoir une distance minimale par rapport au génome étudié. Ensuite, l'outil réalise des alignements entre les génomes sélectionnés et le génome étudié, à l'aide de l'algorithme MAUVE (A. C. Darling et al., 2004), et identifie les régions non alignées et les régions conservées ; les régions non alignées correspondent alors à des GI potentiels.

Dans leur article, les auteurs utilisent cette approche pour définir un benchmark d'outils utilisant le biais de composition : les régions conservées permettent de former un ensemble de régions "négatives" dans leur jeu de données, c'est-à-dire des régions "non GI" (par opposition aux GI qui sont des régions "positives"). IslandPick est intégré dans IslandViewer, un serveur web regroupant un visualisateur de génomes ainsi que différents outils de prédiction utilisant les biais de composition, comme AlienHunter, SIGI et Islander (Langille et al., 2009). IslandViewer a connu de nombreuses mises à jour et améliorations, et est toujours très activement utilisé (Bertelli et al., 2017 ; Dhillon et al., 2013 ; 2015).

4.2.2.4 xenoGI

L'importance des îlots génomiques dans l'évolution des génomes des espèces microbiennes est avérée, néanmoins aucun outil de reconstruction ne permettait de placer les GI identifiés dans un contexte phylogénétique. xenoGI (Bush et al., 2018) est la première méthode qui permet de combiner détection des GI et contexte phylogénétique. xenoGI permet aussi d'aller un peu plus loin dans la définition

d'îlots génomiques puisqu'il peut identifier des GI présents dans plusieurs souches et repérer dans quelle branche d'un arbre phylogénétique le transfert de chaque groupe de gènes a eu lieu, ce qui peut permettre de comprendre le chemin adaptatif suivi par les souches étudiées.

xenoGI commence par calculer un score basé sur la similarité et la conservation de la synténie entre chaque gène des différents génomes. Ensuite, il reconstruit des familles de gènes orthologues en prenant en compte l'arbre phylogénétique. Finalement, il groupe ces familles en îlots d'éléments acquis en même temps, et donc présumés d'avoir une origine commune. Les éléments du *core* génome transférés verticalement sont ainsi acquis à la racine de l'arbre phylogénétique, et les autres sont potentiellement acquis par transfert horizontal à différents nœuds de l'arbre et vont ainsi former les GI prédits par l'outil.

xenoGI permet de générer plus de comparaisons que les outils précédemment mentionnés, et peut analyser jusqu'à 50 génomes de la même espèce ou d'espèces proches.

4.2.3 Faiblesses des approches de détections d'îlots génomiques

Malgré le fait que deux types d'approches utilisent des informations très différentes, chacune des deux catégories précédemment développées pose plusieurs problèmes.

Les approches basées sur les biais de composition sont limitées par le fait que les gènes acquis par transfert horizontaux ne sont pas les seuls à avoir des biais dans leur composition : par exemple, c'est aussi le cas des gènes en interaction avec la membrane bactérienne, comme les transporteurs ou des protéines de surface. Une autre problématique est que la composition d'un groupe de gènes acquis peut être proche de celle du *core* génome s'il provient, par exemple, d'un génome d'une espèce phylogénétiquement apparentée. Dans ce cas, ces groupes de gènes transférés ne seront pas détectés par ces méthodes. De plus, les acquisitions qui ne sont pas récentes vont être difficiles à détecter avec des approches basées sur les biais de composition, car l'usage du code des gènes va petit à petit s'adapter à celui de la bactérie. C'est le cas notamment de l'ancien îlot de *Salmonella enterica* que j'ai mentionné précédemment dans la section 4.1 (Groisman et al., 1999). Aujourd'hui, les gènes qui composent cet îlot chez *S. enterica* sont totalement adaptés à la composition en nucléotides de la bactérie et aucun outil utilisant la composition n'est capable de l'identifier. Seule une approche de génomique comparative utilisant des génomes de plusieurs espèces peut permettre de détecter que cette région a été acquise par transfert horizontal.

Les approches de génomique comparée ne sont pas non plus exemptes de dé-

fauts. Aucune à ma connaissance n'est capable d'analyser plus de quelques dizaines de génomes dans une même analyse, à une époque où des milliers de génomes sont disponibles pour certaines espèces, et où la quantité de données augmente toujours plus vite. Cela impose à l'utilisateur de ces méthodes de limiter son analyse à un ensemble de génomes sélectionnés, pouvant biaiser le résultat final si celui-ci n'est pas représentatif. Seul IslandViewer (Bertelli et al., 2017) propose une approche pour sélectionner automatiquement des génomes, mais leur analyse se fait nécessairement génome par génome et supporte un maximum de 12 génomes pour la détection de GI. Un second point important : la nature des éléments détectés dépend aussi fortement du niveau taxonomique considéré. En effet, utiliser uniquement des génomes de la même espèce ou utiliser des génomes qui appartiennent au même genre, mais à des espèces différentes, ne permettra pas d'identifier le même type d'éléments. La définition d'îlots génomiques va être généralement restreinte aux éléments acquis très récemment dans l'évolution des bactéries étudiées. Cependant, on peut également être intéressé par les acquisitions par transfert horizontal qui ont précédé la spéciation d'une espèce qu'on étudie, à savoir les "anciens" îlots qui sont désormais fixés dans le *core* génome de l'espèce. Le cas échéant, la diversité et la quantité de génomes, nécessaires pour réaliser l'analyse, deviennent limitantes et les méthodes actuelles de génomique comparative sont vite dépassées.

4.3 Points chauds d'insertion

4.3.1 Définition et détection

Les points chauds (hotspots) d'insertion correspondent à des zones des génomes dans lesquelles de nombreux éléments vont s'insérer au cours de l'évolution d'un groupe d'organismes. Cela correspond donc à une zone du génome marquée d'une quantité beaucoup plus importante de changements dans le répertoire génique que le reste du génome.

Là où un îlot génomique est un concept "génome centré", le concept de hotspot se place nécessairement à l'échelle d'un groupe taxonomique, et il faudra des analyses de génomique comparative pour l'identifier.

Malgré l'intérêt général pour les îlots génomiques, il existe peu d'articles dédiés à la détection ou à l'analyse de hotspot d'insertion, sans doute car la limitation en nombre de génomes des approches de génomique comparative rend la tâche particulièrement ardue. Analyser des îlots génomiques différents mais situés dans le même contexte génomique complexifie également leur détection. On ne peut pas se baser sur les îlots génomiques eux-mêmes, et il est nécessaire de se rapporter aux séquences flanquantes des îlots.

Ainsi, les analyses de hotspots d'insertion vont au début se baser principalement

sur des analyses qui n'utilisent pas d'approche globale. A titre d'exemple, Lescat *et al.* ont analysé un hotspot proche d'un tRNA *leuX* dans 12 génomes d'*Escherichia coli*, et dont les séquences flanquantes, c'est-à-dire les gènes du *core* qui sont autour des îlots génomiques concernés, sont les mêmes (Lescat et al., 2009). Les auteurs identifient dans ce hotspot des blocs de gènes conservés ensemble, qu'ils appellent modules, et impliqués potentiellement dans les mêmes fonctions, notamment un module de 23 kb qui inclut plusieurs gènes impliqués dans la résistance aux antibiotiques.

Un autre exemple est donné par Touchon *et al.* avec un hotspot intégré au niveau d'un tRNA pheV, avec des séquences flanquantes encore une fois identiques (Touchon et al., 2009). Les auteurs soulignent que la distribution très hétérogène des gènes dans les régions au travers des génomes peut être un signe de multiples événements d'intégrations, ou alors de recombinaisons très fréquentes entre des éléments intégratifs.

A ma connaissance, il existe 3 méthodes bioinformatiques pour la détection de hotspots d'insertion. Précédemment mentionné, tRIP (Ou et al., 2006) est centré sur un hotspot donné et son but sera justement de déterminer les îlots génomiques évoluant dans le même contexte génomique qu'un îlot d'intérêt défini, en utilisant ses séquences flanquantes.

Le Genome Complexity Browser (Manolov et al., 2020) est génome-centré, il va chercher à déterminer les hotspots en mesurant la complexité des chemins possibles dans un pangénome rapportée à l'échelle d'un génome.

L'approche non nommée de Oliveira *et al.* (Oliveira et al., 2017) repose, quant à elle, sur une analyse de génomique comparée globale basée sur les gènes du *core* génome. La figure 4.3 illustre leur approche appliquée sur plusieurs génomes. Elle nécessite l'usage d'un génome pivot par espèce et une définition très stricte du *core* génome. L'approche va identifier tous les spots d'un pangénome comme étant des enchaînements de gènes du *core* génome pouvant avoir des gènes accessoires entre eux.

Ils vont ensuite définir des hotspots en fonction du nombre de gènes accessoires dans le spot étudié. À ma connaissance, il s'agit de la seule approche permettant une analyse globale des (hot)spots à l'échelle d'un pangénome.

4.3.2 Caractérisation globale des points chauds d'insertion

L'approche de Oliveira *et al.* (Oliveira et al., 2017) étant la première permettant une analyse globale, les auteurs ont cherché à identifier les gènes ou fonctions associées, et à caractériser plus en détails les points chauds, aussi appelés hotspots d'insertion. Ils définissent un spot comme étant un intervalle entre deux gènes du *core* présents successivement sur un génome pivot. Ils définissent ensuite les spots des autres génomes de l'espèce par rapport à ce génome pivot. Ils mesurent que

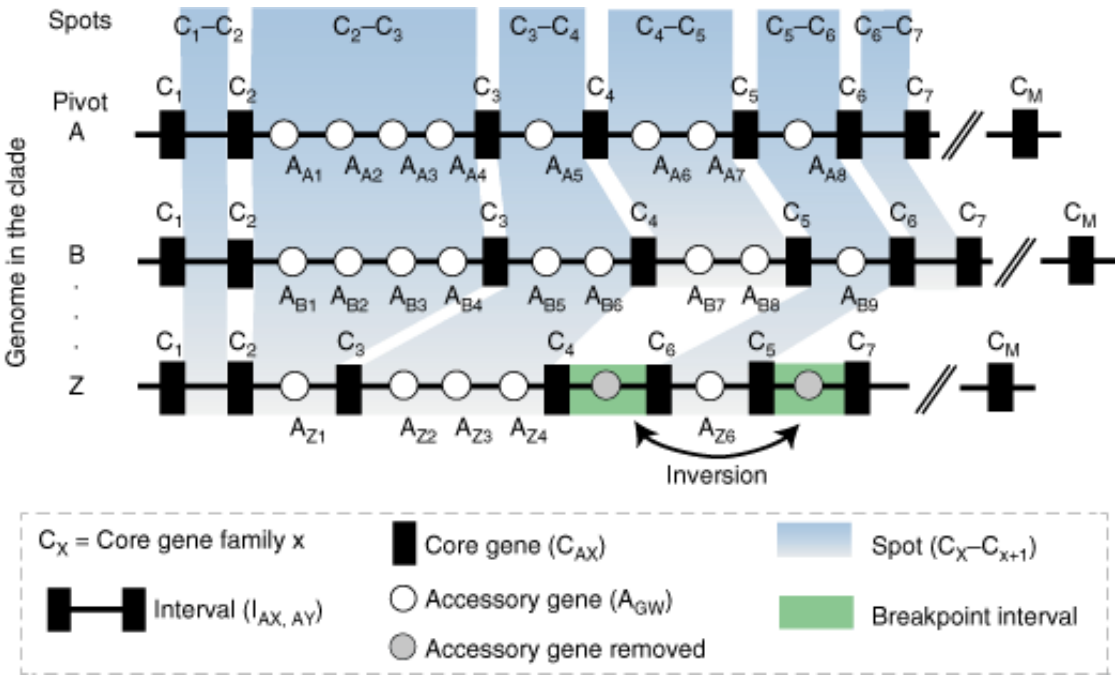


FIGURE 4.3 – Méthode de détection de spots
Cette figure illustre l'approche de détection de spots reposant sur l'identification d'intervalles entre des gènes du *core* génome. Copiée de (OLIVEIRA et al., 2017).

moins de 2 % de l'ensemble des spots sont nécessaires pour inclure 50 % des gènes acquis par transfert horizontal. Les auteurs rapportent par la même occasion que 72.6 % des spots définis sont toujours vides, c'est-à-dire qu'il n'y a aucun gène entre deux gènes du *core*.

A l'aide d'une simulation, les hotspots peuvent être différenciés des spots sur la base de la distribution du nombre de gènes insérés. La simulation consiste à insérer les gènes dans des spots de manière équiprobable, puis le nombre de gènes dans chaque spot est rapporté. La simulation est réalisée plusieurs fois et le 95ième percentile de la distribution ainsi obtenue est choisi pour seuil. Les auteurs définissent ainsi que 1.2 % des spots sont des hotspots. Par la suite, les auteurs ont concentré leurs analyses sur les hotspots, et sur les gènes qu'ils incluent par rapport au reste du génome.

La caractérisation fonctionnelle des gènes retrouvés dans les hotspots permet d'indiquer qu'ils sont enrichis en gènes impliqués dans la motilité, dans des mécanismes de défense, ainsi que dans la transcription, la réplication et la réparation de l'ADN. Ce sont des fonctions souvent retrouvées lors d'analyses différentielles qui vont rechercher les fonctions enrichies dans les gènes accessoires d'un pangénome (FLORES RAMOS et al., 2021 ; ZHU et al., 2019).

De nombreux hotspots incluent aussi des gènes de résistance aux antibiotiques. Environ 90 % des éléments génétiques mobiles (prophages et ICE) ont été trouvés dans des hotspots, mais ils ne représentent qu'un nombre très faible en proportion (23 % et 9 %, respectivement). Le fait que la majorité des régions n'incluent pas d'éléments génétiques mobiles laisse suggérer que la genèse et le changement en contenu génique de ces régions ne sont pas forcément médiés par ces éléments, autrement dit, pas médiés par la transduction ni par la conjugaison.

L'origine des hotspots ne semblent pas non plus être liée à l'intégration d'un unique ensemble de gènes : seuls 8 % des hotspots sont composés de gènes présents uniquement dans une seule souche. Cela indique que les hotspots sont *a priori* plutôt des agglomérats de gènes acquis au cours de plusieurs évènements plutôt qu'issus d'une unique acquisition, ce qui était l'une des hypothèses posées dans l'article de Touchon *et al.* (Touchon et al., 2009) et qui expliquerait peut-être l'organisation modulaire observée à plusieurs reprises dans des îlots ou hotspots (Lescat et al., 2009 ; Ogier et al., 2010 ; Touchon et al., 2009).

Les auteurs ont ensuite testé et mesuré que les gènes flanquants les hotspots étaient plus sujets à la recombinaison homologue. Ils ont identifié 50 % d'évènements de recombinaison dans ces gènes flanquants en comparaison des autres gènes du *core* génome, et 30 % de gènes en plus ayant une incongruence phylogénétique en comparaison de l'arbre des espèces construit grâce à tous les gènes du *core* génome.

Les éléments génétiques mobiles et ICE étant peu présents dans les hotspots, une hypothèse possible est, de manière générale, que la transformation naturelle est un vecteur important dans l'acquisition de nouveaux éléments. Pour tester cela, les auteurs ont comparé les métriques entre les génomes de bactéries étant identifiées comme naturellement compétentes (c'est-à-dire pouvant réaliser la transformation naturelle) et les autres. Ils ont montré que ces espèces avaient plus de hotspots que les autres, avec moins de hotspots incluant des protéines associées à la mobilité des éléments génétiques, et finalement que la recombinaison était 20 % plus fréquente dans les gènes flanquants que dans le reste du génome. Cela indiquerait potentiellement que la recombinaison homologue au niveau des gènes flanquants est une source importante d'intégration de nouveaux éléments génétiques dans le génome.

Chapitre 5

Modules en génomique

Ce chapitre introduit les usages et les méthodes liés à la modularité en génomique. J'y introduis le concept de module, ainsi que son utilité pour améliorer notre compréhension des génomes procaryotes. Ensuite, je détaillerai les différentes sources de modularité, ainsi que les méthodes qui permettent d'identifier des modules.

5.1 Annotation par association

5.1.1 Concept

Un pan entier de la bioinformatique a pour but de prédire les fonctions des organismes que l'on souhaite étudier via leur génome. Nombre de ces prédictions sont réalisées grâce à l'analyse des gènes codant les protéines d'un organisme. Néanmoins, la caractérisation fonctionnelle d'un gène ou d'une famille de gènes est loin d'être simple, et nécessite souvent le travail héroïque de nombreuses spécialités incluant de manière non exhaustive la bioinformatique, la biologie moléculaire, la biochimie et la microbiologie. Bien souvent, même s'il est possible d'identifier des partenaires d'interaction, ou des fonctions générales, les informations ne resteront que parcellaires sauf pour des gènes particulièrement étudiés, par exemple, ceux impliqués dans le métabolisme central ou d'importance dans le domaine médical ou l'industrie. Cela représente un coût humain et financier très important pour des informations qui, rapportées à l'échelle d'un organisme entier, ne représentent qu'une partie infime de ses capacités.

Ainsi, identifier à moindre coût les fonctions et capacités potentielles d'un gène est un objectif particulièrement séduisant pour qui en a la possibilité. Il est, dans une certaine mesure, possible de réaliser cette identification au travers de ce qu'on appelle de manière très générique l'annotation fonctionnelle. L'un de ses aspects,

déjà mentionné dans la section 3.3, est de comparer les séquences d'un génome à des séquences de gènes dont la fonction est connue, soit en alignant des bases de données de gènes expérimentalement caractérisés, ou en reconstruisant des familles de gènes homologues. L'hypothèse sous-jacente est que les gènes d'une même famille sont impliqués dans les mêmes types de fonction, notamment les gènes orthologues qui vont avoir tendance à conserver la même fonction. Ces approches sont très utilisées, mais n'ont un intérêt que si beaucoup de gènes dans le génome ressemblent à des gènes qui ont déjà été étudiés. Le nombre d'organismes étudiés en laboratoire, et dont les gènes sont caractérisés est extrêmement faible comparé à la diversité du vivant. De plus, ceux qui sont bien caractérisés sont plutôt anthropocentrés, puisque l'écrasante majorité d'entre eux sont ceux qui peuvent avoir un impact direct sur la vie humaine : ce sont soit des organismes modèles (*e.g. E. coli* ou *Bacillus subtilis*), des pathogènes, soit des souches impliquées dans des processus industriels.

Il y a un biais d'échantillonnage important dans ce qui est connu, et dès que le génome à étudier sera très distant phylogénétiquement d'une bactérie modèle, on ne pourra pas obtenir beaucoup d'information utilisable. Chez les génomes d'organismes non-modèles, notamment ceux encore non cultivés en laboratoire et issus d'analyses métagénomiques, bien souvent seules les fonctions des gènes ubiquitaires chez les procaryotes seront identifiables et la plupart de celles qui font la spécificité de l'organisme demeureront inconnues.

Pour tenter de passer outre ces limitations dans nos connaissances, un autre type d'approche bioinformatique consiste à utiliser le contexte dans lequel les gènes sont observés. Effectivement, comme on a pu le voir dans la section 2.1.4, les gènes impliqués dans les mêmes fonctions ont tendance à se retrouver dans les mêmes contextes.

Il y a plusieurs manières d'exprimer un même contexte en génomique. Cela peut être le fait de retrouver ces gènes ensemble dans les mêmes organismes, on parlera alors de gènes ayant le même profil phylogénétique. Ce sera aussi des gènes retrouvés ensemble et aux mêmes endroits dans différents génomes, on parlera alors de synténie conservée. Lorsque c'est possible, on peut aussi étudier les gènes dans le contexte de la régulation de leur transcription, et le cas échéant on pourra regrouper les gènes sur le fait qu'ils sont exprimés ensemble dans un même opéron ou non. Une autre source d'information contextuelle est l'étude des fusions/fissions. Parfois, les gènes de protéines qui interagissent peuvent être fusionnés en un unique gène dans certains organismes.

Ces approches de contexte peuvent permettre d'étudier des interactions entre des paires de gènes, mais également d'identifier des ensembles de gènes qui interagissent dans un même processus. Lorsqu'ils sont regroupés en groupes distincts, les gènes ainsi identifiés sont appelés modules.

5.1.2 Modules et modularité

La notion de modules en biologie est large et ne concerne pas que la génomique, cela correspond à la division de structures biologiques en parties standardisées (Winther, 2001). Cela peut concerner le développement, la physiologie, le métabolisme, des complexes protéiques comme des ensembles cellulaires chez les eucaryotes multicellulaires. Ce qui va permettre l'appellation de "module" est la notion d'unicité et de non-recouvrement. Les modules peuvent être répétés et sont souvent conservés au cours de l'évolution dans certains taxons, voire entre des taxons très différents (dans le contexte de fonctions essentielles à la vie ou d'un transfert horizontal, par exemple). Dans des contextes différents, les modules peuvent avoir des évolutions (et donc des compositions) différentes, mais la fonction basique que le module fournit reste la même, et il est le seul à la fournir.

Dans notre cas, on va s'intéresser à deux types de modules, et uniquement à l'échelle du génome : les modules fonctionnels et les modules conservés (Snel et al., 2004). Les deux concepts sont associés mais ne sont en aucun cas identiques. Un module fonctionnel va correspondre à un groupe de gènes qui va permettre une fonction dans un contexte donné, comme une voie métabolique ou la formation d'un complexe protéique pour, par exemple, un système de sécrétion. Un module conservé va correspondre à un ensemble de gènes qui sont gagnés et perdus ensemble au cours de l'évolution dans plusieurs taxons. Les deux concepts peuvent être liés, car un module fonctionnel peut être conservé au cours de l'évolution, et ainsi former un module conservé. La composition dudit module fonctionnel peut varier et inclure des gènes différents qui vont contextuellement répondre à des problématiques différentes pour finalement fournir la même fonction. Le cas échéant, le module conservé ne sera formé que des gènes qui ont été conservés au sein du module fonctionnel dans tous les contextes où il est retrouvé, s'il y en a.

5.2 Identification de modules

Il existe de nombreuses méthodes et approches pour constituer des modules. Beaucoup de ces approches sont des méthodes automatiques qui vont utiliser une source d'information pour l'association entre les gènes et créer ainsi des réseaux dans lesquels des clusters seront calculés. D'autres approches sont des bases de données de modules expertisés, extraits de la littérature scientifique et mis à disposition au travers de bases de données publiques. Ces bases de données sont d'ailleurs souvent utilisées comme référence pour évaluer les méthodes automatiques de reconstruction de modules.

5.2.1 Modules fonctionnels

5.2.1.1 Réseaux d'interactions protéines-protéines

Les réseaux d'interactions protéines-protéines (PPI) sont des réseaux globaux de relations généralement obtenues expérimentalement au travers, par exemple, d'expériences de double-hybride (Fromont-Racine et al., 1997) qui peuvent être effectuées sur une large partie de toutes les protéines d'un organisme. La première expérience de double hybride à large échelle sur une bactérie a été faite sur 261 protéines de *Helicobacter pylori* (Rain et al., 2001). D'autres types d'expériences sont réalisables pour étudier les interactions protéines-protéines, notamment la spectrométrie de masse avec purification par affinité, ou la co-immunoprécipitation, qui sont les sources de PPI les plus communes au sein du jeu de données IMEx (Orchard et al., 2012) qui compile les interactions de plusieurs bases de données expérimentales, comme MIntAct (Orchard et al., 2014) ou DIP (Salwinski et al., 2004). La figure 5.1 illustre, en proportion, les méthodes utilisées pour caractériser des interactions, les types d'interactions et les espèces dans lesquels les expériences ont été réalisées.

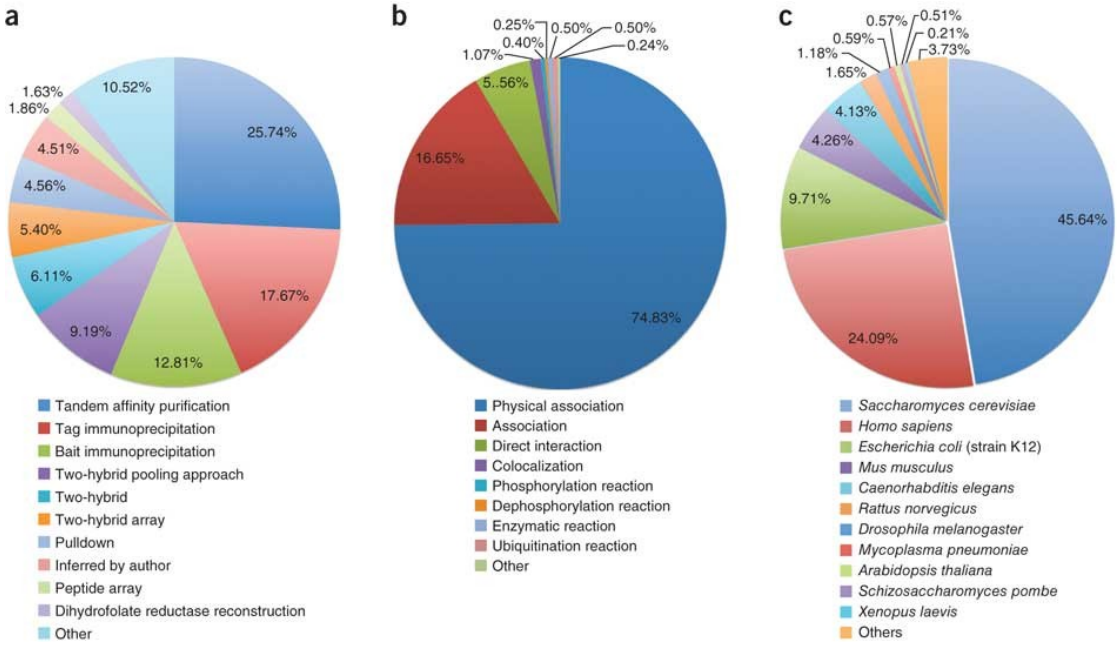


Figure 5.1 – PPI : Méthodes, types d'expériences et les espèces dans lesquelles celles-ci sont réalisées
Figure illustrant les expériences biologiques qui caractérisent les interactions, le type d'interactions détectées et les espèces dans lesquelles celles-ci sont détectées. Copié de Orchard et al., 2012.

Le résultat d'analyses globales de PPI sur un organisme donné est un graphe pondéré où chaque nœud est une protéine et chaque arête est une interaction pondérée par un score indiquant la probabilité ou la force de l'interaction. On peut ainsi, pour chaque protéine étudiée, regarder avec quelles protéines celle-ci peut interagir. A plus large échelle, en appliquant des algorithmes de clustering, on peut découper le graphe en modules fonctionnels en regroupant les protéines qui interagissent ensemble dans des clusters. Il existe de nombreux algorithmes pour réaliser cela. La plupart utilisent des algorithmes très classiques du clustering de graphe. Un exemple possible est un algorithme qui utilise la centralité (J. Chen et al., 2006). Un autre exemple se rapporte au problème de l'arbre de Steiner dans un graphe connexe (Dittrich et al., 2008), qui consiste à identifier un ensemble d'arêtes de poids minimal (dans ce cas, le poids de zéro représente le meilleur score possible) tel que le sous-graphe induit soit connexe et contiennent tous les sommets du graphe. Cette approche permet d'identifier le cheminement des interactions (les arêtes de l'arbre) le plus probable dans ce graphe. L'usage de ce problème informatique en particulier implique des interactions deux à deux uniquement, or certaines protéines peuvent avoir plus d'un partenaire d'interaction.

Certains algorithmes ont pour objectif d'identifier des modules conservés dans les réseaux de PPI. Ils combinent des réseaux de plusieurs organismes placés dans un arbre phylogénétique pour effectuer une prédiction de PPI ancestrales (Dutkowski et al., 2007). Cela permet d'identifier exclusivement les modules fonctionnels qui sont des modules conservés à l'échelle de l'évolution.

Ces méthodes sont malheureusement limitées, car nécessitant des expériences qui peuvent être complexes. On retrouve donc des réseaux de PPI, et par extension des modules fonctionnels uniquement pour quelques organismes très étudiés. Les trois quarts des interactions dans IMEx concernent *Saccharomyces cerevisiae* et *Homo sapiens*, et 10 % *Escherichia coli*, comme illustré dans la figure 5.1.

De plus, toutes les protéines ne sont pas manipulables par l'ensemble des méthodes expérimentales. En double hybride, par exemple, les protéines doivent pouvoir réaliser leurs interactions malgré la modification que l'expérience implique. Un réseau de PPI ne pourra donc généralement pas permettre de retrouver l'ensemble des modules fonctionnels.

5.2.1.2 Fusion/Fission

Les protéines interagissant ensemble pour fournir une même fonction forment un module fonctionnel. Néanmoins, les protéines sont modulaires en soi : une protéine est composée d'un ou plusieurs polypeptides composés d'un ou plusieurs domaines qui vont fournir une ou plusieurs fonctions. Il arrive donc que des gènes fusionnent au cours de l'évolution, codant alors un unique polypeptide portant l'ensemble des domaines de la protéine d'origine, et celui-ci va *a priori* fournir

les mêmes fonctions. L'évolution inverse peut aussi se produire où un gène codant plusieurs domaines d'une protéine peut être fissionné pour donner deux gènes. Généralement, les domaines ainsi regroupés sont impliqués dans une fonction biologique commune (Marcotte et al., 1999), ce qui permet d'identifier que les protéines d'origine étaient impliquées dans un même module fonctionnel (Yanai et al., 2001). Une étude à plus large échelle sur les fusions et fissions de gènes a permis d'identifier néanmoins qu'il y a globalement peu d'événements de ce type (Snel et al., 2000a).

Un exemple d'approche récente pour la détection de gènes fusionnés est la méthode CompositeSearch (Pathmanathan et al., 2018). CompositeSearch utilise un graphe de similarité, où les nœuds sont des gènes et les arêtes des résultats d'alignements. Il réalise un pré-calcul des familles de gènes, ou chacun des gènes sera assigné à une famille au préalable. Il détermine les gènes fusionnés comme étant les gènes qui ont au moins deux voisins dans le graphe de similarité qui sont dans des familles de gènes différentes. Les familles de ces gènes doivent avoir un nombre suffisant de membres, nombre qui est paramétrable.

L'information de fusion dans un organisme est une information très fiable pour inférer une interaction à partir de données génomiques uniquement, mais malheureusement très parcimonieuse. En effet, assez peu de gènes sont concernés et elle n'est pas suffisante pour reconstruire l'ensemble des modules fonctionnels d'un organisme donné, même si ce type d'information peut être très intéressant lorsque couplé à d'autres sources permettant d'inférer la modularité fonctionnelle.

5.2.1.3 Opérons

Les gènes regroupés dans un même opéron sont généralement impliqués dans la même fonction. Le regroupement en opéron facilite l'expression simultanée des gènes et leur action commune pour fournir une fonction, pour la formation de complexe protéique notamment (Shieh et al., 2015). Cela permet naturellement de regrouper les gènes en modules fonctionnels sur la base de leur co-transcription dans un opéron. Chez une bactérie, il est assez aisé d'identifier tous les opérons grâce à une analyse de séquençage d'ARN (RNA-seq) (Yu et al., 2018). Les gènes en opéron sont alors tout simplement ceux qui sont sur le même transcrit. Il est aussi possible de prédire des opérons sur la base de caractéristiques génomiques sans avoir à réaliser une expérience de transcriptomique, notamment en se basant sur la distance génomique entre les gènes, le fait qu'ils soient sur le même brin et potentiellement régulés par les mêmes facteurs (Westover et al., 2005).

Il existe plusieurs bases de données de référence d'opérons. RegulonDB (Santos-Zavaleta et al., 2019) pour *Escherichia coli* K-12 notamment indique, parmi une foultitude d'autres informations, les structures opéroniques de la bactérie et est une source particulièrement fiable. Il existe aussi d'autres bases de données plus

généralistes d'opérons qui vont inclure plusieurs organismes, par exemple DOOR qui compile des prédictions d'opérons validées expérimentalement ou basé sur des données RNA-seq pour des milliers de génomes (Mao et al., 2014).

Il est néanmoins important de noter que des gènes peuvent être associés dans leur fonction, et donc dans le même module fonctionnel, sans être dans le même opéron (Lathe III et al., 2000). Un exemple de cette observation est un ensemble de gènes dédié à la formation de flagelles qui sont présents dans différents génomes, mais pas toujours organisés de la même manière, illustré dans la figure 5.2.

L'information sur l'organisation en opéron du génome est donc importante, mais non suffisante pour inférer l'ensemble des modules fonctionnels d'un organisme.

5.2.1.4 Métabolisme

Le métabolisme concerne les réactions chimiques qui vont permettre de former, transformer ou dégrader des composés. Ces réactions sont produites par des enzymes qui sont généralement des protéines. On va parler de voies métaboliques lorsqu'on parle d'un ensemble de réactions qui partent d'un composé particulier et qui le transforme en un autre composé qui sera le produit final. L'ensemble des gènes qui produisent les enzymes qui permettent cet ensemble de réactions forme ainsi un module fonctionnel. On peut donc se servir des bases de données de voies métaboliques comme KEGG (Kanehisa et al., 2021) ou MetaCyc (Caspi et al., 2020) pour récupérer ou identifier des modules fonctionnels connus impliqués dans le métabolisme. Néanmoins, la granularité à laquelle les bases de données vont considérer ce qu'est une voie métabolique va varier. À l'échelle de KEGG, les voies métaboliques sont décrites dans des cartes (maps) qui sont des ensembles assez larges de réactions qui peuvent mener à la biosynthèse ou à la dégradation de composés, incluant les chemins 'alternatifs'. Ainsi, l'ensemble des réactions dans une carte métabolique KEGG ne sont pas supposées exister dans un même organisme et donc n'auront pas de fonction biologique commune à proprement parler. Cela forme une sorte d'atlas métabolique multiorganisme dédié à des composés et permet d'avoir une vue généralisée du métabolisme. KEGG définit également des modules comme des sous-ensembles de réactions de cartes métaboliques. Ces modules vont regrouper des réactions qui fonctionnent ensemble et donc sont supposées exister dans un même organisme pour réaliser une voie métabolique.

Les voies métaboliques de MetaCyc sont des ensembles de transformation enzymatiques qui vont être retrouvées ensemble dans un même organisme et qui assurent une même fonction biologique (Arnaud et al., 2005). Les voies métaboliques de MetaCyc sont beaucoup plus petites que les cartes de KEGG, et aussi beaucoup plus petites et nombreuses que les modules KEGG (Altman et al., 2013). Certaines voies peuvent même être faites d'une unique réaction, notam-

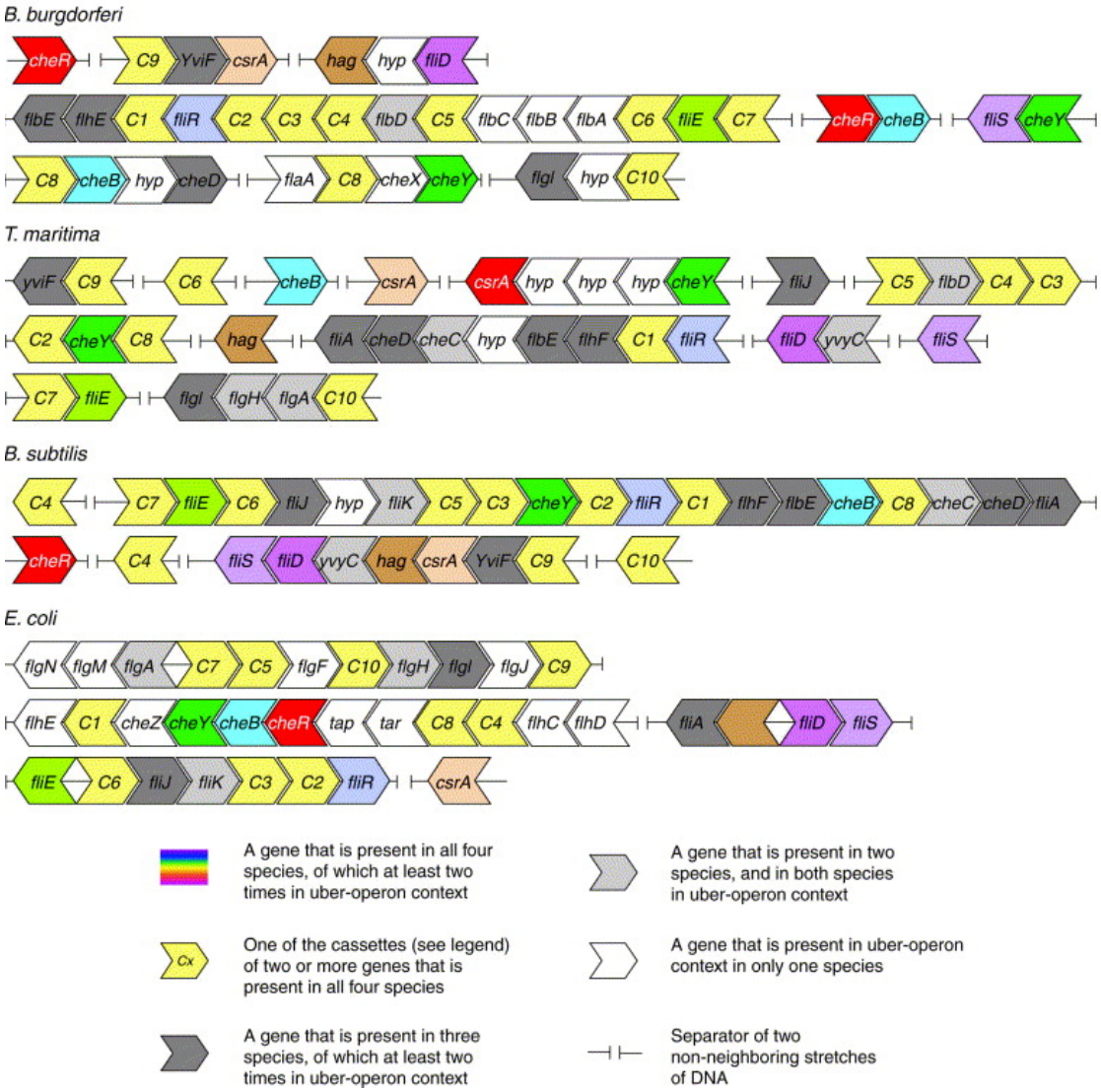


FIGURE 5.2 – Organisation génomique de 4 espèces pour des gènes impliqués dans la formation d'un flagelle
Figure illustrant l'organisation génomique de 4 espèces pour des gènes impliqués dans la formation d'un flagelle. Les gènes impliqués dans la formation du flagelle sont colorés, les gènes gris sont présents dans plusieurs génomes, mais pas tous et les gènes blancs sont présents dans un unique génome. Copié de (LATHE III et al., 2000)

ment lorsqu'une seule transformation est nécessaire pour aller d'un composé à un autre composé d'intérêt.

Plusieurs approches qui visent à retrouver des associations entre les gènes dans

des génomes nouvellement étudiés vont se servir de ces bases de données métaboliques (Anand et al., 2020 ; Aramaki et al., 2020 ; Kanehisa et al., 2016). Néanmoins, elles sont loin d'être complètes, car elles incluent uniquement les réactions et des voies qui ont été expérimentalement caractérisées, et donc représente une diversité biaisée par ce qui est très étudié : quelques eucaryotes dont *Homo sapiens* et *Saccharomyces cerevisiae* et des bactéries modèles comme *Escherichia coli* et *Bacillus subtilis*. Cela pourra être très limitant lorsqu'on travaille sur des organismes non modèles.

5.2.1.5 Identification de fonctions spécifiques

Il existe de nombreux systèmes ou outils qui ont pour but d'identifier des ensembles de gènes qui assurent une fonction unique. Ces groupes de gènes sont des modules fonctionnels par leur définition, un ensemble de gènes qui sont nécessaires pour assurer une fonction. Nous pouvons mentionner, par exemple, MACSyFinder (Abby et al., 2014) qui a pour but d'identifier les systèmes macromoléculaire comme les systèmes de sécrétions, et dont l'approche a été étendue au travers de CRISPRCasFinder(Couvin et al., 2018) pour l'identification de systèmes CRISPR-Cas, DefenseFinder (Tesson et al., 2021) pour identifier plus largement des systèmes de défenses anti-phages, et CONJScan (Cury et al., 2020) pour l'identification de systèmes conjugatifs. Ils utilisent des modèles HMM pour représenter les protéines de chaque système et vont définir si le module fonctionnel est présent ou non en fonction des protéines retrouvées dans les génomes avec un système de règles. Les règles se basent sur la présence obligatoire ou facultative de protéines et sur la co-localisation des gènes pour valider la présence d'un module fonctionnel dans un génome.

AntiSmash (Blin et al., 2019) est un autre outil spécialisé qui lui a pour but d'identifier des clusters de biosynthèse de métabolites secondaires (BGCs). Le système est assez similaire à celui de MACSyFinder. Il utilise la co-localisation des gènes et des modèles qui vont représenter les familles de protéines retrouvées dans les différents systèmes de BGCs. Pour limiter les faux positifs, d'autres modèles de protéines homologues à certains systèmes mais qui ne sont pas impliqués dans des BGCs sont également utilisés.

Utiliser des approches de ce type est très utile pour caractériser certaines fonctions, mais ne permet pas aujourd'hui, comme pour toutes les autres approches de cette section, de donner une vue globale de l'ensemble des modules fonctionnels d'un organisme ou plus largement d'une espèce. Ces approches seraient théoriquement applicables à n'importe quel type de module fonctionnel. On pourrait donc imaginer, dans un avenir lointain, des outils basés sur des modèles et un ensemble de règles expertes issues de la connaissance scientifique pour prédire l'ensemble des types de modules connus.

5.2.2 Modules conservés

5.2.2.1 Occurrences et profils phylogénétiques

Un module conversé en génomique est un ensemble de gènes qui est retrouvé à plusieurs reprises dans différents organismes. Pellegrini *et al.* montrent que l'utilisation de la co-occurrence de gènes dans différents organismes permet d'identifier des blocs de gènes impliqués dans des mêmes fonctions, par exemple, des protéines du ribosome, des protéines impliquées dans la formation d'un flagelle et la biosynthèse de l'histidine (Pellegrini et al., 1999). Les auteurs ont utilisé 17 génomes complets qui étaient séquencés à l'époque. Ils représentent un profil de présence d'une famille de gènes par un vecteur binaire, avec une valeur pour chaque génome, qui sera 1 si la famille est présente et 0 si absente. Ils utilisent une distance de Hamming entre les vecteurs pour évaluer les distances entre les familles de gènes. C'est la première application des profils phylogénétiques pour identifier des groupes de protéines avec une cohérence fonctionnelle.

La manière de calculer les familles de gènes a un impact important sur le résultat de ce type de méthode, et des familles d'orthologues ou des familles d'homologues ne rendront pas forcément les mêmes relations. De nombreuses méthodes vont émerger, et plusieurs métriques autres que la distance de Hamming ont été utilisées pour mesurer les distances entre les vecteurs de présence des familles de gènes, notamment la mesure d'information mutuelle (Mutual information) (Huynen et al., 2000), le coefficient de corrélation de Pearson (Glazko et al., 2004), le test de Fisher(Beck et al., 2018), le coefficient phi (Engelen et al., 2012) ou encore le test binomial en fonction d'une distribution théorique (Whelan et al., 2020). Ces approches ne permettent pas en soi de construire des modules, mais permettent de construire un graphe d'associations à partir duquel on peut éventuellement retrouver des modules conservés, avec des algorithmes de clustering de graphe classique. D'autres approches utilisent un partitionnement statistique, par exemple, SVD-Phy (Franceschini et al., 2016) qui réalise une décomposition en valeur singulière de la matrice de présence/absence.

Les approches se basant sur les profils phylogénétiques sont très intéressantes par leur aspect général car l'ensemble des modules conservés peuvent être potentiellement identifiés. Néanmoins, elles sont très sensibles à beaucoup de paramètres dont notamment le choix des génomes utilisés pour les profils (Muley et al., 2012) mais également leur nombre et leur diversité taxonomique (Škunca et al., 2015).

5.2.2.2 Synténie conservée

5.2.2.2.a Origine et définition

À l'origine, le terme synténie signifie deux gènes qui sont sur le même chromosome (Passarge et al., 1999). Chez les procaryotes, la définition a évolué. Les gènes n'étant regroupés que sur un ou quelques chromosomes, le terme synténie indique l'ordre des gènes dans les génomes. La synténie conservée correspond à la conservation de cet ordre des gènes dans plusieurs génomes. Les chercheurs travaillant sur les eucaryotes appellent parfois cela la "microsynténie". Cela permet de coupler deux types d'information, la présence et la localisation commune des gènes, mais complexifie grandement l'aspect méthodologique du problème. Dans l'approche originale (Tamames et al., 1997), les auteurs comparent deux génomes *E. coli* et *H. influenzae*, et mesurent que les gènes fonctionnellement associés ont tendance à se regrouper au sein des génomes. Il est donc potentiellement intéressant de rechercher ce type de conservation pour supposer que des gènes fonctionnent ensemble.

L'une des premières tentatives d'utiliser la co-localisation de gènes homologues pour former des modules conservés dans plusieurs génomes est décrite dans l'article de Overbeek *et al.* (Overbeek et al., 1999). Les auteurs utilisent 31 génomes séquencés à l'époque, très majoritairement des bactéries et des archées, qu'ils comparent en recherchant des gènes homologues par des approches de type BBH. Ils définissent des ensembles de gènes homologues proches en recherchant des paires de gènes en BBH à proximité dans deux génomes (proximité qu'ils définissent par une distance intergénique inférieure à 300bp). Les auteurs estiment qu'environ 35% des gènes avec une fonction enzymatique dans une voie métabolique connue sont conservés en synténie. Les auteurs présument néanmoins que cette proportion devrait augmenter avec le nombre de génomes disponibles. Néanmoins, leur approche devient vite irréalisable car le nombre de comparaisons est quadratique en fonction du nombre de génomes.

Beaucoup d'approches vont se développer par la suite qui vont tenter de prendre en compte les développements de la connaissance autour de la synténie et l'association fonctionnelle, et étendre le concept aux eucaryotes ayant des génomes qui sont, par certains aspects, plus complexes. Je vais maintenant introduire plusieurs de ces approches, mais n'en donnerai pas une liste exhaustive. Je ne connais malheureusement pas de revue complète récente sur ces approches qui pourrait fournir une telle liste.

5.2.2.2.b STRING

STRING est une base de données généraliste d'interactions protéines-protéines connues ou prédites. STRING est de fait une approche basée sur les graphes lo-

cale et non globale, mais je lui dédie une partie pour son importance dans la communauté. La première version de la base de donnée STRING (Snel et al., 2000b) va permettre de généraliser l'approche de recherche de synténie conservée de manière ciblée : par rapport à un gène cible, retrouver les gènes en synténie de cette cible qui sont situés à moins de 300bp du gène d'intérêt. Leur approche suit plusieurs itérations où les nouveaux gènes détectés comme synténiques deviennent gènes cibles et permettent d'étendre le bloc de synténie. Suivant cette approche, les mêmes auteurs vont tenter d'identifier des modules conservés dans 38 génomes, principalement procaryotes, à partir d'un graphe de relations de synténie inférées (Snel et al., 2002). Le graphe ainsi obtenu contient 3033 nœuds et comporte 517 composantes connexes majoritairement petites (taille moyenne 2.7), mais dont une composante comporte la moitié des nœuds, soit 1611. L'analyse de cette composante géante leur a permis de dire que ce sont des sous-clusters très connectés liés par des protéines qu'ils appellent "linker proteins" qui ont tendance à être multifonctionnelles ou être impliquées dans plusieurs processus. En séparant la grande composante en supprimant les linker proteins, ils obtiennent 265 sous-clusters. L'analyse de ces 265 sous-clusters et des autres composantes connexes, comparée à la classification COG, leur a permis de conclure que ces gènes sont dans des modules conservés ayant globalement des fonctions communes.

La base de données STRING s'est beaucoup développé par la suite. Elle va aussi inclure une quantité d'information liée à d'autres sources de contexte génomique que la synténie, notamment : la fusion/fission de gènes, la co-occurrence des gènes dans les génomes, la co-expression, les bases de données expertes externes et la co-occurrence des gènes dans la littérature. Ces informations sont combinées dans un graphe pondéré, où la pondération est une combinaison de scores de confiance venant de chaque type de relation (Von Mering et al., 2005). L'algorithme utilisé par STRING évoluera pour autoriser autre chose qu'un accord parfait entre les occurrences des gènes en utilisant la mesure d'information mutuelle. Les auteurs chercheront aussi à corriger les biais liés au nombre de génomes séquencés dans une branche de la phylogénie en regroupant dans un unique nœud les protéines des taxons dans lesquels la présence ou l'absence de gènes spécifiques est identique (Von Mering et al., 2003). L'algorithme de calcul de synténie ne changera pas au cours des versions suivantes, contrairement aux autres approches comme celles se basant sur la co-expression, sur le minage de texte, et sur les bases de données expertes (Franceschini et al., 2012 ; Lars J Jensen et al., 2009 ; Von Mering et al., 2007). La quantité et la diversité des données disponibles va grandement augmenter et plusieurs interfaces graphiques seront développées dans les versions suivantes de STRING (Szklarczyk et al., 2010 ; 2015 ; 2016 ; 2021). La base de donnée inclut, au 15 septembre 2021, 14 094 organismes et 67 millions de protéines qui interagissent dans 2 milliards de relations.

5.2.2.2.c Approches par ensembles ordonnés

Les algorithmes utilisant des ensembles ordonnés ont pour but d'identifier des sous-ensembles de gènes formés de n gènes conservés parmi m génomes. La première itération de ce type d'algorithme fut GeneTeams (Bergeron et al., 2002). Dans leur approche originale, qui ne fonctionne qu'avec deux génomes, chaque chromosome est un ensemble ordonné de gènes uniques (il ne peut pas y avoir de duplications) où les gènes communs aux deux génomes sont identifiés. Les auteurs cherchent les sous-ensembles maximaux tels que les éléments ne sont pas séparés de plus de σ gènes parmi les deux génomes. Tous les algorithmes de la même famille se baseront sur les mêmes structures, c'est-à-dire des sous-ensembles ordonnés. Plusieurs algorithmes viendront améliorer la méthode initiale, d'abord en autorisant les duplications avec COGTeams (Xin He et al., 2004), puis en traitant des domaines protéiques plutôt que des familles de gènes avec DomainTeams (Pasek et al., 2005). D'autres approches (Ling et al., 2009) vont permettre plus de comparaisons en filtrant et en découpant les ensembles ordonnés avant de comparer les sous-ensembles découpés avec une approche basée sur l'algorithme *Apriori* (Agrawal et al., 1994) dont le but est de trouver des sous-ensembles communs maximaux parmi n ensembles.

Une autre approche basée sur les ensembles est celle d'OrthoCluster (Zeng et al., 2008). Le principe est de découper un génome en fenêtres glissantes de taille d (en nombre de gènes), où chaque fenêtre est centrée sur un gène différent, et de comparer tous les ensembles ainsi générés à tous les autres ensembles générés de la même manière sur un autre génome. L'approche est simple, mais le nombre de comparaisons explose avec le nombre de génomes, et beaucoup d'ensembles communs seront chevauchants si la taille de la synténie conservée est supérieure à d.

5.2.2.2.d Approches par graphes dirigés

Il y a eu beaucoup d'approches utilisant les graphes, mais peu sont capables de comparer plus de deux génomes. L'approche de BlockFinder (Rödelsperger et al., 2007) calcule des familles de gènes (indirectement) en retrouvant les cliques dans un graphe de similarité. Ensuite, il construit un graphe orienté acyclique (DAG) dont les nœuds sont ces cliques et relient les nœuds entre eux s'ils sont voisins dans au moins m génomes. Les blocs de synténie sont alors définis comme les chemins les plus longs dans le graphe.

Une seconde approche dans le même article, Syntenator, représente les génomes par un graphe dit partiellement ordonné, où les gènes (ou paires de gènes) sont les nœuds. L'approche recherche un alignement optimal entre deux graphes, en utilisant des relations de similarité entre les gènes des deux graphes, avec un algo-

rithme de programmation dynamique. Leur approche permet d'aligner entre eux plusieurs graphes ou génomes, ce qui permet de comparer plusieurs génomes dans une même analyse. Les auteurs ont ensuite publié une version améliorée de leur approche qu'ils appelleront Cyntenator (Rödelsperger et al., 2010). Elle utilise un arbre phylogénétique pour guider l'ordre dans lequel les génomes sont alignés et inclue la distance phylogénétique dans les calculs de programmation dynamique, en pénalisant plus fortement la perte de nœuds lorsque les génomes comparés sont proches.

5.2.2.2.e MicroScope et C3-part

MicroScope est une plateforme d'analyse de génomes microbiens (Vallenet et al., 2020), développée et maintenue au LABGeM, le laboratoire où s'effectue cette thèse. Une des composantes clés de cette plateforme est que pour chaque génome, il est possible de retrouver la synténie conservée par rapport à n'importe quel autre génome de la base de données. Pour calculer les synténies, MicroScope utilise l'algorithme de C3-PART (Boyer et al., 2005), qui est un algorithme conçu pour comparer des graphes qui peuvent potentiellement être issus de sources différentes, par exemple, un génome et des voies métaboliques.

Pour la recherche de synténie, chaque génome est représenté sous forme de graphe où les nœuds sont des gènes et les arêtes des relations de voisinage dans un même génome, ou de similarité entre des génomes différents. Une fermeture transitive de taille σ est réalisée sur les relations de voisinages. Ensuite, l'algorithme de C3-PART est appliqué : il va chercher à extraire les composantes connexes communes (CCC) aux deux graphes. Une CCC est un ensemble de nœuds tel qu'ils sont tous atteignables au travers de n'importe quel type d'arêtes, c'est-à-dire qu'il existe deux sous-graphes connexes, un dans chaque génome, dont les nœuds sont tous connectés dans le graphe qui compare ces deux génomes. Cela permet d'identifier des blocs de taille maximale, qui sont appelés des blocs de synténies. La généralisation à n génomes est simple, car il suffit de chercher des CCC au sein de n génomes plutôt que deux, néanmoins cette approche est très sensible aux erreurs et la moindre différence va naturellement diminuer la taille des CCC obtenus en sortie, puisqu'il faut que tous les gènes d'un même groupe soient connectés dans tous les graphes.

Pour résoudre cette problématique, une seconde version (Deniélou et al., 2011) introduit le concept de quorum, où les gènes doivent être présents dans au moins $q(\leq n)$ génomes (ou plus généralement graphes, car leur approche s'applique sur n'importe quel type de graphe) parmi n. Pour réaliser cela, les auteurs introduisent un élément *don't care* qui autorise des non-match dans un nombre q de graphes. Pour l'utiliser, les auteurs ne font plus directement des comparaisons de graphes, mais définissent des *spines* qui représentent les blocs de synténie pos-

sibles, qui sont des ensembles de taille n, où chaque élément vient d'un graphe différent ou est un élément *don't care*. Chacun de ces ensembles va inclure entre 0 et $q-1$ éléments *don't care*. Les auteurs construisent ensuite un multigraphe où chaque nœud est un *spine*, et une arête est ajoutée entre toutes les *spines* à chaque fois que ceux-ci ont des correspondances entre leurs ensembles. Cela forme alors un multigraphe où toutes les arêtes sont associées à un graphe d'entrée, et où deux nœuds peuvent avoir plusieurs arêtes en commun (en l'occurrence, une par graphe où ils ont des correspondances). Pour extraire les blocs de synténie de ce graphe, il suffit de regrouper tous les nœuds qui ont au moins q arêtes et où les graphes n'ayant pas d'arêtes correspondent à des éléments *don't care* dans les *spines*.

Cette approche permet d'obtenir un ensemble exact de blocs de synténie, mais va difficilement au-delà de quelques dizaines de génomes.

5.2.2.2.f Gecko

Les approches du logiciel Gecko sont des méthodes basées sur les chaînes de caractères. Les génomes sont modélisés par des chaînes où les gènes sont des lettres et les blocs de synténie sont des intervalles communs entre les chaînes. Gecko (T. Schmidt et al., 2007) recherche tous les sous-ensembles communs maximaux entre les chaînes qui contiennent les mêmes sous-chaînes, sans contrainte d'ordre ou de nombre d'occurrences. Cette recherche est réalisée entre toutes les paires de génomes qui sont données en entrée. Une faiblesse de ce type d'approche (et qui est commune à beaucoup d'autres) est que les sous-ensembles doivent avoir exactement le même contenu, donc il peut y avoir des sous-ensembles chevauchants, ce qui complexifie grandement l'analyse des résultats. Pour résoudre cela, les auteurs fusionnent les sous-ensembles qui ont suffisamment de gènes en commun, ce qui permet de relâcher la condition d'exactitude des sous-ensembles et de faciliter les analyses qui en découlent. Cela ne résout que partiellement le problème car avec suffisamment de données des chevauchements entre les sous-ensembles sont toujours observés.

Une nouvelle version, appelée Gecko3 (Winter et al., 2016), propose une interface graphique et une amélioration de l'algorithme au niveau des conditions d'exactitude des sous-ensembles. Gecko3 définit des "clusters de gènes de références" qui ont des occurrences exactes dans au moins un des génomes et possiblement inexactes dans un nombre suffisant d'autres génomes. La mesure d'inexactitude est basée sur le nombre de gènes qui doivent être ajoutés ou supprimés du cluster pour correspondre à l'occurrence exacte. Les auteurs déclarent réussir à travailler avec des centaines de génomes avec leur approche grâce à des ruses d'ingénieries algorithmiques.

5.2.2.2.g CSBFinder

CSBFinder (Svetlitsky et al., 2019) est une approche dont le but est d'identifier l'ensemble des blocs colinéaires de synténie (CSB, conserved syntenic block) dans des centaines, voir milliers, de génomes procaryotes. Des gènes sont dits colinéaires s'ils sont sur le même brin. La notion de colinéarité était depuis longtemps utilisée chez les eucaryotes, mais beaucoup moins chez les procaryotes même si d'autres outils de détection de blocs de synténie se sont comparés à des bases de données d'opérons. Les auteurs utilisent un arbre des suffixes, utilisable grâce à la notion de colinéarité qui permet de définir des suffixes. Ils vont commencer par énumérer l'ensemble des directons (gènes successifs sur le même brin) dans l'ensemble des génomes. Les auteurs construisent ensuite un arbre des suffixes avec ces directons, dont l'alphabet est l'ensemble des familles de gènes. Les CSB sont définis comme étant tous les éléments d'un ensemble séparés par moins de k éléments dans au moins q génomes. Leur définition a le même problème que bien d'autres, qui est qu'il peut y avoir parfois de nombreux gènes en commun entre des CSB différents. Pour résoudre ce problème, les auteurs regroupent les CSB en familles avec un algorithme très ressemblant à celui de CD-hit. Ils itèrent sur les CSB du plus grand au plus petit et associent un CSB a une famille existante si celui-ci lui ressemble suffisamment, ressemblance qui est calculée avec un index de Jaccard entre les gènes des deux CSB comparés. Si aucune famille ne remplit la condition, une nouvelle famille de CSB est créée. Cela permet de résoudre partiellement le problème, mais, en pratique, des chevauchements entre les familles de CSB sont toujours notables.

L'approche est limitée aux gènes strictement conservés en un unique opéron, néanmoins pour certaines fonctions l'organisation peut être multi-opéroniques. Pour résoudre cette problématique, les auteurs ont développé par la suite un second algorithme appelé CSBFinder-S (Svetlitsky et al., 2020). Celui-ci utilise un algorithme de *match-point arithmetic* qui est dit beaucoup plus efficace pour des larges valeurs de k (le nombre d'éléments non inclus dans un CSB qui séparent les éléments d'un même CSB).

5.2.2.3 Phylogénie

La phylogénétique est l'inférence de l'histoire évolutive entre des entités via l'observation de traits héritables comme notamment les séquences d'acides aminés des protéines. Un résultat classique d'une analyse phylogénétique est un arbre phylogénétique qui va décrire les relations de parenté entre ce qui a été étudié, que ce soit des gènes, des génomes ou des organismes. On peut donc se baser sur ces arbres pour essayer d'identifier ce qui suit une évolution commune. Plusieurs approches utilisent le concept général qui est que deux familles de gènes qui ap-

paraissent conjointement dans les génomes et qui subissent les mêmes événements évolutifs peuvent être fonctionnellement associées. Il y a plusieurs approches qui permettent d'identifier des familles de gènes qui suivent une évolution similaire.

Dans un article de 2001 (Pazos et al., 2001), les auteurs suggèrent de calculer le coefficient de corrélation entre deux familles de gènes d'intérêt, en utilisant les matrices de distances entre les protéines d'une même famille comme proxy de la phylogénie. Les auteurs soulignent néanmoins que ces matrices sont des représentations approximatives des arbres de gènes mais, à l'époque, il n'existait pas vraiment d'approche qui permettait de comparer les arbres phylogénétiques aisément.

Les auteurs de Barker *et al.* (Barker et al., 2005) proposeront une alternative qui est basée sur la comparaison d'arbres de gènes réconciliés — ce sont des arbres de gènes annotés avec des événements évolutifs pour chaque nœud avec l'aide d'un arbre d'espèce. Ils mesurent sur une analyse de 15 génomes que les paires de protéines qui partagent au moins deux à trois événements évolutifs de pertes ou de gains sont souvent systématiquement fonctionnellement associées. Leur approche phylogénétique permet de différencier les motifs qui apparaissent uniquement par héritage de ceux qui sont réellement des gains et des pertes communes, ce que les approches de profils phylogénétiques ne permettent pas malgré leur nom peut-être mal choisi. Le problème de différentier les co-occurrences par héritage de celles par événements communs sera notamment discuté dans l'article de Maddison *et al.* (Maddison et al., 2015) où les auteurs soulignent qu'il n'y a pas véritablement de tests pour les variables catégorielles qui permettent de différencier ces deux cas. Une solution sera proposée dans le contexte des pangénomes bactériens dans l'approche de Scoary (Brynildsrud et al., 2016) qui est une méthode d'association pour identifier les familles de gènes responsables d'un phénotype tentant de prendre en compte la structure de la phylogénie. Les auteurs de Scoary soulignent notamment qu'utiliser un test de Fisher pour comparer des vecteurs de présence/absence est inadapté car les dépendances entre les lignées (c.-à-d. l'héritabilité) est une violation du modèle aléatoire implicite du test.

Toujours dans le contexte des pangénomes bactériens, Pantagruel (Lassalle et al., 2019) est une méthode spécifiquement dédiée à l'identification de paires de gènes qui ont suivi une évolution corrélée dans un arbre de gène réconcilié. Les auteurs utilisent pour cela un score de similarité qui décrit le nombre moyen d'événements évolutifs communs dans l'histoire évolutive de deux familles de gènes. Leur approche permet de constituer un graphe où les nœuds sont des familles et les arêtes des associations obtenues avec leur approche. Les auteurs soulignent néanmoins que le réseau obtenu est particulièrement dense.

Toutes les méthodes se basant sur des arbres ou des proxys de ceux-ci renvoient des relations inférées mais aucune ne tente explicitement de reconstruire des modules conservés.

5.3 Liens entre modules fonctionnels et modules conservés

Les modules fonctionnels sont générés grâce aux relations qui sont inférées au sein d'un ou plusieurs génomes donnés, que ça soit expérimentalement ou au travers d'une structure opéronique ou de fusions de gènes. Les modules conservés quant à eux regroupent des gènes qui partagent des événements évolutifs grâce à leur profil de présence et, éventuellement, de co-localisation. L'association entre les deux peut sembler de prime abord aisée : ce qui est conservé l'est pour une raison d'utilité, voire de nécessité, et avec assez de signal on pourrait supposer être capable de reconstruire des modules conservés qui sont effectivement des modules fonctionnels.

Dans l'article de Snel *et al.* (Snel et al., 2004), les auteurs posent de nombreuses questions autour de la modularité et de ce qu'est un module fonctionnel ou un module conservé plus formellement. Est-ce qu'il faut considérer uniquement les voies métaboliques ou les complexes protéiques ? *quid* des protéines co-régulées ? Ils soulignent aussi que même lorsque les approches les plus stringentes sont utilisées dans la reconstruction de familles de gènes, c'est-à-dire la reconstruction de familles d'orthologues, des différences fonctionnelles entre les protéines de la même famille sont retrouvées. Le cas échéant, comment les prendre en compte ? Ils ont tenté d'évaluer, en utilisant 110 génomes majoritairement procaryotes, à quel point les modules fonctionnels étaient conservés au cours de l'évolution en fonction de différentes manières de considérer les deux types de modules. Ils ont aussi cherché à mesurer la modularité des modules fonctionnels en fonction de leurs catégories fonctionnelles, notamment pour les modules métaboliques, les complexes protéiques et les modules transcriptionnels. Les auteurs mesurent la modularité en calculant la déviation par rapport à une modularité parfaite et à un modèle nul d'une distribution aléatoire de gènes, qu'ils définissent de deux manières différentes : une qui ignore totalement la phylogénie et une qui tente de la prendre en compte en ne comparant que des ensembles de gènes avec une fréquence similaire parmi les génomes. En considérant l'ensemble de leurs modules fonctionnels, les auteurs démontrent que ceux-ci sont plus proches d'une distribution aléatoire que de modules parfaitement conservés. Néanmoins, ils concluent que la plupart des modules fonctionnels sont plus modulaires que l'aléatoire. Entre leurs deux manières de mesurer le modèle nul, les auteurs notent, peut-être étonnamment, que considérer la phylogénie leur fait mesurer une modularité beaucoup plus faible qu'en l'ignorant totalement. Cela semble indiquer, pour les auteurs, qu'une portion importante de la modularité observée ne peut pas être discriminée des effets de la phylogénie. Par la suite, les auteurs ont cherché les sources potentielles d'erreurs dans leurs modules. Pour corriger leur jeu de données, ils se sont ensuite servis

du fait que combiner les signaux venant de types de modules différents permettait d'avoir une bien meilleure précision (Von Mering et al., 2002) et ont ainsi combiné les informations de leurs 9 sources de modules en ne conservant, pour chaque jeu de données, que les protéines vues ensemble dans au moins un autre jeu de données de modules. La modularité mesurée en filtrant ainsi leur jeu de données était largement supérieure pour les deux manières de mesurer la modularité, mais la tendance observée vis-à-vis de la phylogénie était la même. Néanmoins, aucun de leurs jeux de données n'était mesuré comme parfaitement modulaire, même avec les corrections appliquées en les comparant. Les auteurs ont ensuite comparé les analyses de modularité de leurs différents jeux de données. Ils mesurent notamment que les voies métaboliques de KEGG et EcoCyc (base de données sur le métabolisme de *E. coli* K-12) sont très modulaires, mais que celles de EcoCyc le sont beaucoup plus. Ils mesurent aussi que les opérons extraits de regulonDB sont très modulaires, contrairement aux unités transcriptionnelles d'autres sources de données. Les régulons eux montrent moins de modularité que les opérons. En analysant plus spécifiquement EcoCyc, les auteurs mesurent que les voies métaboliques de biosynthèse sont plus modulaires que les voies de dégradation, suggérant que ces voies avaient une plus grande flexibilité : il y aurait peut-être plus de manières de casser des composés que de les synthétiser.

La variabilité des voies de dégradation a été observée aussi à l'échelle pangénomique. Dans l'article de Vieira *et al.* (Vieira et al., 2011), les auteurs ont étudié un panmétabolome de *E. coli*. Concrètement, ils ont recherché toutes les voies connues au sein du pangénome. Ils ont notamment mesuré que les réactions retrouvées dans le *core* génome sont enrichies en voies de biosynthèse et que les voies de dégradation étaient au contraire plus présentes dans le génome accessoire, suggérant que ces voies de dégradation étaient plus susceptibles de changer et d'être transférées horizontalement ou perdues que les voies de biosynthèse.

Ces analyses nous permettent de conclure que s'il y a effectivement un lien fort entre modules fonctionnels et modules conservés ceux-ci ne sont pas identiques. Pour certaines fonctions, la flexibilité semble être nécessaire à leur adaptation et celles-ci ne pourront être identifiées par des approches de reconstruction de modules conservés. Pour d'autres fonctions, contraintes par des nécessités physiques, comme les complexes protéiques, ou physiologiques, comme les voies de biosynthèses, la modularité sera intrinsèquement nécessaire au maintien de la fonction dans l'organisme et permettra l'identification des gènes qui la réalise au travers de ces approches. Comme pour toutes les autres méthodes permettant d'inférer des interactions entre les protéines, les méthodes basées sur les modules conservés ne permettent pas de décrire l'ensemble des interactions.

Troisième partie

Résultats

Maintenant que j'ai introduit tous les concepts qui ont été au cœur de cette thèse, je vais désormais détailler les travaux que j'ai pu réaliser autour de ces concepts. Le premier chapitre de cette partie, qui a été introduit par le chapitre 3 sur la génomique comparative et les pangénomes, présente la méthode PPanGGOLiN de reconstruction d'un graphe de pangénome et de son partitionnement. Le second chapitre de cette partie, qui a été introduit par le chapitre 4 sur les îlots génomiques et les spots d'insertions, présente la méthode panRGP de détection de régions de plasticité génomique dans les pangénomes. Le troisième chapitre de cette partie, qui a été introduit par le chapitre 5 sur les modules fonctionnels, présente la méthode panModule pour la recherche de modules conservés dans les îlots génomiques directement dans un graphe de pangénome. Le quatrième chapitre concerne la constitution d'une base de données de pangénomes, nommée panGBank.

Chapitre 6

PPanGGOLiN

6.1 Graphes de pangénome et partitionnement statistique

6.1.1 Objectifs de PPanGGOLiN

L'immense majorité des auteurs de logiciels ou d'analyses de pangénomes préfèrent utiliser une classification dichotomique du pangénome, réalisée avec un seuil arbitrairement choisi basé sur la fréquence des familles de gènes pour déterminer le *core* génome. Pourtant, une grande faiblesse de ces approches est que, avec un nombre suffisant de génomes, le *core* tend à diminuer bien en dessous des prédictions qui avaient été faites initialement dans les articles de (Tettelin et al., 2005) ou (Willenbrock et al., 2007) par exemple. Biologiquement, le *core* représente ce qui caractérise l'espèce. On s'attend donc à ce que celui-ci soit stable à partir du moment où les génomes que l'on manipule sont effectivement représentatifs de cette espèce. Une approche statistique pourrait permettre de résoudre ces problèmes. Néanmoins, la seule méthode de partitionnement statistique existante, micropan (Snipen et al., 2009 ; 2015), ne passe pas à l'échelle au-delà de 300 génomes et surtout se base exclusivement sur la fréquence des familles de gènes pour sa classification. Or, on a pu le voir brièvement dans la sous-section 2.1.4, un gène n'évolue pas seul, mais toujours dans un contexte particulier, propre à sa fonction et à l'espèce dans laquelle il est.

La méthode PPanGGOLiN a été conçue pour répondre à ces deux problématiques. Premièrement, il s'agit de proposer un modèle statistique qui permet d'obtenir un partitionnement biologiquement crédible, c'est-à-dire un *core* génome stable quel que soit l'échantillon, tant qu'il y a un nombre suffisant de génomes dans l'analyse. Deuxièmement, il faut que ce modèle statistique ne prenne pas en compte la seule fréquence des familles de gènes, mais aussi le contexte dans lequel

elles évoluent.

6.1.2 Principe de l'approche et analyses

Pour permettre de prendre en compte le contexte des gènes, PPanGGOLiN utilise un modèle de graphe de pangénome. Le principe d'un graphe de pangénome est de représenter l'ensemble du pangénome sous forme de graphe, ou les nœuds sont des familles de gènes et les arêtes sont des liens de voisinage génomique entre les gènes. Ce graphe sera un agrégat de l'ensemble des génomes qui constituent le pangénome. Il sera ensuite partitionné en utilisant une méthode appelée NEM, pour Neighborhood Expectation Maximisation (Ambroise et al., 1997), qui combine un champ de Markov caché (pour l'aspect graphe et contexte) avec un modèle mixé de Bernoulli (pour l'aspect présence des familles). Ce modèle permet de partitionner le pangénome en K classes, respectivement reclassées en 3 parties :

— le génome *persistent* : Celui-ci correspond au *core* et représente les familles de gènes qui sont persistantes dans l'espèce étudiée. Cette partie inclut uniquement la classe la plus présente.

— le génome *shell* : Il s'agit des familles de gènes présentes à des fréquences intermédiaires au sein du pangénome. Cette partie inclut toutes les sous-classes intermédiaires.

— Le génome *cloud* : Il inclut des familles de gènes présentes très rarement, voir une unique fois dans le pangénome. Cette partie inclut uniquement la classe la moins présente.

L'article de PPanGGOLiN, publié le 19 Mars 2020, décrit la méthode statistique et le modèle de graphe utilisé. Il illustre ensuite sur 88 espèces de bactéries, qui contenaient plus de 100 génomes dans GenBank, la stabilité des résultats de la méthode proposée par rapport à l'usage d'un seuil arbitraire sur la fréquence des familles, qui est classiquement utilisé dans la littérature pour l'analyse des pangénomes.

Une analyse plus complète du *shell* montre que, lorsque PPanGGOLiN prédit un *shell* très important en proportion de la taille des génomes, le pangénome a généralement plus de 3 classes prédites lors du partitionnement et que le contenu du *shell* a tendance à être fortement corrélé avec la phylogénie. Cette corrélation indiquerait une structuration forte du *shell* lorsque les génomes étudiés semblent pouvoir se découper en sous-populations. On note deux exceptions à cette observation, *Shigella sonnei* et *Limosilactobacillus reuteri* (anciennement *Lactobacillus reuteri*). Dans le cas de *S. sonnei*, c'est un génome notoirement connu pour être un génome en réduction avec beaucoup de gènes fragmentés et pseudogénéisés, ce qui peut expliquer un large *shell*, mais qui serait non structuré par la phylogénie.

S'agissant de *L. reuteri*, cette bactérie colonise les intestins des vertébrés et sa phylogénie est fortement structurée par les hôtes des différentes souches. Apparemment, les profils de familles de gènes observés sont partagés entre des lignées distinctes et ces familles peuvent être des facteurs d'adaptation à un même hôte.

Finalement, une application de PPanGGOLiN sera faite sur des données de métagénomique, à partir de MAG (Metagenome Assembled-genomes) qui sont des génomes notoirement de mauvaise qualité par rapport aux génomes issus d'un séquençage plus classique d'isolats. On illustre alors la capacité, à l'époque unique, de PPanGGOLiN d'être capable de retrouver le génome *persistent* des espèces en utilisant des données fragmentées et incomplètes.

Ce travail, notamment le développement de la méthode statistique a été conduit dans le cadre de la thèse de Guillaume Gautreau (Gautreau, 2020). Ma participation s'est faite notamment autour du développement du logiciel pour le rendre plus facilement utilisable, plus parallélisable lors de son exécution et plus modulable pour pouvoir y ajouter d'autres méthodes par la suite. J'ai travaillé sur la définition de la structure de données stockée dans un fichier HDF5 pour chaque pangénome, qui permet de relire ou de refaire uniquement certaines analyses parmi celles réalisables par PPanGGOLiN. J'ai participé à différentes analyses de l'article notamment celles autour des MAGs et de la structuration du *shell* relativement à la phylogénie. J'ai aussi eu le plaisir de réécrire le wiki du logiciel.

Par la suite, j'ai maintenu le logiciel (et le wiki, en partie) et agrémenté celui-ci d'autres fonctionnalités pratiques, occasionnellement demandées par des utilisateurs. J'ai aussi réalisé le support, au travers des 'issues' et 'pull requests' que les utilisateurs peuvent laisser sur le dépôt de code GitHub et aidé, comme je l'ai pu, certains utilisateurs avec leurs analyses de pangénome par d'autres canaux de communication.

6.2 Article 1 : PPanGGOLiN : depicting microbial diversity via a partitioned pangenome graph

PLOS COMPUTATIONAL BIOLOGY

RESEARCH ARTICLE

PPanGGOLiN: Depicting microbial diversity via a partitioned pangenome graph

Guillaume Gautreau[1], Adelme Bazin[1], Mathieu Gachet[1], Rémi Planel[1¤a], Laura Burlot[1], Mathieu Dubois[1], Amandine Perrin[2,3], Claudine Médigue[1], Alexandra Calteau[1], Stéphane Cruveiller[1¤b], Catherine Matias[4], Christophe Ambroise[5], Eduardo P. C. Rocha[2], David Vallenet[1]*

1 LABGeM, Génomique Métabolique, CEA, Genoscope, Institut François Jacob, Université d'Évry, Université Paris-Saclay, CNRS, Evry, France, **2** Microbial Evolutionary Genomics, Institut Pasteur, CNRS, UMR3525, Paris, France, **3** Sorbonne Université, Collège doctoral, Paris, France, **4** Laboratoire de Probabilités, Statistique et Modélisation, Sorbonne Université, Université de Paris, Centre National de la Recherche Scientifique, Paris, France, **5** Laboratoire de Mathématiques et Modélisation d'Evry, UMR CNRS 8071, Université d'Evry Val d'Essonne, Evry, France

¤a Current address: Hub de Bioinformatique et Biostatistique - Département Biologie Computationnelle, Institut Pasteur, USR 3756 CNRS, Paris, France
¤b Current address: PathoQuest SAS, BioPark – bâtiment B, 11 rue Watt, 75013 Paris, France
* vallenet@genoscope.cns.fr

Check for updates

OPEN ACCESS

Citation: Gautreau G, Bazin A, Gachet M, Planel R, Burlot L, Dubois M, et al. (2020) PPanGGOLiN: Depicting microbial diversity via a partitioned pangenome graph. PLoS Comput Biol 16(3): e1007732. https://doi.org/10.1371/journal.pcbi.1007732

Editor: Christos A. Ouzounis, CPERI, GREECE

Received: November 19, 2019

Accepted: February 12, 2020

Published: March 19, 2020

Peer Review History: PLOS recognizes the benefits of transparency in the peer review process; therefore, we enable the publication of all of the content of peer review and author responses alongside final, published articles. The editorial history of this article is available here: https://doi.org/10.1371/journal.pcbi.1007732

Copyright: © 2020 Gautreau et al. This is an open access article distributed under the terms of the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

Data Availability Statement: Archaeal and bacterial genomes were downloaded from the NCBI FTP server (ftp://ftp.ncbi.nlm.nih.gov/genomes/genbank) 17 April 2019. Metagenome-Assembled Genomes were downloaded from

Abstract

The use of comparative genomics for functional, evolutionary, and epidemiological studies requires methods to classify gene families in terms of occurrence in a given species. These methods usually lack multivariate statistical models to infer the partitions and the optimal number of classes and don't account for genome organization. We introduce a graph structure to model pangenomes in which nodes represent gene families and edges represent genomic neighborhood. Our method, named PPanGGOLiN, partitions nodes using an Expectation-Maximization algorithm based on multivariate Bernoulli Mixture Model coupled with a Markov Random Field. This approach takes into account the topology of the graph and the presence/absence of genes in pangenomes to classify gene families into persistent, cloud, and one or several shell partitions. By analyzing the partitioned pangenome graphs of isolate genomes from 439 species and metagenome-assembled genomes from 78 species, we demonstrate that our method is effective in estimating the persistent genome. Interestingly, it shows that the shell genome is a key element to understand genome dynamics, presumably because it reflects how genes present at intermediate frequencies drive adaptation of species, and its proportion in genomes is independent of genome size. The graph-based approach proposed by PPanGGOLiN is useful to depict the overall genomic diversity of thousands of strains in a compact structure and provides an effective basis for very large scale comparative genomics. The software is freely available at https://github.com/labgem/PPanGGOLiN.

https://opendata.lifebit.ai/table/SGB. All analyses described here were run using PPanGGOLiN software (version 1.0). PPanGGOLiN source code is freely available from https://github.com/labgem/PPanGGOLiN under a CeCILL license. All relevant data are within the manuscript and its Supporting Information files.

Funding: This research was supported in part by the IRTELIS and Phare PhD programs of the French Alternative Energies and Atomic Energy Commission (CEA) for GG and AB respectively, the French Government "Investissements d'Avenir" programs (namely FRANCE GENOMIQUE [ANR-10-INBS-09-08], the INSTITUT FRANÇAIS DE BIOINFORMATIQUE [ANR-11-INBS-0013], and the Agence Nationale de la Recherche [Projet ANR-16-CE12-29 for EPCR]). The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Competing interests: The authors have declared that no competing interests exist.

Author summary

Microorganisms have the greatest biodiversity and evolutionary history on earth. At the genomic level, it is reflected by a highly variable gene content even among organisms from the same species which explains the ability of microbes to be pathogenic or to grow in specific environments. We developed a new method called PPanGGOLiN which accurately represents the genomic diversity of a species (i.e. its pangenome) using a compact graph structure. Based on this pangenome graph, we classify genes by a statistical method according to their occurrence in the genomes. This method allowed us to build pangenomes even for uncultivated species at an unprecedented scale. We applied our method on all available genomes in databanks in order to depict the overall diversity of hundreds of species. Overall, our work enables microbiologists to explore and visualize pangenomes alike a subway map.

Introduction

The analyses of the gene repertoire diversity of species—their pangenome—have many applications in functional, evolutionary, and epidemiological studies [1, 2]. The core genome is defined as the set of genes shared by all the genomes of a taxonomic unit (generally a species) whereas the accessory (or variable) genome contains genes that are only present in some genomes. The latter is crucial to understand bacterial adaptation as it contains a large repertoire of genes that may confer distinct traits and explain many of the phenotypic differences across species. Most of these genes are acquired by horizontal gene transfer (HGT) [3]. This usual dichotomy between core and accessory genomes does not consider the diverse ranges of gene frequencies in a pangenome. The main problem in using a strict definition of the core genome is that its size decreases as more genomes are added to the analysis [4] due to gene loss events and technical artifacts (i.e. sequencing, assembly or annotation issues). As a consequence, it was proposed in the field of synthetic biology to focus on persistent genes, i.e. those conserved in a large majority of genomes [5]. The persistent genome is also called the soft core [6], the extended core [7, 8] or the stabilome [9]. These definitions advocate for the use of a threshold frequency of a gene family within a species above which it is considered as *de facto* core gene. Persistent gene families are usually defined as those present in a range comprised between 90% [10] and 99% [11] of the strains in the species. This approach addresses some problems of the original definition of core genome but requires the setting of an appropriate threshold. The gene frequency distribution in pangenomes is extensively documented [7, 8, 12–16]. Due to the variation in the rates of gene loss and gain of genes, the gene frequencies tend to show an asymmetric U-shaped distribution regardless of the phylogenetic level and the clade considered (with the exception of few species having non-homogeneous distributions as described in [17]). Thereby, as proposed by Koonin and Wolf [12] and formally modeled by Collins and Higgs [14], the pangenome can be split into 3 classes: (1) persistent genome, for the gene families present in almost all genomes; (2) shell genome, for gene families present at intermediate frequencies in the species; (3) cloud genome, for gene families present at low frequency in the species.

The study of pangenomes in microbiology now relies on the comparison of hundreds to thousands of genomes of a single species. The analysis of this massive amount of data raises computational and algorithmic challenges that can be tackled because genomes within a species have many homologous genes and it is possible to design new compact ways of representing and manipulating this information. As suggested by Chan *et al.* [18], a consensus representation of multiple genomes would provide a better analytical framework than using individual

reference genomes. Among others, this proposition has led to a paradigm shift from the usual linear representation of reference genomes to a representation as variation graphs (also named "genome graphs" or "pangenome graphs") bringing together all the different known variations as multiple alternative paths. Methods [19–21] have been developed aiming at factorizing pangenomes at the genome sequence-level to capture all the nucleotide variations in a graph that enables variant calling and improves the sensitivity of the read mapping (summarized in [22]).

The method presented here, named PPanGGOLiN (Partitioned PanGenome Graph Of Linked Neighbors), introduces a new representation of the gene repertoire variation as a graph, where each node represents a family of homologous genes and each edge indicates a relation of genetic contiguity. PPanGGOLiN fills the gap between the standard pangenomic approach (that uses a set of independent and isolated gene families) and sequence-level pangenome graph (as reviewed in [23]). The interest of a gene-level graph compared to a sequence graph is that it provides a much more compact structure in clades where gene gains and losses are the major drivers of adaptation. This comes at the cost of disregarding polymorphism in genes and ignoring variation in intergenic regions and introns. However, the genomes of prokaryotes have very small intergenic regions and are almost devoid of introns justifying a focus on the variation of gene repertoires [12], which can be complemented by analysis of intergenic and intragenic polymorphism. PPanGGOLiN uses a new statistical model to classify gene families into persistent, cloud, and one or several shell partitions. To the best of our knowledge three statistical methods are available to partition a pangenome. Two of them use probabilistic models that partition dichotomously the pangenome only into core and accessory components [24, 25]. Conversely, the method proposed and implemented by Snipen *et al.* [26, 27] (micropan R package) classifies a pangenome in *K* partitions using a Binomial Mixture Model relying on gene family frequencies. Unlike these three methods, PPanGGOLiN is not based on frequencies but combines both the patterns of occurrence of gene families and the pangenome graph topology to perform the classification. In the following sections we present an overview of the method, an illustration of a pangenome graph and then the partitioning of a large set of prokaryotic species from GenBank. We evaluate the relevance of the persistent genome computed by PPanGGOLiN in comparison to the classical soft core genome. Next, we illustrate the importance of the shell structure and dynamics in the study of the evolution of microbial genomes. Finally, we compare GenBank results to the ones obtained with Metagenome-Assembled Genomes (MAGs) to validate the use of PPanGGOLiN for metagenomic applications.

Results and discussion

Overview of the PPanGGOLiN method

PPanGGOLiN builds pangenomes for large sets of prokaryotic genomes (i.e. several thousands) through a graphical model and a statistical method to classify gene families into three classes: persistent, cloud, and one or several shell partitions. It uses as input a set of annotated genomes with their coding regions classified in homologous gene families. As depicted in Fig 1, PPanGGOLiN integrates information on protein-coding genes and their genomic neighborhood to build a graph where each node is a gene family and each edge is a relation of genetic contiguity (two families are linked in the graph if they contain genes that are neighbors in the genomes). Thanks to this graphical model, the structure of the pangenome is resilient to fragmented assemblies: an assembly gap in one genome can be offset by information from other genomes, thus maintaining the link in the graph. To partition this graph, we established a statistical model taking into consideration that persistent genes share conserved genomic organizations along genomes (i.e. synteny conservation) [28] and that horizontally transferred genes

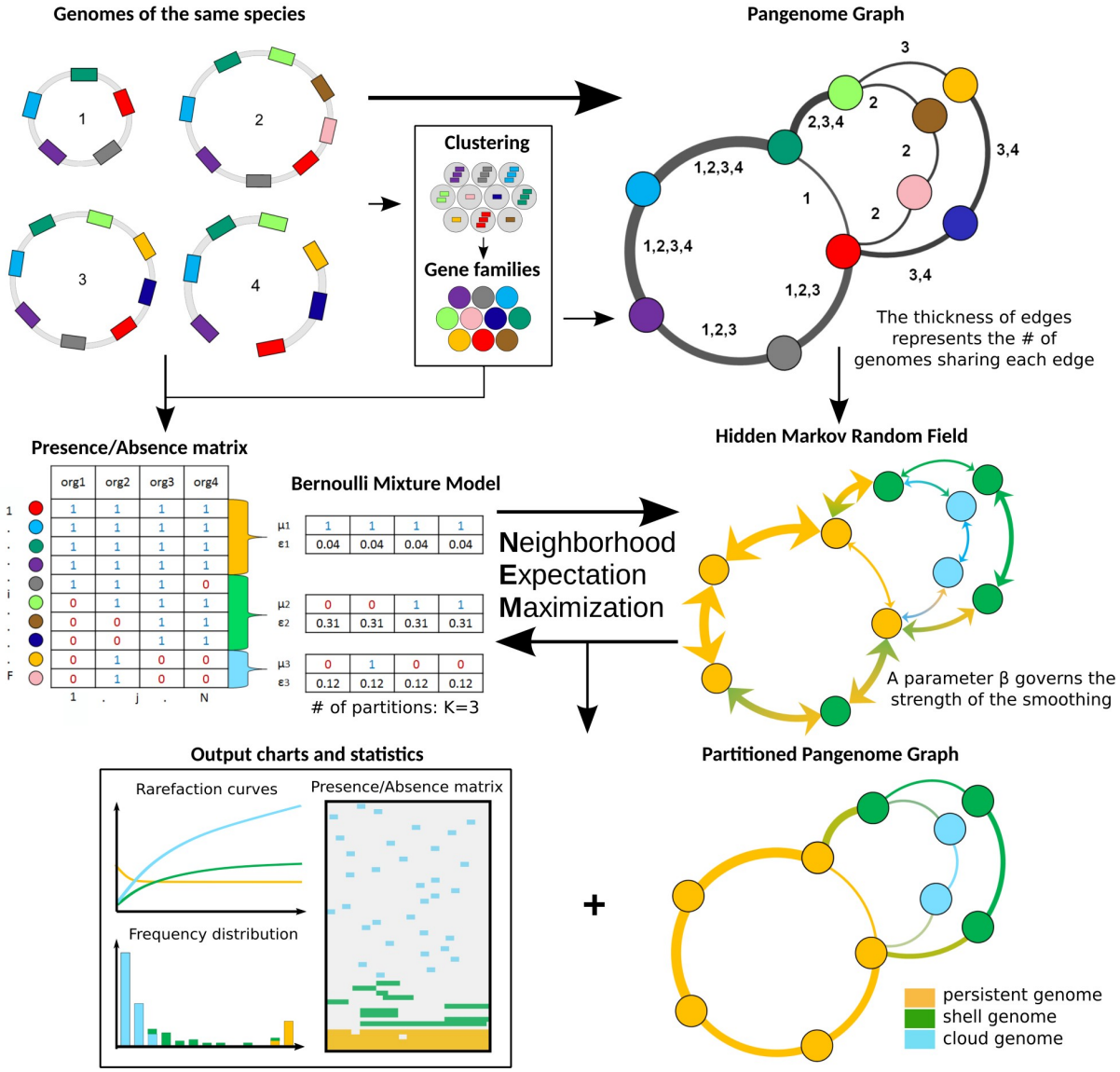


Fig 1. Flowchart of PPanGGOLiN on a toy example of 4 genomes. The method requires annotated genomes of the same species with their genes clustered into homologous gene families. Annotations and gene families can be predicted by PPanGGOLiN or directly provided by the user. Based on these inputs, a pangenome graph is built by merging homologous genes and their genomic links. Nodes represent gene families and edges represent genomic neighborhood. The edges are labeled by identifiers of genomes sharing the same gene neighborhood. In parallel, gene families are encoded as a presence/absence matrix that indicates for each family whether or not it is present in the genomes. The pangenome is then divided into K partitions ($K = 3$ in this example) by estimating the best partitioning parameters through an Expectation-Maximization algorithm. The method involves the maximization of the likelihood of a multivariate Bernoulli Mixture Model taking into account the constraint of a Markov Random Field (MRF). The MRF network is given by the pangenome graph and it favors two neighbors to be more likely classified in the same partition. At the end of this iterative process, PPanGGOLiN returns a partitioned pangenome graph where persistent, shell and cloud partitions are overlaid on the neighborhood graph. In addition, many tables, charts and statistics are provided by the software. The number of partitions (K) can either be provided by the user or determined by the algorithm.

https://doi.org/10.1371/journal.pcbi.1007732.g001

(i.e. shell and cloud genes) tend to insert preferentially in a few chromosomal regions (hotspots) [29]. Thereby, PPanGGOLiN favors two gene families that are consistent neighbors in the graph to be more likely classified in the same partition. This is achieved by a hidden Markov Random Field (MRF) whose network is given by the pangenome graph. In parallel, the

pangenome is also represented as a binary Presence/Absence (P/A) matrix where the rows correspond to gene families and the columns to genomes. Values are 1 for the presence of at least one member of the gene family and 0 otherwise. This P/A matrix is modeled by a multivariate Bernoulli Mixture Model (BMM). Its parameters are estimated via an Expectation-Maximization (EM) algorithm taking into account the constraints imposed by the MRF. Each gene family is then associated to its closest partition according to the BMM. This results in a partitioned pangenome graph made of nodes that are classified as either persistent, shell or cloud. The strength of the MRF constraints increases according to a parameter called β (if $\beta = 0$, the effect of the MRF is disabled and the partitioning only relies on the P/A matrix) and it depends on the weight of the edges of the pangenome graph which represents the number of gene pairs sharing the neighborhood. Another originality of our method is that, even if the number of partitions (K) is estimated to be equal to 3 (persistent, shell, cloud) in most cases (see 'Analyses of the most represented species in databanks' section), more partitions can be used if the pangenome matrix contains several contrasted patterns of P/A. These additional partitions are considered to belong to the shell genome and reflect a heterogeneous structure of the shell (see 'Shell structure and dynamics' section).

Illustration of a partitioned pangenome graph depicting the *Acinetobacter baumannii* species

We computed the pangenome of 3 117 *Acinetobacter baumannii* genomes from GenBank using PPanGGOLiN. For the persistent, shell and cloud genomes, we obtained 3 084, 1 529 and 64 833 gene families, respectively. If we compare our results with those of Chan *et al.* study [18], the size of the persistent genome predicted by PPanGGOLiN is included in their soft core estimation ranging from 2 833 (95% of presence) to 3 126 (75% of presence) gene families using 249 *A. baumannii* genomes. On the partitioned pangenome graph built with PPanGGOLiN (Fig 2), the gene families classified as persistent (orange nodes) correspond to the conserved paths that are interrupted by many islands composed of shell (green nodes) and cloud genomes (blue nodes). These islands appear to be frequently inserted in hotspots of the persistent genome thus pinpointing regions of genome plasticity. The average node degree within the same partition is 2.80 for the persistent genome while the shell genome has a higher average degree (3.95, P = 5.0e-6 with a bilateral unpaired 2-sample Student's t test) and the cloud a lower one (1.97, P = 3.3e-40 with the same test). The shell genome is the most diversified in terms of network topology with many interconnections between families reflecting a mosaic composition of regions from different HGT events [29]. The major part of the cloud has a shell-like graph topology with a large connected component containing 60% of the nodes. In addition, the cloud also contains isolated components that are nearly linear (3 606 components having on average 4.25 nodes) and singletons (10 575 nodes), presumably because it includes very recently acquired genetic material. Finally, large families of mobile genes, mostly transposable elements, can be easily detected because they constitute hubs (i.e. highly connected nodes) in the graph. They vary rapidly their genetic neighborhoods and can be found in multiple loci.

As an example of a more detailed analysis that can be done using the graph, a zoom on a region containing the genes required for the synthesis of capsular polysaccharides is highlighted in Fig 2. *A. baumannii* strains are involved in numerous nosocomial infections and their capsule plays key roles in the overall fitness and pathogenicity. Indeed, it protects the bacteria against environmental stresses, host immune responses and can confer resistance to some antimicrobial compounds [30]. Over one hundred distinct capsule types and their corresponding genomic organization have been reported in *A. baumannii* [31]. A zoom on this

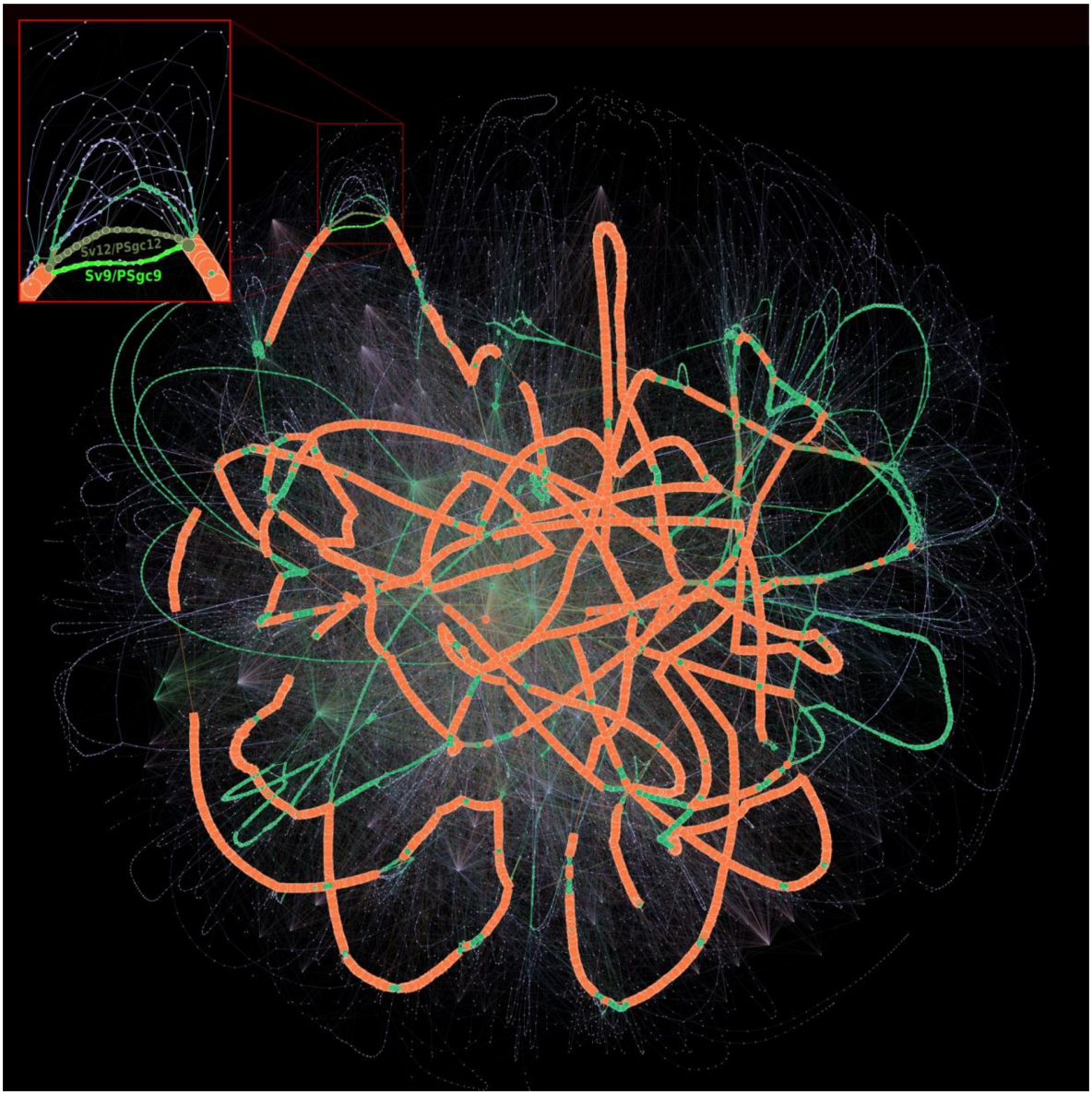


Fig 2. Partitioned pangenome graph of 3 117 *Acinetobacter baumannii* genomes. This partitioned pangenome graph of PPanGGOLiN displays the overall genomic diversity of 3 117 *Acinetobacter baumannii* strains from GenBank. Edges correspond to genomic colocalization and nodes correspond to gene families. The thickness of the edges is proportional to the number of genomes sharing that link. The size of the nodes is proportional to the total number of genes in each family. The edges between persistent, shell and cloud nodes are colored in orange, green and blue, respectively. Nodes are colored in the same way. The edges between gene families belonging to different partitions are shown in mixed colors. For visualization purposes, gene families with less than 20 genes are not shown on this figure although they comprise 84.68% of the nodes (families mostly composed of a single gene). The frame in the upper left corner shows a zoom on a branching region where multiple alternative shell and cloud paths are present in the species. This region is involved in the synthesis of the major polysaccharide antigen of *A. baumannii*. The two most frequent paths (Sv12/PSgc12 and Sv9/PSgc9) are highlighted in khaki and fluo green. The Gephi software (https://gephi.org) [32] with the ForceAtlas2 algorithm [33] was used to compute the graph layout with the following parameters: Scaling = 8000, Stronger Gravity = True, Gravity = 4.0, Edge Weight influence = 1.3.

https://doi.org/10.1371/journal.pcbi.1007732.g002

region of the graph shows a wide variety of combinations of genes for the synthesis of capsular polysaccharides. Based on the 3 117 *A. baumannii* genomes available in GenBank, we detected 229 different paths, sharing many common portions, but only a few are conserved in the species (only 24 paths are covered by more than 10 genomes). Among them, two alternative shell

paths seem to be particularly conserved (from the *gnaA* to the *weeH* genes in the figure 3 of [31]). Based on the nomenclature of [31], one (colored in khaki green in the Fig 2) corresponds to the serovar called PSgc12, contains 14 gene families of the shell genome and is fully conserved in 581 genomes. The other (colored in fluo green in the Fig 2) corresponds to the serovar PSgc9 (equivalent to PSgc7), contains 11 gene families of the shell genome and is fully conserved in 408 genomes. This analysis illustrates how the partitioned pangenome graph of PPanGGOLiN can be useful to study the plasticity of genomic regions. Thanks to its compact structure in which genes are grouped into families while preserving their genomic neighborhood information, it summarizes the diversity of thousands of genomes in a single picture and allows effective exploration of the different paths among regions or genes of interest.

Analyses of the most represented species in databanks

We used PPanGGOLiN to analyze all prokaryotic species of GenBank for which at least 15 genomes were available. This is the minimal number of genomes we recommend to ensure a relevant partitioning. The quality of the genomes was evaluated before their integration in the graph to avoid taxonomic assignation errors and contamination that can have a major impact on the analysis of pangenomes (see Materials and methods). This resulted in a dataset of 439 species pangenomes, whose metrics are available in S1 File. We focused our analysis on the 88 species containing at least 100 genomes (Fig 3). This data was used for in-depth analysis of persistent and shell genomes (see the two next sections). Proteobacteria, Firmicutes and Actinobacteria are the most represented phyla in this dataset and comprise a variety of species, genome sizes and environments. In contrast, Spirochaetes, Bacteroidetes and Chlamydiae phyla are represented by only one or two species (*Leptospira interrogans*, *Bacteroides fragilis*, *Flavobacterium psychrophilum* and *Chlamydia trachomatis*). For each species, we computed the median and interquartile range of persistent, shell and cloud families in the genomes. As expected, we observed a large variation in the range of these values: from pathogens with reduced genomes such as *Bordetella pertussis* or *C. trachomatis* which contain only a small fraction of variable gene families (less than ≈5% of shell and cloud genomes) to commensal or environmental bacteria such as *Bifidobacterium longum* and *Burkholderia cenocepacia* whose shell represents more than ≈35% of the genome. Furthermore, for a few species the number of estimated partitions (K) is greater than 3 (11 out of 88 species), especially for those with a higher fraction of shell genome. Hence, our method provides a statistical justification for the use of three partitions as a default in pangenome analyses, while indicating that species with large shell content might be best modeled using more partitions (see 'Shell structure and dynamics' section).

Estimation of the persistent genome in comparison to the soft core approach

To demonstrate the added value of PPanGGOLiN, we compared our statistical method to a classical approach where persistent genes are those present in at least 95% of the genomes (generally called the soft core approach). Indeed, this threshold is very often used in pangenomic studies probably because it is the default parameter in Roary [34] which is to date the most cited software to build bacterial pangenomes. In the 88 studied species, the number of persistent gene families is greater than or equal to the soft core with an average of 11% (SD = 9%) of additional families (see Fig 3 and S1 File). Furthermore, persistent gene families include those of the soft core with the exception of very few gene families (12 families in total for all studied species). The gene family frequencies in each of the 88 pangenomes are available in S1 Fig. For four species, *Pseudomonas stutzeri*, *Clostridium perfringens*, *Clostridium*

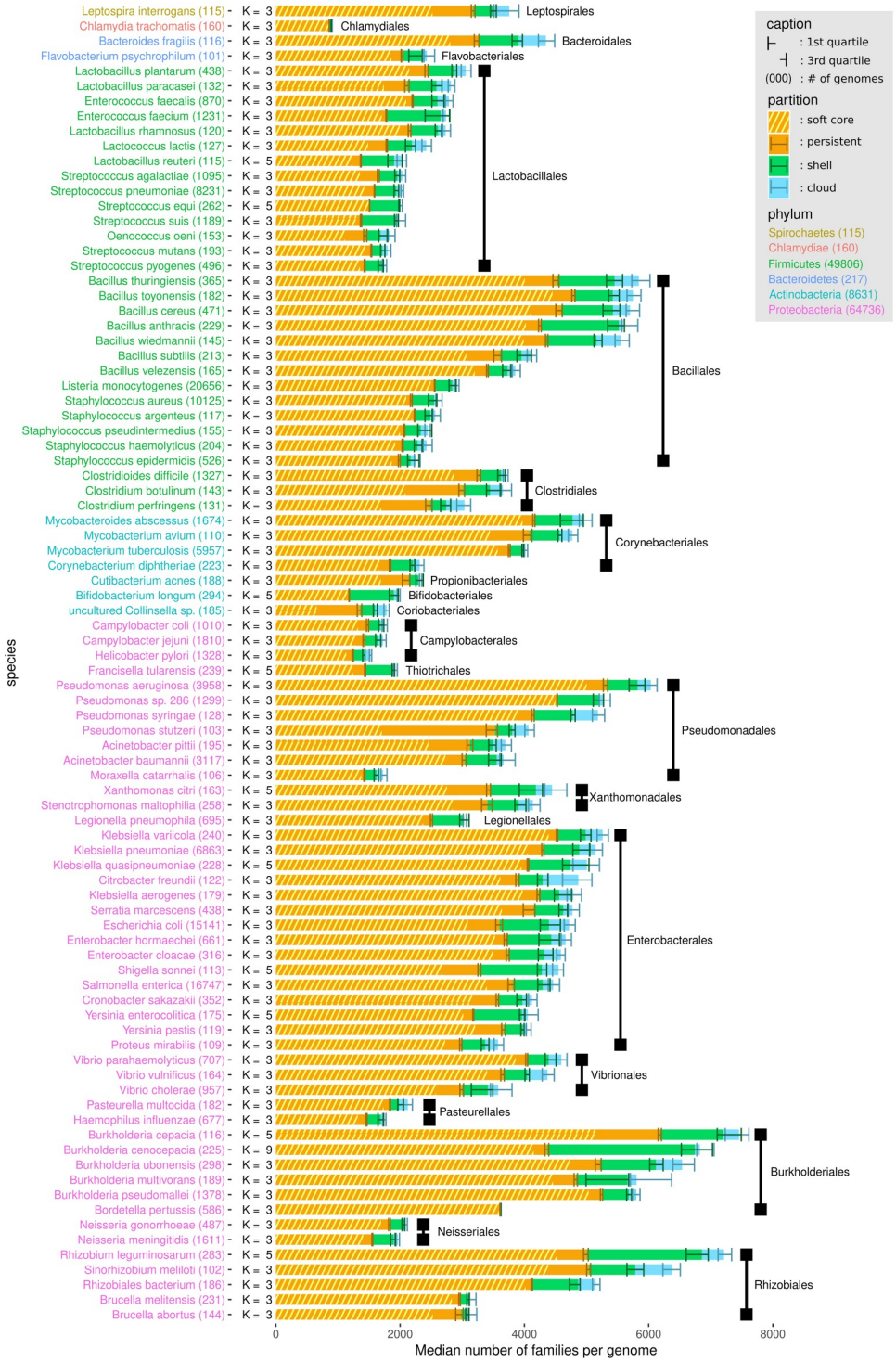


Fig 3. Distribution of PPanGGOLiN partitions in the genomes of the most represented species in GenBank. Each horizontal bar shows the median number of gene families per genome among the different PPanGGOLiN partitions (persistent, shell and cloud) in the 88 most represented species in GenBank (having at least 100 genomes). The error bars represent the interquartile ranges. Hatched areas on the persistent genome bars show the median number of gene families for the soft core (⩾95% of presence). The species names are colored according to their phylum and sorted by taxonomic order and then by decreasing cumulative bar size. Next to the species names, the number of genomes is indicated in brackets and the number of partitions (*K*) that was automatically determined by PPanGGOLiN is also shown.

https://doi.org/10.1371/journal.pcbi.1007732.g003

botulinum and *Colinsella sp*., the size of the soft core genome is unexpectedly small and represents less than 55% of the genomes whereas it is above 75% for the PPanGGOLiN persistent. For the first three species, this could be due to sampling effects and species heterogeneity. For the last one (*Colinsella sp*.), this could be explained by the fact that the species is made of incomplete genomes from metagenomes (i.e. MAGs) that were submitted as complete genomes in GenBank.

For an in-depth comparison of these approaches, we performed multiple resamplings of the genome dataset for each species in order to measure the variability of the pangenome metrics and the impact of genome sampling according to an increasing number of genomes considered in the analyses (hereafter called rarefaction curves, see Materials and methods for more details and S2 Fig as an example for *Lactobacillus plantarum*). These rarefaction curves indicate whether the number of families tends to stabilize, increase or decrease. To this end, the curves were fit with the Heaps' law where γ represents the growth tendency [35] (hereafter called γ-tendency). The persistent component of a pangenome is supposed to stabilize after the inclusion of a certain number of genomes, which means it has a γ-tendency close to 0. In addition, interquartile range (IQR) areas along the rarefaction curves were computed to estimate the variability of the predictions in relation to the sampling. Small IQR areas mean that the predictions are stable and resilient to sampling. Using these metrics, the PPanGGOLiN predictions of the persistent genome were evaluated in comparison to the soft core approach.

We observed that the γ-tendency of the PPanGGOLiN persistent is closer to 0 than that of the soft core approach (mean of absolute γ-tendency = 9.1e-3 versus 2.5e-2, P = 1.5e-9 with a one-sided paired 2-sample Student's t-test) with a lower standard deviation error too (mean = 5.3e-04 versus 2.1e-03, P = 9.5e-11 with one-sided paired 2-sample Student's t-test) (see Fig 4 and S1 File). A major problem of the soft core approach is that the γ-tendency is high for many species (32 species have a γ-tendency above 0.025), suggesting that the size of the persistent genome is not stabilized and tends to be underestimated. Besides, the IQR area of the PPanGGOLiN prediction is far below the one of the soft core genome (mean = 4906.6 versus 11645.9, P = 8.9e-07 with a unilateral paired 2-sample Student's t test). It can be partially explained because the threshold used in the soft core method induces a 'stair-step effect' along the rarefaction curves depending on the number of genomes sampled. This is illustrated on S2 Fig showing a step every 20 genomes (i.e. corresponding to $20 = \frac{100}{100-95}$ where 95% is the threshold of presence used) on the soft core curve of *L. plantarum*. We found a total of 20 species having atypical values of γ-tendency (absolute value above 0.05) and/or IQR area (above 15 000) for the soft core and only 2 species for the persistent genome of PPanGGOLiN, which are *Bacillus anthracis* and *Burkholderia cenocepacia*. For *B. cenocepacia*, it could be explained by the high heterogeneity of its shell (see next section), which is made of several partitions and complicates its distinction from the persistent genome during the process of partitioning. For *Bacillus anthracis*, the source of variability to define the persistent genome is a result of an incorrect taxonomic assignation in GenBank of about 17% of the genomes that are, according to the Genome Taxonomy DataBase (GTDB) [36], actually *B. cereus* or *B. thuringiensis*. This issue was not detected by our taxonomy control procedure because these species are at the boundary of the conspecific genomic distance threshold used (see Materials and methods). Some of persistent gene families of *bona fide B. anthracis* may therefore shift between persistent or shell partitions depending on the resampling. Excluding these misclassified genomes, we predicted a larger persistent genome than the one of the initial full set of genomes (about a thousand gene families more) with a γ-tendency much closer to 0 (-0.017 versus a γ-tendency of 0.036 for the soft core genome) and a lower IQR area (8367.0 vs 32167.1). Altogether, these results suggest that our approach provides a more robust partitioning of gene families in the

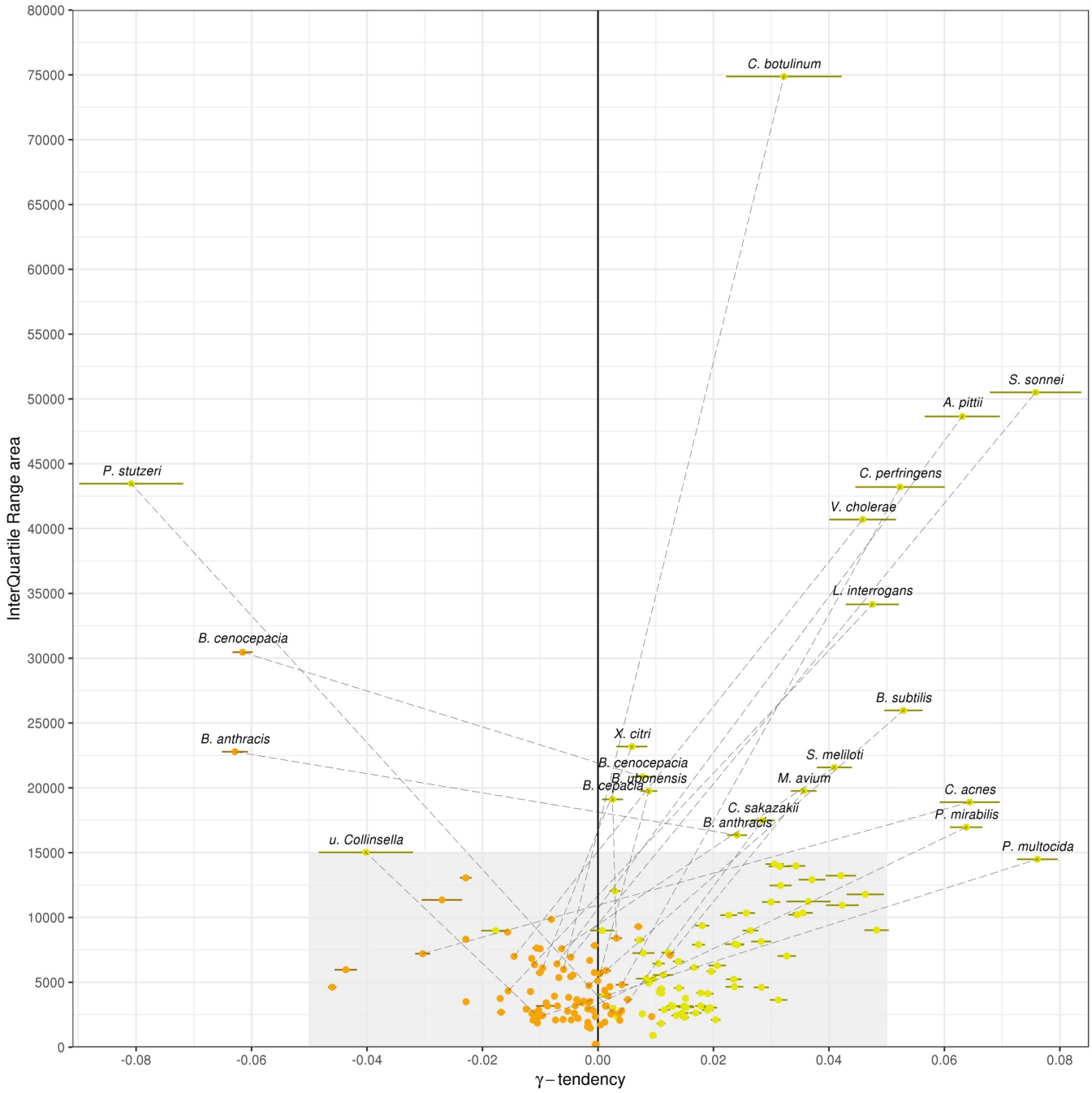


Fig 4. γ-tendencies and IQR areas of the persistent and the soft core rarefaction curves. Each of the 88 most abundant species in GenBank are represented by two points: orange points correspond to the PPanGGOLiN persistent values and yellow points to the ones of the soft core (⩾95% of presence). A dashed line connects the 2 points if either the soft core or the persistent values are not in the range of the grey area ($-0.05 \leqslant \gamma \leqslant 0.05$ and $0 \leqslant IQR_{area} \leqslant 15000$). The colored horizontal bars show the standard errors of the fitting of rarefaction curves via the Heaps' law.

https://doi.org/10.1371/journal.pcbi.1007732.g004

persistent genome than the use of arbitrary thresholds. Indeed, the statistical method behind PPanGGOLiN uses directly the information of the gene family P/A whereas the soft core is based only on frequency values. PPanGGOLiN can then classify families with similar frequencies in different partitions by distinguishing them according to their pattern of P/A in the

matrix and their genomic neighborhood. The main drawback of using family frequency to partition pangenomes is that even if it was possible to determine the best threshold for each species it would still not take into account that some persistent gene families may have atypically low frequency. This may be due to high gene losses in the population or technical reasons like belonging to a genomic region that is difficult to assemble (i.e. genes that are missing or fragmented in draft genome assemblies).

Shell structure and dynamics

Two types of pangenome evolution dynamics are generally distinguished: open pangenomes and closed ones [1, 2, 35]. From rarefaction curves, the dynamics of pangenomes can be assessed using the γ-tendency of a Heaps' law fitting (see Materials and methods). A low γ-tendency means a rather closed pangenome whereas a higher γ-tendency means a rather open pangenome. A closed pangenome rigorously means a stabilized pangenome and we found no species obeying this strict criterion (that is to say $\gamma = 0$). This suggests that instead of using binary classifications for pangenomes, it is more useful to quantify the degree of openness of pangenomes given the flux of horizontal gene transfers and gene loss [7]. We computed rarefaction curves for the 88 studied species and determined the γ-tendency for different pangenome components (see S1 File and S3 Fig). The distribution of γ values of the PPanGGOLiN shell genome shows a greater amplitude of values than the other components of the pangenome such as the whole pangenome or the accessory component. This indicates that the main differences in terms of genome dynamics between species seem to reside in the shell genome.

As expected, we found a positive correlation (Spearman's $\rho = 0.46$, P = 8.2e-06) between the total number of shell gene families in a species and the γ-tendency of the shell (S4 Fig). This means that species with high γ-tendency do accumulate genes that are maintained and exchanged in the population at relatively low frequencies, suggesting they may be locally adaptive. More surprisingly, although one could expect that larger genomes have a larger fraction of variable gene repertoires, the fraction of shell and cloud genes per genome does not correlate with the genome size (Spearman's $\rho = 0.007$, P = 0.95, Fig 5). The results remain qualitatively similar when analyzing the shell or the cloud separately (see S5 and S6 Figs). During this analysis, we noticed that, among host-associated bacteria with relatively small genomes (between ≈2000 and ≈3000 genes), three species (*Bifidobacterium longum*, *Enterococcus faecium* and *Streptococcus suis*) have a high fraction of shell genes ($>$ 28%) but low shell γ-tendency. Two of them (*B. longum* and *E. faecium*) are found in the gut of mammals and the third (*S. suis*) in the upper respiratory tract of pigs. They differ from other host-associated species in our dataset that are mainly human pathogens (e.g. bacteria of the genus *Corynebacterium*, *Neisseria*, *Streptococcus*, *Staphylococcus*) and have a low fraction of shell genes ($<$ 20%). It is possible that these three species have specialized in their ecological niches while maintaining a large and stable pool of shell genes for their adaptation to environmental stress. Further analysis would be required to confirm this hypothesis.

We then investigated the importance of the phylogeny of the species on the patterns of P/A of the shell gene families (shell structure). To this end, Spearman's rank correlations were computed between a Jaccard distance matrix generated on the basis of patterns of P/A of the shell gene families and a genomic distance obtained by Mash pairwise comparisons between genomes [37]. Mash distances were shown to be a good estimate of evolutionary distances for closely related genomes [38]. This correlation was examined in relation to the fraction of gene families that are part of the shell genome for each species (Fig 6). We observed that species with a high fraction of shell ($>$ 20% of their genome) have a shell structure that is mainly explained by the species phylogeny (i.e. shell P/A are highly correlated with genomic distances,

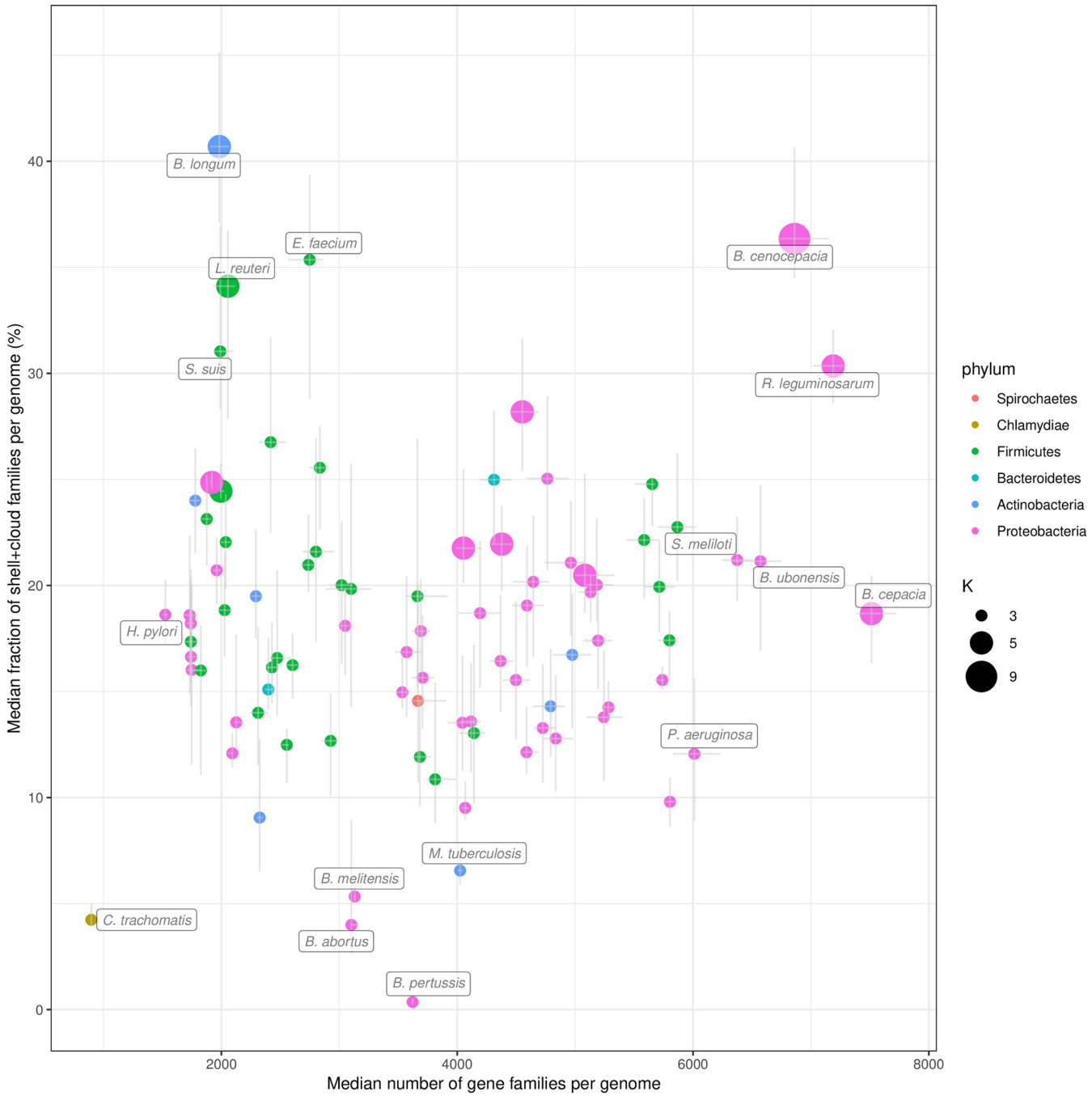


Fig 5. Fraction of the variable (shell + cloud) families per genome compared to the number of gene families. The results for the 88 most abundant species in GenBank are represented. The error bars show the interquartile ranges of the two variables. The points are colored by phylum and their size corresponds to the number of partitions (*K*) used.

https://doi.org/10.1371/journal.pcbi.1007732.g005

Spearman's $\rho > 0.75$). In addition, PPanGGOLiN predicts a number of partitions (*K*) for these species often greater than 3. Hence, their shell is more heterogeneous between subclades and becomes structured in several partitions whereas for species with a single shell partition the shell is less structured, possibly indicating many gene exchanges between strains from different

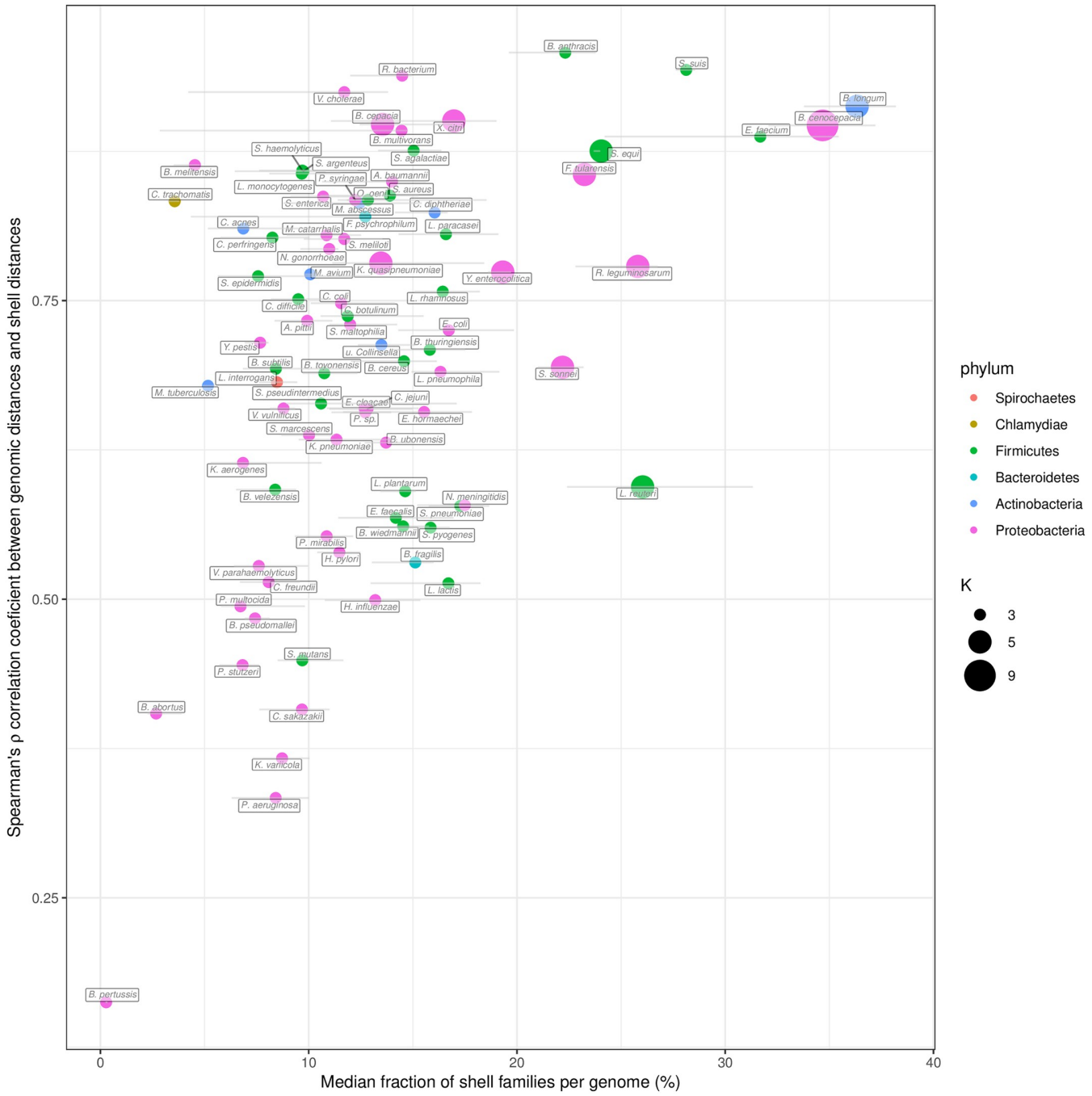


Fig 6. Spearman's *ρ* correlation coefficients between the shell genome presence/absence patterns and the MASH genomic distances compared with the shell fraction per genome. The results for the 88 most abundant species in GenBank are represented. The error bars show the interquartile ranges of the shell fraction. The points are colored by phylum and their size corresponds to the number of partitions (*K*) used.

https://doi.org/10.1371/journal.pcbi.1007732.g006

lineages. Among the nine species with a large shell genome (excluding *B. anthracis* due to taxonomic assignation errors), only two of them (*Shigella sonnei* and *Lactobacillus reuteri*) showed a relatively low correlation of their shell structure with the phylogeny (Fig 6). For *S. sonnei*, this could be explained by a high number of gene losses in the shell of this species that result

from convergent gene loss mediated by insertion sequences (preprint: [39]). For *L. reuteri*, these bacteria colonize the gastrointestinal tract of a wide variety of vertebrate species and have diversified into distinct phylogenetic clades that reflect the host where the strains were isolated, but not their geographical provenance [40]. As illustrated in S7 Fig, the shell of *L. reuteri* shows patterns of P/A that are only partially explained by the species phylogeny. Indeed, we observed clusters of families present across strains from distinct lineages that could contain factors for adaptation to the same host. In contrast, the shell structure of *B. longum* strongly depends on phylogenetic distances showing a clear delineation of adult and infant strains that have specialized into two subspecies (see S8 Fig).

We would like to stress the importance of the shell in the study of the evolutionary dynamics of bacteria. The shell content reflects the adaptive capacities of species through the acquisition of new genes that are maintained in the population. We found that the proportion of shell genes does not increase with the genome size. Instead, the shell accounts for a large fraction of the genomes of species when it is structured in several partitions. We can assume that those species are made of non-homogeneous subclades harboring specific shell genes which contribute to the specialization of the latter. Finally, it could be of interest to associate phenotypes to patterns of shell families that co-occur in different lineages independently of the phylogeny.

Analysis of Metagenome-Assembled Genomes in comparison with isolate genomes

The graph approach should make our tool robust to gaps in genome data, making it a useful tool to analyze pangenomes obtained from MAGs. To test this hypothesis, we built the pangenomes of the Species-level Genome Bins (SGBs, clusters of MAGs that span a 5% genetic diversity and are assumed to belong to the same species) from the recent paper of Pasolli *et al.* [41]. This study agglomerated and consistently built 4 930 SGBs (154 723 MAGs) from 13 studies focussed on the composition of the human microbiome. We skipped the quality control step (already performed by the authors), and computed the pangenomes following the procedure we used for the GenBank species. The only parameter which differs is the *K* value which is set to 3 as the detection of several shell partitions is difficult for MAGs because of their incompleteness. To make the comparison with GenBank species, SGBs were grouped according to their estimated species taxonomy (provided by the supplementary table S4 of [41]). In this table, we noticed potential errors in the taxonomic assignation of two species (*Blautia obeum* and *Chlamydia trachomatis* corresponding to SGBs 4844 and 6877, respectively) and thus excluded them from the analysis. Keeping the same constraint as previously, only species with at least 15 genomes in both MAGs and GenBank were used for the comparison. A total of just 78 species (corresponding to 151 SGBs) could be analyzed as a lot of microbiome species are laborious to cultivate and thus less represented in databanks (see S2 File). Then, we compared the MAG pangenome partitions predicted by PPanGGOLiN with those obtained with GenBank genomes. To perform this, we aligned MAG and GenBank families for each species and computed the percentage of common families for each partition (see Materials and methods for details and S2 File for detailed results).

We observed that the size of the estimated persistent genome of MAGs is similar to the one of GenBank genomes for most species (Fig 7). In 55 out of the 78 species, the absolute fold change of persistent size is less than 1.2 and 90% (SD = 5%) of its content is common between MAGs and GenBank genomes. The 23 other species with more important differences showed smaller persistent genomes with only 60% (SD = 15%) of the persistent genome of GenBank being found in MAGs. For these species, the PPanGGOLiN method missed a fraction of the persistent genome due to the incompleteness of MAGs. Indeed, in such cases, the

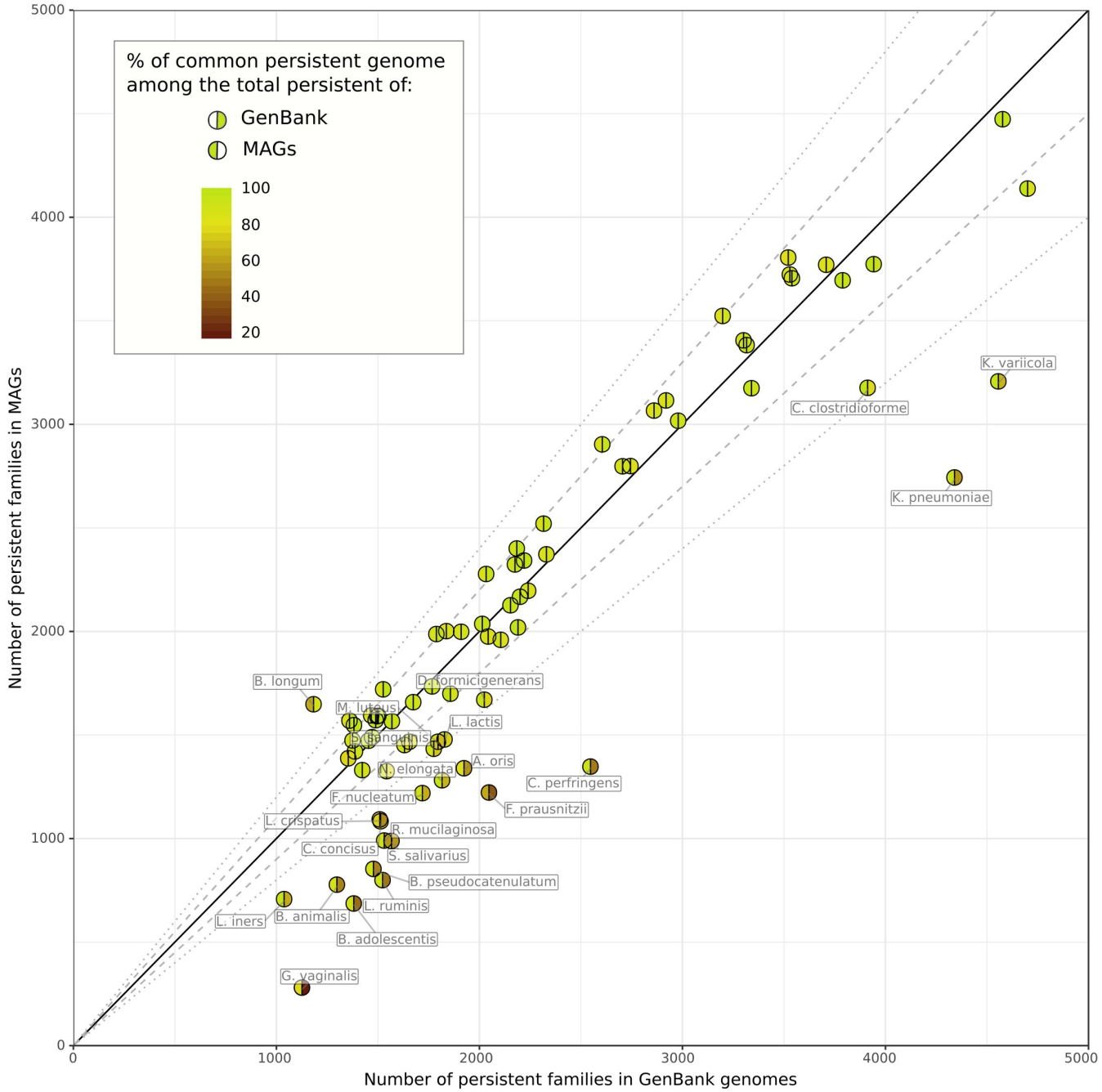


Fig 7. Illustration of the persistent genome overlaps between GenBank genomes and MAGs. Results for 78 species are represented. The colors of the hemispheres provide the percentage of common persistent gene families among the total persistent of MAGs (left hemisphere) or GenBank genomes (right hemisphere). The solid, dashed and dotted lines indicate the identity, a fold change of 1.1 and a fold change of 1.2 between the persistent genome sizes.

https://doi.org/10.1371/journal.pcbi.1007732.g007

missing gene families are mostly classified in the shell of the MAGs which contains 32% (SD = 11%) of the GenBank persistent families. Nevertheless, 89% (SD = 9%) of the MAG persistent families match the GenBank ones, meaning that PPanGGOLiN correctly assigned persistent families for MAGs even if the persistent genome of these 23 species is incomplete.

However, two species, *Bifidobacterium longum* and *Faecalibacterium prausnitzii*, have less than 75% of their MAG persistent families in common with GenBank ones. For *B. longum*, this could be explained by the fact that the MAGs were obtained mostly from human adult samples while this species in databanks are from a broader host range (infants and pigs). It means that the MAG persistent might contain additional genes related to host-specificity. As a matter of fact, 412 gene families from the MAG persistent (25% of the total MAG persistent) are found in the GenBank shell which supports our hypothesis. For *F. prausnitzii*, the differences might be explained by a poor estimation of the persistent using GenBank data due to the low number of considered genomes (17 genomes versus 4232 MAGs). As expected, the soft core (based on the usual threshold of 95% presence) is unrealistically low in the MAG species with only ≈98 gene families on average and only 4 species out of 78 having more than 500 families classified in the soft core (see S2 File). Hence, the soft core approach is not well adapted to the analysis of MAGs. Furthermore, using lower thresholds of presence is not adequate because defining a unique threshold for all the families misses the heterogeneity of gene family presence in MAGs.

To explore the diversity within the pangenome of each species, we compared the shell of GenBank genomes and MAGs for the 55 ones with similar persistent genomes. Interestingly, we observed for all the 55 species only a partial overlap between the MAGs and GenBank shells (see S9 Fig). Indeed, as the MAGs are obtained only from a specific environment (i.e. the human microbiome), the diversity of GenBank is not fully captured by MAGs. It is especially the case for most of the Firmicutes and Proteobacteria. Conversely, most of the MAGs of Bacteroidetes phylum cover more than half of GenBank diversity while containing a large fraction of shell genes that are lacking in the shell of isolate genomes (i.e. less than 45% of the families are represented in the shell of GenBank). As already reported by Pasolli *et al.* [41], this confirms that the MAGs considerably improve the estimate of the genetic diversity of Bacteroidetes which are key players in the gut microbiome.

In summary, we have shown that PPanGGOLiN is able to provide an estimation of the persistent genome even using MAGs, which may miss significant numbers of genes and be contaminated by fragments from other genomes. This is especially the case for the accessory genome because its assembly coverage and nucleotide composition generally differ from those of the persistent genome making the binning of these regions more difficult. Nevertheless, PPanGGOLiN is able to find shell gene families in MAGs bringing new genes that may be important for species adaptation in the microbiome. Hence, it enables further analyses, even for uncultured species lacking reference genomes, such as the reconstruction of the core metabolism from the persistent genome to predict culture media or the study of the landscape of horizontally transferred genes within species.

Conclusion

We have presented here the PPanGGOLiN method that enables the partitioning of pangenomes in persistent, shell and cloud genomes using a gene family graph approach. This compact structure is useful to depict the overall genomic diversity of thousands of strains highlighting variable paths made of shell and cloud genes within the persistent backbone. The statistical model behind PPanGGOLiN makes a more robust estimation of the persistent genome in comparison to classical approaches based on gene family frequencies in isolate genomes and also in MAGs. The definition of shell partitions based on statistical criteria allowed us to understand genome dynamics within species. We observed different patterns of shell with regard to phylogeny that may suggest different adaptive paths for the diversification of the species. It should be stressed that genome sampling is one of the main limitations of pangenome

studies and can therefore influence PPanGGOLiN partitioning especially for the shell genome. An improvement in the method could be to normalize the data to remove sampling bias. But as suggested by Brockhurst *et al.* [42], this issue should first be examined from a biological perspective by collecting and analyzing genomes from ecologically coherent microbial populations or ecotypes.

Future applications of PPanGGOLiN could include the prediction of genomics islands within the shell and cloud genomes. A first version of this application (Bazin *et al.*, in preparation) is already integrated in the MicroScope genome analysis platform [43]. Next, it would be interesting to determine the architecture of these variable regions by predicting conserved gene modules using information on the occurrence of families and their genomic neighborhood in the pangenome graph. Regarding metagenomics, pangenome graphs of PPanGGOLiN could be used as a reference (i.e. instead of individual genomes) for species quantification by mapping short or long reads on the graph to compute the coverage of the persistent genome. Indeed, each gene families of the partitioned pangenome graph could embed a variation graph as an alignment template [19]. Moreover, coverage variation in the shell or cloud genomes could allow the detection of strain-specific paths in the graph that are signatures of distinctive traits within microbiotes.

To conclude, the graph-based approach proposed by PPanGGOLiN provides an effective basis for very large scale comparative genomics and we hope that drawing genomes on rails like a subway map may help biologists navigate the great diversity of microbial life.

Materials and methods

To explain the partitioning of pangenomes, we first need to describe the method based on the P/A matrix only (BinEM) and then the method built upon it that uses the pangenome graph to improve the partitioning (NEM).

Modeling the P/A matrix via a multivariate Bernoulli Mixture Model

PPanGGOLiN aims to classify patterns of P/A of gene families into K partitions ($K \in \mathbb{N}$; $K \geqslant 3$). Input data consists of a binary matrix X in which a x_{ij} entry is 1 if family i is present in a genome j and 0 otherwise (Fig 1) where $1 \leqslant i \leqslant F$ in each of the F gene families and $1 \leqslant j \leqslant N$ in each of the N genomes. A first approach for partitioning the data relies on a multivariate Bernoulli Mixture Model (BMM) estimated through the Expectation-Maximization (EM) algorithm [44] (named the BinEM method). The number of partitions K may be greater than 3 (persistent, shell and cloud) due to the possible presence of antagonist P/A patterns among the different strains of a species. Therefore, two of the partitions will correspond to the persistent and cloud genome and a number of $K - 2$ partitions will correspond to the shell genome. The value of K can be either provided by the user or determined automatically (see next section).

In the BMM, the matrix comprises data vectors $X_i = (x_{ij})_{1 \leqslant j \leqslant N}$ describing P/A of families, which are assumed to be independent and identically distributed with a mixture distribution given by:

$$P(X_i = (x_{ij})_{1 \leqslant j \leqslant N}) = \sum_{k=1}^{K} \pi_k \prod_{j=1}^{N} \epsilon_{kj}^{|x_{ij} - \mu_{kj}|} (1 - \epsilon_{kj})^{1 - |x_{ij} - \mu_{kj}|}$$

where $\pi = (\pi_1, \ldots, \pi_k, \ldots, \pi_K)$ denotes the mixing proportions satisfying $\pi_k \in [0, 1]$; $(\sum_{k=1}^{K} \pi_k) = 1$ and where π_k is the unknown proportion of gene families belonging to the k^{th} partition. Moreover, $\mu_k = (\mu_{kj})_{1 \leqslant j \leqslant N} \in \{0; 1\}^N$ are the centroid vectors of P/A of the k^{th} partition representing the most probable binary states and $\epsilon_k = (\epsilon_{kj})_{1 \leqslant j \leqslant N} \in \left[0, \frac{1}{2}\right]^N$ are the unknown

vectors of dispersion around μ_k. The default values of the dispersion vector ϵ_k associated to each centroid vector μ_k are constrained to be identical for all the ϵ_{kj} of a specific k partition (for all the genomes of a specific partition) in order to avoid over-fitting but it is possible to release this constraint. The parameters of this model, as well as corresponding partitions, are estimated by the EM algorithm. To speed up the computation of the EM algorithm, a heuristic is used to initialize the BMM parameters in order to converge to a relevant partitioning using fewer EM-steps. This heuristic consists in setting π_k with equiprobable proportions equal to $1/K$ while the ϵ_{kj} and μ_{kj} parameters are initialized triangularly.

Given $s = 1/\lceil K/2 \rceil$, the triangular initialization consists of:

$$\{\mu_{kj}\}_{1 \leqslant k \leqslant K/2, 1 \leqslant j \leqslant N} = 1$$
$$\{\mu_{kj}\}_{K/2 < k \leqslant K, 1 \leqslant j \leqslant N} = 0$$
$$\{\epsilon_{kj}\}_{1 \leqslant k \leqslant K/2, 1 \leqslant j \leqslant N} = s \cdot k$$
$$\{\epsilon_{kj}\}_{K/2 < k \leqslant K, 1 \leqslant j \leqslant N} = s \cdot (K - k + 1)$$

An interesting consequence of this initialization is that the persistent genome will be the first partition ($k = 1$) while the cloud genome will correspond to the last partition ($k = K$). This particular initialization solves the classical label switching problem in our context.

Partitioning of the P/A matrix

To perform the partitioning of the P/A matrix, each gene family i must be allocated to a single partition. The variables $\{Z_i\}_{1 \leqslant i \leqslant F}$ with a state space $\{1, \ldots, K\}$ indicate the partition to which each gene family i belongs. Therefore, once the NEM parameters are optimized, the method automatically assigns the gene families to their most probable partition z_i according to the model if their estimated posterior probability is above 0.5. If no partition can be assigned in this way, then the gene family is assigned to the shell (partition with intermediate frequency).

Selection of the optimal number of partitions (K)

To determine the optimal K, named $\hat{K}$, the algorithm runs multiple partitionings with increasing values of K. After a few steps of the EM algorithm (10 steps by default), the Integrated Completed Likelihood (*ICL*) [45] is computed for each K. The *ICL* corresponds to the Bayesian Information Criterion (*BIC*) [46] penalized by the estimated mean entropy and is calculated as:

$$ICL(K) = BIC(K) - \sum_{k=1}^{K}\sum_{i=1}^{F} p(z_i \mid X, \hat{\theta}, k) \log\left(p(z_i \mid X, \hat{\theta}, k)\right); \forall p(z_i \mid X, \hat{\theta}, k) > 0$$

and

$$BIC(K) = \log \mathbb{P}_K(X \mid \hat{\theta}) - 1/2 dim(K) \log F$$

where $\log \mathbb{P}_K(X \mid \theta)$ is the data log-likelihood under a multivariate BMM with K partitions and $\theta = (\{\pi_k\}_{1 \leqslant k \leqslant K}, \{\mu_{kj}\}_{1 \leqslant k \leqslant K, 1 \leqslant j \leqslant N}, \{\epsilon_{kj}\}_{1 \leqslant k \leqslant K, 1 \leqslant j \leqslant N})$. This log-likelihood can be calculated as follows:

$$\log \mathbb{P}_K(X \mid \theta) = \sum_{i=1}^{F} \log\left(\sum_{k=1}^{K} \pi_q \prod_{j=1}^{N} \epsilon_{kj}^{|x_{ij} - \mu_{kj}|} (1 - \epsilon_{kj})^{1 - |x_{ij} - \mu_{kj}|}\right)$$

Moreover, $\hat{\theta}$ is the maximum likelihood estimator (approximated through the BinEM

algorithm) and $dim(K)$ is the dimension of the parameter space for this model. Here, $dim(K) = K(N + 2)$ if the dispersion vector ϵ_k associated to each centroid vector μ_k is constrained to be identical for all the ϵ_{kj} of a specific k partition and $dim(K) = K(2N + 1)$ if the dispersion vector ϵ_k is free. Relying on this criterion, the best number of partitions is selected as $\hat{K} = \arg\min_K ((1 - \delta_{ICL})ICL(K))$ where δ_{ICL} is a sufficiently small margin to avoid choosing a too high K value that would provide no significant gain compared to a lower value of K (by default $\delta_{ICL} = 0.05 \times (max(ICL) - min(ICL))$).

Generation of the pangenome graph

PPanGGOLiN uses a graph-based representation to store and visualize pangenomes. In this graph, the nodes correspond to gene families and the edges to genetic contiguity (i.e. genes that are direct neighbors in a genome). Two nodes are connected if the corresponding gene families contain at least one pair of genes that are adjacent in a genome. Edges are labeled with the corresponding genome identifiers and weighted by the proportion of genomes sharing that link. This process results in a pangenome graph (see Fig 2 as an example).

Formally, a pangenome graph $G = (V, E)$ is a graph having a set of vertices $V = \{(v_i)_{(1 \leqslant i \leqslant F)}\}$ where F is the number of gene families in the pangenome associated with a set of edges $E = \{e_{i \sim i'}\} = \{(v_i, v_{i'})\}$, $v_i \in V$, $v_{i'} \in V$ where the couple of vertices $(v_i, v_{i'})$ are gene families having their genes $(v_{i,j}, v_{i',j})$ adjacent on the genome j and where the function $countNeighboringGenes$ $(v_i, v_{i'})$ counts the adjacency occurrences in the N genomes. Each edge $\{e_{i \sim i'}\}$ has a weight $w_{i \sim i'}$ where $w_{i \sim i'} = \frac{1}{N} \sum_{j=1}^{N} countNeighboringGenes(v_{i,j}, v_{i',j})$.

Partitioning via Neighboring Expectation-Maximization

From the graph previously described, the neighborhood information of the gene families is used to improve the partitioning results. Indeed, the BinEM approach described above is extended by combining the P/A matrix X with the pangenome graph G. This relies on a hidden Markov Random Field (MRF) model whose graph structure is given by G. In this model, each node belongs to some unobserved (hidden) partitions which are distributed among gene families according to a MRF which favors two neighbors to be more likely classified in the same partition. Conditional on this hidden structure, the binary vectors of P/A are independent and follow a multivariate Bernoulli distribution with proportion vectors depending on the associated partition. This approach is called NEM, as it relies on the Neighboring Expectation-Maximization algorithm [47–49]. As such, NEM tends to smooth the partitioning by grouping gene families that have a weighted majority of neighbors belonging to the same partition. The previously introduced latent variables $\{Z_i\}_{1 \leqslant i \leqslant F}$, that indicate the partition to which each gene family belongs are now distributed according to a MRF. More precisely, they have the following Gibbs distribution:

$$\mathbb{P}(\{Z_i\}_{1 \leqslant i \leqslant F}) = W_\beta^{-1} \exp\left(\sum_{i=1}^{F} \sum_{k=1}^{K} \pi_k 1_{Z_i = k} + \beta \frac{F}{\sum_{i \sim i'} w_{i \sim i'}} \sum_{i \sim i'} w_{i \sim i'} 1_{Z_i = Z_{i'}}\right)$$

where 1_A is the indicator function of event A and the second sum concerns every pair $(i \sim i')$ of neighbor gene families. The parameter $\beta \geq 0$ corresponds to the coefficient of spatial regularity. The $\frac{F}{\sum_{i \sim i'}^{E} w_{i \sim i'}}$ is a corrector term ensuring that the strength of the spatial smoothing is balanced regardless of the number of gene families. Indeed, when the number of genomes (N) increases, the number of gene families (F) tends to be higher than the sum of the edge weights.

Finally,

$$W_\beta = \sum_{\{\tilde{z}_i\}\in\{1\dots K\}^F} \exp\Big(\sum_{i=1}^{F}\sum_{k=1}^{K}\pi_k 1_{\tilde{z}_i=k} + \beta\frac{F}{\sum_{i\sim i'} w_{i\sim i'}}\sum_{i\sim i'} w_{i\sim i'} 1_{\tilde{z}_i=\tilde{z}_{i'}}\Big)$$

is a normalizing constant. Note that W_β cannot be computed, due to a large number of possible configurations. The degree of dependence between elements is controlled by the parameter β. Neighboring elements will be more inclined to belong to the same group with a higher value of this parameter. Here, the data vectors $(X_i)_{1\leqslant i\leqslant F}$ are not independent anymore. However, conditional on the latent groups $(Z_i)_{1\leqslant i\leqslant F}$, they are independent and follow the multivariate Bernoulli distribution:

$$\mathbb{P}(\{X_i\}_{1\leqslant i\leqslant F}|\{Z_i\}_{1\leqslant i\leqslant F}) = \prod_{i=1}^{F}\prod_{j=1}^{N}\epsilon_{Z_i j}^{|x_{ij}-\mu_{Z_i j}|}(1-\epsilon_{Z_i j})^{1-|x_{ij}-\mu_{Z_i j}|}.$$

Many different techniques may be used to approximate the maximum likelihood estimator in the hidden MRF. NEM relies on a mean-field approximation for the distribution of the latent random variables $Z_{i1\leqslant i\leqslant F}$ conditional on the observations. It should be noted that the optimal number of partitions (K) is not determined automatically using NEM and is therefore first estimated using the BinEM approach.

Issues resulting from high-dimensional statistics and parallelization

As plenty of statistical approaches, NEM is not adapted to high dimensional settings (i.e. whenever the condition $F >> N$ is not satisfied). This can occur in pangenomics as the discovery rate of new families in the pangenome slightly decreases when new genomes are added. Mathematical solutions to this problem seem to exist [50–52] for example via the weighting of genomes (based on their respective contribution to the pangenome diversity) or via sparse partitioning methods. An improvement of NEM should include these solutions and could be a perspective of this work.

Pangenome software must be designed to scale up to thousands of genomes. NEM scales quadratically with the number of genomes and is hard to parallelize. Thus, it leads to intensive computations when thousands of genomes are included in the analysis.

Our solution to the mentioned issues is to sample the genomes in chunks and to perform multiple partitioning in parallel. Each family must be involved in at least $N_{total}/N_{samples}$ samplings and will be partitioned only if it is classified in the same partition in at least 50% of the samplings where it is present (absolute majority). If some families do not respect this condition, we continue sampling until all gene families have been partitioned. Chunks have to be large enough to be representative, therefore a size of at least 500 genomes is advised.

Analysis of isolate genomes and Metagenome-Assembled Genomes

To obtain the set of isolate genomes to be analyzed, we downloaded all archaeal and bacterial genomes (220 561 genomes) of the GenBank database at the date of the 17th of April 2019. We removed genome assemblies that do not respect quality control criteria defined by GenBank. They correspond to entries with an assembly status flag different from "status = latest" in the "assembly_status.txt" files. In addition, genomes were discarded if they had more than 1000 contigs or a L90 > 100. These filters allowed us to exclude poor quality assemblies, some of which may correspond to contaminated genomes and others to incomplete ones. For each species (identified by its NCBI species taxid), a pairwise genomic distance matrix was computed using Mash (version 2.0) [37]. To avoid redundancy, if several genomes are at a Mash

distance < 0.0001, only one was kept (the one having the lowest number of contigs). A single linkage clustering using SiLiX (version 1.2.11) [53] was then performed on the adjacency graph of the Mash distance matrix considering only distances below or equal to 0.06. This Mash distance corresponds to a 94% Average Nucleotide Identity (ANI) cutoff which is a usual value to define species [54]. Genomes that were not in the largest connected component were discarded to remove potential taxonomic assignation errors. Only species having at least 15 remaining genomes were then considered for the analysis. The list of all the GenBank assembly accessions used after filtering is available in S3 File. This dataset consists of 439 species encompassing 136 287 genomes (see S1 File). MAGs from the Pasolli *et al*. study [41] were downloaded from https://opendata.lifebit.ai/table/SGB. In this dataset, the genomes are already grouped in Species Genome Bins. These SGBs do not exactly match the GenBank taxonomy. Thus, SGBs assigned with the same species name (column "estimated taxonomy" in the supplementary table S4 of [41]) were merged to allow comparison with GenBank. SGBs that do not have a taxonomy assigned at the species level were not considered. A total of 583 species encompassing 698 SGBs and 71 766 MAGs were analyzed but only MAGs from 78 species were finally compared to GenBank genomes. To avoid introducing a bias in our analysis due to heterogeneous gene calling, GenBank annotations were not considered as they were obtained using a variety of annotation workflows. Genomes from GenBank and Pasolli *et al* were consistently annotated using the procedure implemented in PPanGGOLiN. Prodigal (version 2.6.2) [55] is used to detect the coding genes (CDS). tRNA and tmRNA genes are predicted using Aragorn (version 1.2.38) [56] whereas the rRNA are detected using Infernal (version 1.1.2) [57] with HMM models from Rfam [58]. In the case of overlaps between a RNA and a CDS, the overlapping CDS are discarded. Homologous gene families were determined using MMseqs2 (version 8-fac81) [59] with the following parameters: coverage = 80% with cov-mode = 0, minimal amino acid sequence identity = 80% and cluster-mode = 0 corresponding to the Greedy Set Cover clustering mode. PPanGGOLiN partitioning was executed on each species using the NEM approach with a parameter $\beta = 2.5$. The nodes having a degree above 10 (which is the default parameter) were not considered to smooth the partitioning via the MRF. The number of partitions (K) was determined automatically for each NCBI species using a $\delta_{ICL} = 0.05$ and iterating between 3 and 20 for the possible values of K. K was fixed at 3 for the MAG analysis. The partitioning was done using chunks of 500 genomes when there were more than 500 genomes in a species. To compare PPanGGOLiN results between MAGs and GenBank genomes for each species, the representative sequences of each MAG gene family (extracted using the mmseqs2 subcommand: "result2repseq") were aligned (using mmseqs2 "search") on those of GenBank genomes. If the best hit of the query had a sequence identity $> 80\%$ and a coverage $> 80\%$ of the target, the 2 corresponding gene families of each dataset were associated.

Rarefaction curves

To represent the pangenome evolution according to the number of sequenced genomes, a multiple resampling approach was used. For each species with at least 100 genomes, 8 rarefaction curves showing the evolution of the pangenome and the persistent, shell, cloud, soft core, soft accessory, exact core and exact accessory components were computed for sample sizes of 1 to 100 genomes randomly drawn from the set of all genomes of the species. Each sample size was analyzed using 30 different samples. For each sample, the number of partitions K is automatically determined between 3 and the K obtained on all the genomes of the species. A non-linear Least Squares Regression was performed to fit the rarefaction curves with Heaps' law $F = \kappa N^{\gamma}$ where F is the number of gene families, N the number of genomes, γ the tendency of

the evolution and κ a proportional factor [35]. Subset sizes ≤ 15 were not used for the fitting as they are sometimes too variable to ensure a good fitting. The function "scipy.optimize.curve_fit" of the Python scipy package (version 1.0.0), based on the Levenberg-Marquardt algorithm, was used to fit the rarefaction curves. For each subset size, the median and quartiles were calculated to obtain a ribbon of interquartile ranges (IQR) along the rarefaction curves. We call the area of this ribbon the IQR area (see S2 Fig as an example).

PPanGGOLiN software implementation

PPanGGOLiN was designed to be a software suite performing the annotation of the genomic sequences, building the gene families and the pangenome graph before partitioning it. Users can also provide their own annotations (GFF3 or GBFF format) and gene families. The application stores its data in a compressed HDF5 file but can also return the graph in GEXF or JSON formats and the P/A matrix with the partitioning in CSV or Rtab files (similarly to the ones provided by Roary [34]). It also generates several illustrative figures, some of which are presented in the article. PPanGGOLiN was developed in the Python 3 and C languages and is intended to be easily installable on Linux and Mac OS systems via a BioConda package [60] (see https://bioconda.github.io/recipes/ppanggolin/README.html). The code is also freely available on the GitHub website at the following address: https://github.com/labgem/PPanGGOLiN.

Supporting information

S1 Fig. Density distributions of the gene family frequencies of each partition. Results for the 88 most abundant species in GenBank are represented in addition with a global distribution of the gene family frequencies from all the species. Density values of the cloud genome above 100 (y-axis) were trimmed for visualization purpose. The dashed yellow vertical bars indicate the threshold of frequency (⩾95%) used to delimit the soft core genome.
(PDF)

S2 Fig. Evolution of the persistent, shell, soft core and exact core metrics of Lactobacillus plantarum compared to the number of genomes. The rarefaction curves represent the evolution of the partition sizes as a function of an increasing number of genomes in random subsets of genomes. Plain lines connect the medians while colored areas represent the interquartile ranges. A regression curve (bold dashed line) is drawn fitting all the points of each partition by the Heaps' law ($F = \kappa N^{\gamma}$). The total area of the interquartile ranges (IQR) is indicated for each partition.
(TIF)

S3 Fig. Density distributions of the Heaps' law γ-tendencies. These γ-tendencies were obtained by fitting a Heaps' law on rarefaction curves between subset sizes of 15 to 100 genomes in the 88 most abundant species in GenBank. The exact core median and exact accessory are not shown.
(TIF)

S4 Fig. Shell γ-tendency compared to the total number of shell families normalized by the median number of gene families per genome in each species. Results for the 88 most abundant species in GenBank are represented. The points are colored by phylum and their size corresponds to the number of partitions (K) used.
(TIF)

S5 Fig. Fraction of shell families per genome compared to the number of gene families. Results for the 88 most abundant species in GenBank are represented. The points are colored by phylum and their size corresponds to the number of partitions (*K*) used.
(TIF)

S6 Fig. Fraction of cloud families per genome compared to the number of gene families. Results for the 88 most abundant species in GenBank are represented. The points are colored by phylum and their size corresponds to the number of partitions (*K*) used.
(TIF)

S7 Fig. Presence/Absence matrix of the shell genome of *L. reuteri* ordered by a Neighbor Joining tree based on the MASH distances. The leaves of the tree are colored by host or origin. This information was obtained from the metadata in GenBank files (host and isolation source qualifiers).
(TIF)

S8 Fig. Presence/Absence matrix of the shell genome of *B. longum* ordered by a Neighbor Joining tree based on the MASH distances. The leaves of the tree are colored by species clusters defined by the GTDB database (release R04-RS89), namely (*B. infantis* or *B. longum*). "NA" values correspond to genomes not available in GTDB.
(TIF)

S9 Fig. Illustration of the shell genome overlaps between MAGs or GenBank of 55 species. The x-axis represents the percentage of common shell of the GenBank shell while the y-axis corresponds to the percentage of common shell of the MAGs shell. Diamonds and squares represent MAGs and GenBank genomes, respectively. They are colored by phylum and their size indicates the number of genomes.
(TIF)

S1 File. Table compiling all the metrics obtained from the pangenomes of the 439 GenBank species. This is a CSV file.
(CSV)

S2 File. Table compiling all the metrics obtained from the comparison of PPanGGOLiN results between MAGs and GenBank genomes in 78 species. This is a CSV file.
(CSV)

S3 File. List of GenBank assembly accessions for the 439 studied species. This is a TSV file where each line corresponds to all the GenBank assembly accession used in this study for each 'species id' in the NCBI taxonomy.
(TSV)

Acknowledgments

We acknowledge Alexandre Renaux and Jonathan Mercier for their preliminary insights on pangenome graphs. We thank Mélanie Buy for drawing the PPanGGOLiN logo. Finally, we thank Guilhem Royer, Valentin Sabatet, Johan Rollin, Mohammed-Amin Madoui, Tom Delmont, Nicolas Pons and Pierre Peterlongo for all their advice along this work.

Author Contributions

Conceptualization: David Vallenet.

Data curation: Guillaume Gautreau, Adelme Bazin, Mathieu Gachet, Rémi Planel, Laura Burlot, Mathieu Dubois, Amandine Perrin.

Formal analysis: Guillaume Gautreau, Adelme Bazin.

Investigation: Guillaume Gautreau, Adelme Bazin.

Methodology: Guillaume Gautreau, Catherine Matias, Christophe Ambroise.

Software: Guillaume Gautreau, Adelme Bazin.

Supervision: David Vallenet.

Visualization: Guillaume Gautreau.

Writing – original draft: Guillaume Gautreau, Adelme Bazin, Eduardo P. C. Rocha, David Vallenet.

Writing – review & editing: Guillaume Gautreau, Adelme Bazin, Mathieu Dubois, Claudine Médigue, Alexandra Calteau, Stéphane Cruveiller, Catherine Matias, Christophe Ambroise, Eduardo P. C. Rocha, David Vallenet.

References

1. Tettelin H, Masignani V, Cieslewicz MJ, Donati C, Medini D, Ward NL, et al. Genome analysis of multiple pathogenic isolates of Streptococcus agalactiae: implications for the microbial "pan-genome". Proc Natl Acad Sci USA. 2005; 102(39):13950–13955. https://doi.org/10.1073/pnas.0506758102 PMID: 16172379
2. Medini D, Donati C, Tettelin H, Masignani V, Rappuoli R. The microbial pan-genome. Curr Opin Genet Dev. 2005; 15(6):589–594. https://doi.org/10.1016/j.gde.2005.09.006 PMID: 16185861
3. Treangen TJ, Rocha EPC. Horizontal Transfer, Not Duplication, Drives the Expansion of Protein Families in Prokaryotes. PLOS Genetics. 2011; 7(1):1–12. https://doi.org/10.1371/journal.pgen.1001284
4. Lukjancenko O, Wassenaar TM, Ussery DW. Comparison of 61 sequenced Escherichia coli genomes. Microb Ecol. 2010; 60(4):708–720. https://doi.org/10.1007/s00248-010-9717-3 PMID: 20623278
5. Acevedo-Rocha CG, Fang G, Schmidt M, Ussery DW, Danchin A. From essential to persistent genes: a functional approach to constructing synthetic life. Trends Genet. 2013; 29(5):273–279. https://doi.org/10.1016/j.tig.2012.11.001 PMID: 23219343
6. Contreras-Moreira B, Vinuesa P. GET_HOMOLOGUES, a versatile software package for scalable and robust microbial pangenome analysis. Appl Environ Microbiol. 2013; 79(24):7696–7701. https://doi.org/10.1128/AEM.02411-13 PMID: 24096415
7. Lapierre P, Gogarten JP. Estimating the size of the bacterial pan-genome. Trends Genet. 2009; 25(3):107–110. https://doi.org/10.1016/j.tig.2008.12.004 PMID: 19168257
8. Bolotin E, Hershberg R. Horizontally Acquired Genes Are Often Shared between Closely Related Bacterial Species. Front Microbiol. 2017; 8:1536. https://doi.org/10.3389/fmicb.2017.01536 PMID: 28890711
9. Vesth T, Wassenaar TM, Hallin PF, Snipen L, Lagesen K, Ussery DW. On the Origins of a Vibrio Species. Microbial Ecology. 2010; 59(1):1–13. https://doi.org/10.1007/s00248-009-9596-7 PMID: 19830476
10. Periwal V, Patowary A, Vellarikkal SK, Gupta A, Singh M, Mittal A, et al. Comparative whole-genome analysis of clinical isolates reveals characteristic architecture of Mycobacterium tuberculosis pangenome. PLoS ONE. 2015; 10(4):e0122979. https://doi.org/10.1371/journal.pone.0122979 PMID: 25853708
11. Livingstone PG, Morphew RM, Whitworth DE. Genome Sequencing and Pan-Genome Analysis of 23 Corallococcus spp. Strains Reveal Unexpected Diversity, With Particular Plasticity of Predatory Gene Sets. Front Microbiol. 2018; 9:3187. https://doi.org/10.3389/fmicb.2018.03187 PMID: 30619233
12. Koonin EV, Wolf YI. Genomics of bacteria and archaea: the emerging dynamic view of the prokaryotic world. Nucleic Acids Res. 2008; 36(21):6688–6719. https://doi.org/10.1093/nar/gkn668 PMID: 18948295
13. Baumdicker F, Hess WR, Pfaffelhuber P. The infinitely many genes model for the distributed genome of bacteria. Genome Biol Evol. 2012; 4(4):443–456. https://doi.org/10.1093/gbe/evs016 PMID: 22357598

14. Collins RE, Higgs PG. Testing the infinitely many genes model for the evolution of the bacterial core genome and pangenome. Mol Biol Evol. 2012; 29(11):3413–3425. https://doi.org/10.1093/molbev/mss163 PMID: 22752048

15. Lobkovsky AE, Wolf YI, Koonin EV. Gene frequency distributions reject a neutral model of genome evolution. Genome Biology and Evolution. 2013;. https://doi.org/10.1093/gbe/evt002 PMID: 23315380

16. Bolotin E, Hershberg R. Gene Loss Dominates As a Source of Genetic Variation within Clonal Pathogenic Bacterial Species. Genome Biol Evol. 2015; 7(8):2173–2187. https://doi.org/10.1093/gbe/evv135 PMID: 26163675

17. Moldovan MA, Gelfand MS. Pangenomic Definition of Prokaryotic Species and the Phylogenetic Structure of Prochlorococcus spp. Frontiers in Microbiology. 2018; 9:428. https://doi.org/10.3389/fmicb.2018.00428 PMID: 29593678

18. Chan AP, Sutton G, DePew J, Krishnakumar R, Choi Y, Huang XZ, et al. A novel method of consensus pan- chromosome assembly and large-scale comparative analysis reveal the highly flexible pan-genome of Acinetobacter baumannii. Genome Biol. 2015; 16:143. https://doi.org/10.1186/s13059-015-0701-6 PMID: 26195261

19. Garrison E, Siren J, Novak AM, Hickey G, Eizenga JM, Dawson ET, et al. Variation graph toolkit improves read mapping by representing genetic variation in the reference. Nat Biotechnol. 2018; 36(9):875–879. https://doi.org/10.1038/nbt.4227 PMID: 30125266

20. Kim D, Paggi JM, Park C, Bennett C, Salzberg SL. Graph-based genome alignment and genotyping with HISAT2 and HISAT-genotype. Nat Biotechnol. 2019; 37(8):907–915. https://doi.org/10.1038/s41587-019-0201-4 PMID: 31375807

21. Rakocevic G, Semenyuk V, Lee WP, Spencer J, Browning J, Johnson IJ, et al. Fast and accurate genomic analyses using genome graphs. Nat Genet. 2019; 51(2):354–362. https://doi.org/10.1038/s41588-018-0316-4 PMID: 30643257

22. Consortium TCPG. Computational pan-genomics: status, promises and challenges. Brief Bioinformatics. 2016.

23. Zekic T, Holley G, Stoye J. Pan-Genome Storage and Analysis Techniques. Methods Mol Biol. 2018; 1704:29–53. https://doi.org/10.1007/978-1-4939-7463-4_2 PMID: 29277862

24. van Tonder AJ, Mistry S, Bray JE, Hill DM, Cody AJ, Farmer CL, et al. Defining the estimated core genome of bacterial populations using a Bayesian decision model. PLoS Comput Biol. 2014; 10(8): e1003788. https://doi.org/10.1371/journal.pcbi.1003788 PMID: 25144616

25. Gumiere T, Meyer K, Burns AR, Gumiere SJ, Bohannan BJM, Andreote FD. A probabilistic model to identify the core microbial community. bioRxiv. 2018.

26. Snipen L, Almøy T, Ussery DW. Microbial comparative pan-genomics using binomial mixture models. BMC Genomics. 2009; 10:385. https://doi.org/10.1186/1471-2164-10-385 PMID: 19691844

27. Snipen L, Liland KH micropan: an R-package for microbial pan-genomics BMC Bioinformatics. 2015; 16:79. https://doi.org/10.1186/s12859-015-0517-0 PMID: 25888166

28. Fang G, Rocha EP, Danchin A. Persistence drives gene clustering in bacterial genomes. BMC Genomics. 2008; 9(1):4. https://doi.org/10.1186/1471-2164-9-4 PMID: 18179692

29. Oliveira PH, Touchon M, Cury J, Rocha EPC. The chromosomal organization of horizontal gene transfer in bacteria. Nat Commun. 2017; 8(1):841. https://doi.org/10.1038/s41467-017-00808-w PMID: 29018197

30. Singh JK, Adams FG, Brown MH. Diversity and Function of Capsular Polysaccharide in Acinetobacter baumannii. Front Microbiol. 2018; 9:3301. https://doi.org/10.3389/fmicb.2018.03301 PMID: 30687280

31. Hu D, Liu B, Dijkshoorn L, Wang L, Reeves PR. Diversity in the major polysaccharide antigen of Acinetobacter baumannii assessed by DNA sequencing, and development of a molecular serotyping scheme. PLoS ONE. 2013; 8(7):e70329. https://doi.org/10.1371/journal.pone.0070329 PMID: 23922982

32. Bastian M, Heymann S, Jacomy M. Gephi: An Open Source Software for Exploring and Manipulating Networks; 2009. Available from: http://www.aaai.org/ocs/index.php/ICWSM/09/paper/view/154.

33. Jacomy M, Venturini T, Heymann S, Bastian M. ForceAtlas2, a Continuous Graph Layout Algorithm for Handy Network Visualization Designed for the Gephi Software. PLOS ONE. 2014; 9(6):1–12. https://doi.org/10.1371/journal.pone.0098679

34. Page AJ, Cummins CA, Hunt M, Wong VK, Reuter S, Holden MT, et al. Roary: Rapid large-scale prokaryote pan genome analysis. Bioinformatics. 2015; 31(22):3691–3693. https://doi.org/10.1093/bioinformatics/btv421 PMID: 26198102

35. Tettelin H, Riley D, Cattuto C, Medini D. Comparative genomics: the bacterial pan-genome. Curr Opin Microbiol. 2008; 11(5):472–477. https://doi.org/10.1016/j.mib.2008.09.006 PMID: 19086349

36. Parks DH, Chuvochina M, Waite DW, Rinke C, Skarshewski A, Chaumeil PA, et al. A standardized bacterial taxonomy based on genome phylogeny substantially revises the tree of life. Nat Biotechnol. 2018; 36(10):996–1004. https://doi.org/10.1038/nbt.4229 PMID: 30148503

37. Ondov BD, Treangen TJ, Melsted P, Mallonee AB, Bergman NH, Koren S, et al. Mash: fast genome and metagenome distance estimation using MinHash. Genome Biology. 2016; 17(1):132. https://doi.org/10.1186/s13059-016-0997-x PMID: 27323842

38. Criscuolo A. A fast alignment-free bioinformatics procedure to infer accurate distance-based phylogenetic trees from genome assemblies. Research Ideas and Outcomes. 2019; 5:e36178. https://doi.org/10.3897/rio.5.e36178

39. Hawkey J, Monk JM, Billman-Jacobe H, Palsson B, Holt KE. Impact of insertion sequences on convergent evolution of Shigella species. bioRxiv. 2019.

40. Oh PL, Benson AK, Peterson DA, Patil PB, Moriyama EN, Roos S, et al. Diversification of the gut symbiont Lactobacillus reuteri as a result of host-driven evolution. ISME J. 2010; 4(3):377–387. https://doi.org/10.1038/ismej.2009.123 PMID: 19924154

41. Pasolli E, Asnicar F, Manara S, Zolfo M, Karcher N, Armanini F, et al. Extensive Unexplored Human Microbiome Diversity Revealed by Over 150,000 Genomes from Metagenomes Spanning Age, Geography, and Lifestyle. Cell. 2019; 176(3):649–662. https://doi.org/10.1016/j.cell.2019.01.001 PMID: 30661755

42. Brockhurst MA, Harrison E, James PJ, Richards T, McNally A, MacLean C The Ecology and Evolution of Pangenomes Current Biology. 2019; 29(20):1094–1103.

43. Vallenet D, Calteau A, Dubois M, Amours P, Bazin A, Beuvin M, et al. MicroScope: an integrated platform for the annotation and exploration of microbial gene functions through genomic, pangenomic and metabolic comparative analysis. Nucleic Acids Research. 2019. https://doi.org/10.1093/nar/gkz926

44. Dempster AP, Laird NM, Rubin DB. Maximum likelihood from incomplete data via the EM algorithm. JOURNAL OF THE ROYAL STATISTICAL SOCIETY, SERIES B. 1977; 39(1):1–38.

45. Biernacki C, Celeux G, Govaert G. Assessing a mixture model for clustering with the integrated completed likelihood. IEEE Transactions on Pattern Analysis and Machine Intelligence. 2000; 22(7):719–725. https://doi.org/10.1109/34.865189

46. Schwarz G. Estimating the Dimension of a Model. Ann Statist. 1978; 6(2):461–464. https://doi.org/10.1214/aos/1176344136

47. Ambroise C, Dang M, Govaert G. Clustering of Spatial Data by the EM Algorithm. In: Soares A, Gómez-Hernandez J, Froidevaux R, editors. geoENV I—Geostatistics for Environmental Applications. Dordrecht: Springer Netherlands; 1997. p. 493–504.

48. Ambroise C, Govaert G. Convergence of an EM-type algorithm for spatial clustering. Pattern Recognition Letters. 1998; 19(10):919–927. https://doi.org/10.1016/S0167-8655(98)00076-2

49. Dang M, Govaert G. Spatial Fuzzy Clustering using EM and Markov Random Fields. In: International-Journal of System Research and Information Science; 1998. p. 183–202.

50. Bouguila N. On multivariate binary data clustering and feature weighting. Computational Statistics and Data Analysis. 2010; 54(1):120–134. https://doi.org/10.1016/j.csda.2009.07.013

51. Yamamoto M, Hayashi K. Clustering of multivariate binary data with dimension reduction via L1-regularized likelihood maximization. Pattern Recognition. 2015; 48(12):3959–3968. https://doi.org/10.1016/j.patcog.2015.05.026

52. Śmieja M, Hajto K, Tabor J. Efficient mixture model for clustering of sparse high dimensional binary data. Data Mining and Knowledge Discovery. 2019.

53. Miele V, Penel S, Duret L. Ultra-fast sequence clustering from similarity networks with SiLiX. BMC Bioinformatics. 2011; 12:116. https://doi.org/10.1186/1471-2105-12-116 PMID: 21513511

54. Konstantinidis KT, Tiedje JM. Genomic insights that advance the species definition for prokaryotes. Proc Natl Acad Sci USA. 2005; 102(7):2567–2572. https://doi.org/10.1073/pnas.0409727102 PMID: 15701695

55. Hyatt D, Chen GL, Locascio PF, Land ML, Larimer FW, Hauser LJ. Prodigal: prokaryotic gene recognition and translation initiation site identification. BMC Bioinformatics. 2010; 11:119. https://doi.org/10.1186/1471-2105-11-119 PMID: 20211023

56. Laslett D, Canback B. ARAGORN, a program to detect tRNA genes and tmRNA genes in nucleotide sequences. Nucleic Acids Res. 2004; 32(1):11–16. https://doi.org/10.1093/nar/gkh152 PMID: 14704338

57. Nawrocki EP, Eddy SR. Infernal 1.1: 100-fold faster RNA homology searches. Bioinformatics. 2013; 29(22):2933–2935. https://doi.org/10.1093/bioinformatics/btt509 PMID: 24008419

58. Kalvari I, Argasinska J, Quinones-Olvera N, Nawrocki EP, Rivas E, Eddy SR, et al. Rfam 13.0: shifting to a genome-centric resource for non-coding RNA families. Nucleic Acids Res. 2018; 46(D1):D335–D342. https://doi.org/10.1093/nar/gkx1038 PMID: 29112718

59. Steinegger M, Soeding J. Sensitive protein sequence searching for the analysis of massive data sets. Nat Biotech. 2017. https://doi.org/10.1038/nbt.3988

60. Grüning B, Dale R, Sjödin A, Chapman BA, Rowe J, Tomkins-Tinch CH, et al. Bioconda: sustainable and comprehensive software distribution for the life sciences. Nature methods. 2018; 15(7):475–476. https://doi.org/10.1038/s41592-018-0046-7 PMID: 29967506

6.3 Évolution de PPanGGOLiN

PPanGGOLiN était initialement une méthode de partitionnement statistique de pangénomes. Pour faciliter et surtout encourager son usage, il est devenu petit à petit une suite logicielle qui permet de réaliser de multiples analyses autour des pangénomes avec différents types de formats d'entrée.

Aujourd'hui, PPanGGOLiN permet d'avoir des entrées simples à fournir pour les utilisateurs, c'est-à-dire des fichiers fasta qui représentent les génomes, ou des fichiers gff ou gbff qui contiennent les génomes et leurs annotations, ou éventuellement un fichier qui représente l'association des gènes en familles de gènes. Les entrées permettent de définir toutes les opérations que PPanGGOLiN va réaliser, c'est-à-dire éventuellement l'annotation syntaxique des génomes si elle est nécessaire, le clustering des gènes en familles de gènes si c'est nécessaire, la construction du graphe de pangénome, le partitionnement et l'écriture de plusieurs fichiers de sorties, notamment des figures ou des fichiers qui permettent d'utiliser d'autres outils pour compléter des analyses.

Plusieurs nouveautés non publiées de PPanGGOLiN, et plusieurs articles se comparant à PPanGGOLiN par différents aspects ont été publiés par d'autres groupes, et certains méritent mentions.

6.3.1 Construction des familles de gènes

Lors de la publication de l'article, PPanGGOLiN utilisait un workflow du logiciel MMseqs2 (Steinegger et al., 2017) pour réaliser le calcul des familles de gènes. Par la suite, nous avons remarqué qu'une large proportion des gènes du *cloud* génomes était, non pas des gènes nouvellement acquis, mais des fragments de gènes existants du *shell* ou du *persistent*. J'ai donc développé une approche qui se place après l'étape originale de MMseqs2, qui a pour but d'associer les familles qui correspondraient à des fragments de gènes aux familles dont ils sont potentiellement issus.

Une fois les familles calculées avec MMseqs2, une séquence représentante est extraite de chaque famille. Toutes les séquences représentantes sont ensuite alignées avec MMseqs2 avec les mêmes paramètres d'identité et de couverture que la première étape (quels que soient ces paramètres), mais en utilisant un paramètre de couverture qui ne considère que la couverture de la plus petite séquence comparée. Ensuite, un graphe de similarité est construit, où chaque nœud est une famille de gènes résultant du premier clustering, et les arêtes sont les alignements obtenus avec MMseqs2 entre les représentantes de ces familles. Le nombre de gènes inclus dans ces familles ainsi que la longueur de la protéine représentante est étiqueté dans chaque nœud.

Ensuite, chaque nœud v est considéré séparément ainsi :

— Pour tous les voisins u de v, si u a une séquence représentante plus longue et inclut plus de gènes, u est indiqué comme potentielle famille "parente" de v, et v est annoté comme fragment.

— Tous les gènes associés à v sont annotés comme étant des fragments et réassociés à leur famille parente u.

— Si à son tour v est identifiée comme fragment d'une autre famille v', alors toutes les familles formellement identifiées comme fragments de v sont également associées à v'.

Finalement, on obtient un sous-ensemble des familles originales, dont certains gènes membres sont annotés comme fragments.

Cet algorithme de défragmentation des familles n'a jamais été publié, mais il fût testé par les auteurs de Panaroo (Tonkin-Hill et al., 2020), qui ont comparé leur méthode dont l'un des objectifs est relativement similaire (éliminer les fragments) à notre approche, parmi d'autres. PPanGGOLiN apparaît alors dans leur article comme la deuxième meilleure approche (parmi l'ensemble des logiciels qu'ils ont testés) en terme d'annotations conflictuelles (la meilleure étant leur propre approche, bien évidemment). Ils ont par la même occasion mesuré les ressources informatiques consommées par les différents logiciels, et PPanGGOLiN est le plus rapide d'entre eux d'un facteur 10, sur le plus gros pangénome de leur article, constitué de 1000 génomes et pour un usage en RAM équivalent (mais légèrement supérieur) aux autres outils. PPanGGOLiN fut aussi testé dans une évaluation récente des méthodes de pangénomique (Urhan et al., 2021), en comparaison à Panaroo (Tonkin-Hill et al., 2020), Ptolemy qui a été fait par le même groupe qui a fait cette évaluation (Salazar et al., 2018), Roary (Page et al., 2015) et PIRATE (Bayliss et al., 2019). PPanGGOLiN était alors le 2nd à avoir le plus de gènes du *core* et le 2nd à avoir le moins de gènes unique, ce qui fait de PPanGGOLiN un excellent compromis. La conclusion des auteurs ce benchmark est que la meilleure approche, sur le jeu de données qu'ils ont utilisés, est potentiellement d'utiliser une combinaison de Ptolemy et panaroo.

6.3.2 Identification du *persistent* dans les MAGs

PPanGGOLiN était le premier outil de pangénomique capable d'identifier convenablement le génome *persistent* dans des MAGs, malgré leur possible incomplétion et fragmentation en contigs. Depuis sa publication, un second outil est sorti avec le même objectif, mOTUpan (Buck et al., 2021). L'outil utilise un partitionnement statistique, mais n'utilise pas la colocalisation et ne travaille que sur la fréquence des gènes, et semble être particulièrement intéressant dans un contexte où on doit travailler avec des MAGs très incomplets. Il semble réussir le partitionnement, là

où PPanGGOLiN semble échouer à partir du moment où la complétion descend sous 70 %, dans le sens ou la taille du *persistent* devient beaucoup plus faible (et instable) par rapport à ce qui est attendu. Une hypothèse émise est que dans le cas de génomes très fragmentés, utiliser uniquement 2 partitions plutôt que 3 peut donner de meilleurs résultats. Pour éventuellement supporter cette hypothèse, PPanGGOLiN autorise désormais à un utilisateur de fournir un nombre choisi de partitions allant de 2 à l'infini.

6.4 Conclusion

PPanGGOLiN a permis d'introduire une approche statistique de partitionnement d'un graphe de pangénome à la communauté scientifique, qui jusqu'alors ne pouvait utiliser que des approches utilisant des seuils arbitraires, qui ne prenaient pas en compte le contexte génomique des familles de gènes qui constituent le pangénome. Il permet d'obtenir un partitionnement *a priori* stable quel que soit le nombre de génomes, tant qu'ils sont suffisamment complets et peut manipuler plusieurs dizaines de milliers de génomes d'une même espèce, ce qui devrait être suffisant pour la plupart des analyses de pangénomique pour les années à venir.

La quantité d'information accessible au travers de ce graphe est colossale à l'échelle de la génomique et la suite du travail de ma thèse a été de développer des fonctionnalités permettant de rendre accessible la bioanalyse de ces données. Ces bioanalyses sont réalisables notamment grâce à l'information apportée simultanément par les partitions, et par le graphe de pangénome lui-même.

Une part de ces fonctionnalités est l'identification de Régions de Plasticité Génomiques (Regions of Genomic Plasticity, ou RGP, en angais) qui contiennent majoritairement des îlots génomiques. Les îlots génomiques sont des parties clés de l'évolution des génomes procaryotes et contiennent la majorité des variations du génome en termes de contenu en gènes. Identifier ainsi des portions variables d'un graphe de pangénome est ce qui a été développé au travers du chapitre suivant et des méthodes incluses dans panRGP.

Chapitre 7

panRGP

7.1 Détection de régions de plasticité génomique dans un pangénome

Fondamentalement, un graphe de pangénome est un objet de génomique comparative. La génomique comparative, comme on a pu le voir dans la section 4.2.2, a déjà été utilisée pour identifier des îlots génomiques, mais aucune méthode *a priori* ne peut utiliser toute la diversité génomique à laquelle on a accès aujourd'hui. Une des raisons pour cela est que, en théorie, réaliser toutes ces comparaisons entre tous les génomes d'une espèce est une combinatoire de ce nombre de génomes. Néanmoins, un graphe de pangénome pour une espèce est une alternative à toutes les comparaisons que l'on pourrait réaliser entre tous les génomes d'une espèce.

Une fois celui-ci construit, l'étape coûteuse étant déjà réalisée, il ne reste en théorie plus qu'à extraire les portions variables. Ces portions variables ne correspondent pas uniquement à des îlots génomiques : on y retrouve aussi des plasmides, et éventuellement des gènes perdus dans certaines lignées. panRGP vient répondre à cette problématique, en identifiant des régions de plasticité génomique, ou RGP (Regions of Genomic Plasticity) dans le pangénome.

Dans un second temps, on peut vouloir analyser les régions de plasticité au regard du contexte dans lequel elles évoluent. Ainsi, être capable d'identifier que certaines RGP sont dans le même contexte *persistent* est particulièrement intéressant, car cela permet de suivre l'évolution de ces régions dans l'espèce. Cela correspond typiquement à l'étude de spots, ou de hotspots (points chauds), dans les génomes. Peu de méthodes de génomique comparée dédiées à cette problématique sont capables d'aller au-delà de quelques dizaines de génomes. panRGP, publié le 29 décembre 2020, introduit donc une méthode capable de comparer les bordures de gènes *persistent* de centaines de milliers de RGP. L'approche est analogue à celle présentée dans Oliveira et al., 2017 mais autorise une petite

variabilité dans la bordure, et ne repose pas sur l'usage d'un génome pivot.

7.2 Article 2 : panRGP : a pangenome-based method to predict genomic islands and explore their diversity

Bioinformatics, 36(26), 2020, i651–i658
doi: 10.1093/bioinformatics/btaa792
ECCB2020

Genomes

panRGP: a pangenome-based method to predict genomic islands and explore their diversity

Adelme Bazin, Guillaume Gautreau, Claudine Médigue, David Vallenet*,† and Alexandra Calteau*,†

LABGeM, Génomique Métabolique, CEA, Genoscope, Institut François Jacob, Université d'Évry, Université Paris-Saclay, CNRS, Evry, France

*To whom correspondence should be addressed.

†The authors wish it to be known that, in their opinion, the last two authors should be regarded as Joint Last Authors.

Abstract

Motivation: Horizontal gene transfer (HGT) is a major source of variability in prokaryotic genomes. Regions of genome plasticity (RGPs) are clusters of genes located in highly variable genomic regions. Most of them arise from HGT and correspond to genomic islands (GIs). The study of those regions at the species level has become increasingly difficult with the data deluge of genomes. To date, no methods are available to identify GIs using hundreds of genomes to explore their diversity.

Results: We present here the panRGP method that predicts RGPs using pangenome graphs made of all available genomes for a given species. It allows the study of thousands of genomes in order to access the diversity of RGPs and to predict spots of insertions. It gave the best predictions when benchmarked along other GI detection tools against a reference dataset. In addition, we illustrated its use on metagenome assembled genomes by redefining the borders of the *leuX* tRNA hotspot, a well-studied spot of insertion in *Escherichia coli*. panRPG is a scalable and reliable tool to predict GIs and spots making it an ideal approach for large comparative studies.

Availability and implementation: The methods presented in the current work are available through the following software: https://github.com/labgem/PPanGGOLiN. Detailed results and scripts to compute the benchmark metrics are available at https://github.com/axbazin/panrgp_supdata.

Contact: vallenet@genoscope.cns.fr or acalteau@genoscope.cns.fr

1 Introduction

Horizontal gene transfer (HGT) is a major mechanism that shapes gene repertoires of bacterial species providing and maintaining diversity at the population level (Niehus *et al.*, 2015; Ochman *et al.*, 2000). This pervasive evolutionary process spreads genes between, potentially very distant, bacterial lineages (Thomas and Nielsen, 2005) and is a significant source of gene novelty (Treangen and Rocha, 2011). In prokaryotes, HGT is promoted by three main mechanisms: conjugation, transformation or transduction. Clusters of consecutive genes likely acquired by HGT are commonly described as genomic islands (GIs) (Hacker and Kaper, 2000). GIs are part of the flexible gene pool of bacteria and may bring an evolutionary advantage allowing adaptations to new environments or bringing new pathogenicity capacities for instance (Hacker and Carniel, 2001). They are widely distributed in pathogenic and environmental microorganisms outlining the interest that researchers have in studying their evolution and functional impact on bacterial populations.

GIs are characterized by their large size (>10 KB), a usually different G+C content compared with the rest of the chromosome for recent acquisitions (Lawrence and Ochman, 1997), and are often associated with mobile elements, such as transposons, integrons, integrative conjugative elements and prophages. Many GIs insertion sites are associated with tRNA-encoding genes and are flanked with repeat structures (Dobrindt *et al.*, 2004). Some of those insertion sites, called hotspots, are more active than the rest of the genome in terms of acquisition rate of new elements and tend to have a much more diverse gene content even between closely related individuals (Oliveira *et al.*, 2017). Since GIs carry so many genes of interest, countless methods have been published to detect and analyze them (Bertelli *et al.*, 2019; Langille *et al.*, 2008; Lu and Leong, 2016). These methods are often grouped into two categories: composition-based methods and comparative genomic-based methods. A recent review describes many of the methods that were developed in this field (Bertelli *et al.*, 2019) and benchmarks them using a curated dataset (Langille *et al.*, 2008).

Nowadays, a deluge of microbial genomes is available in public databanks with more than half a million prokaryotic genomes in Genbank (last accessed March 2, 2020) (Sayers *et al.*, 2019). In parallel, environmental data made of metagenome assembled genomes (MAGs) or single-cell assembled genomes are increasing

© The Author(s) 2020. Published by Oxford University Press. All rights reserved. For permissions, please e-mail: journals.permissions@oup.com

Downloaded from https://academic.oup.com/bioinformatics/article/36/Supplement_2/i651/6055938 by Genoscope CNS user on 05 August 2021

dramatically. Hence, conducting comparative genomics studies on hundreds to thousands of genomes has become a challenge and can lead to millions of pairwise comparisons requiring intensive computations for analysis. Accurately identifying GIs in all of those genomes that may be incomplete and fragmented is becoming crucial to get a global overview of the diversity within species.

To tackle this challenge, methods based on pangenomes could be the answer. The concept of pangenome corresponds to the entire gene repertoire of a taxonomic group (Tettelin *et al.*, 2005). A pangenome can be described by two components: the core genome, which contains genes shared by all individuals, and the variable genome, which gathers every other genes. Lately, multiple methods have been developed to study pangenome structures and to perform comparative studies on hundreds of genomes (Fouts *et al.*, 2012; Gautreau *et al.*, 2020; Page *et al.*, 2015; Snipen and Liland, 2015). Among them, the PPanGGOLiN method proposes a representation of all genes of all genomes in a pangenome graph where nodes represent gene families and edges represent genomic neighborhood (Gautreau *et al.*, 2020). It uses a statistical model that combines the information of the presence/absence of gene families and the topology of the pangenome graph to classify gene families in three partitions as following: (i) the *persistent* genome, which corresponds to genes that are present in most individuals of the studied clade, (ii) the *shell* genome, which groups genes that are conserved between some individuals of the group but not most and (iii) the *cloud* genome, which corresponds to genes that are rare within the population and found only in one or a few individuals. The *shell* and *cloud* genomes are partitions of the variable genome. The *persistent* genome is conceptually similar to the core genome but it is more adapted to large-scale genomic comparisons as it allows for missing genes due to punctual evolutionary loss events or technical reasons, such as assembly or gene calling artifacts (Gautreau *et al.*, 2020).

As most newly acquired genes are expected to arise from HGT events (Treangen and Rocha, 2011), it is expected that most of the genes included in the *shell* and *cloud* genomes have a non-vertical origin and are either part of GIs or plasmids. Here, we use the concept of regions of genome plasticity (RGP) to refer to regions composed of *shell* and *cloud* genomes (Mathee *et al.*, 2008; Ogier *et al.*, 2010; Vallenet *et al.*, 2009). We expect RGPs to mostly consist of GIs or plasmids. In the case of significant genome reduction in the studied species, regions that have been lost in some individuals might be included in the *shell* genome and thus considered as RGPs. While GIs have previously been studied in the scope of pangenomes and were shown to include most of the variable genome in different species (Kittichotirat *et al.*, 2011; Zhu *et al.*, 2019), so far no method uses the concept of pangenome to predict them.

To study the evolution of GIs in a population, it may be of interest to look at spots of insertion within a pangenome. The name spot of insertion has been used previously to describe the variable genome of multiple individuals that was located in-between the same core genes (Lescat *et al.*, 2009; Oliveira *et al.*, 2017). A related concept can be found in the literature in the name of flexible genomic regions, which is a group of flexible GIs (Chan *et al.*, 2015), a term originally used to describe the variable genome of different individuals, which was located in-between the same core genes and involved in similar functions (Rodriguez-Valera and Ussery, 2012). We will use the term spot throughout this article and we do not assume that genes in a same spot have related functions.

In this article, we propose a new method called panRGP, which detects RGPs and gathers them into spots of insertion to study the dynamics of GIs. It is a comparative genomic-based method that uses a pangenome graph reconstructed from hundreds to thousands of genomes of the same species. We benchmarked panRGP along a selection of other tools against a previously published dataset on GIs (Langille *et al.*, 2008). Finally, we illustrated its use on incomplete and fragmented genomes, such as MAGs in the context of the analysis of an insertion spot in *Escherichia coli*.

2 Materials and methods

2.1 PanRGP method

The panRGP method predicts RGPs from a query genome that is annotated with a set of protein-coding genes. It uses as input a partitioned pangenome graph that is built from the genomes of related organisms usually from the same species. This graph relies on the PPanGGOLiN data structure (Gautreau *et al.*, 2020) where nodes are homologous gene families and edges indicate a relation of genetic contiguity. The different steps for the detection of RGPs are shown in Figure 1 and are detailed in Subsections 2.1.1 and 2.1.2. Firstly, each gene of the query genome is assigned to the pangenome partition of its gene family and thus classified as *persistent*, *shell* or *cloud*. Secondly, a score is computed for each gene sequentially along the genome. It is based on both the gene partition and the score of the previous gene. Finally, RGPs are detected using the gene scores and correspond to sequences of *variable* (*shell* or *cloud*) genes possibly interrupted by few *persistent* genes. We also consider persistent genes that belong to multigenic families as potential members of RGPs. Indeed, some of them (e.g. genes encoding transposases or integrases) are associated to mobile genetic elements and are frequently found at the extremities of RGPs. In addition, RGPs from different genomes can be grouped in spots of insertion based on their conserved flanking *persistent* genes using the pangenome graph as explained in Subsection 2.1.3.

2.1.1 Computation of initial gene scores

The initial step consists in assigning a score to each gene $g, \forall g \in C$, where C is an ordered set of genes that are present on each contig of a genome assembly. The gene scores s_g are computed sequentially along the contigs as follows:

$$s_g = (s_{g-1} + f(g))^+$$

s_{g-1} is the score of previous gene on the contig. If a gene has no previous neighbor the s_{g-1} score is 0. $f(g)$ is a function whose result depends on the gene partition and on whether or not the gene belongs to a multigenic gene family:

$$f(g) = \begin{cases} -(p)^n & \text{if the gene is } \textit{persistent} \text{ and not multigenic} \\ \nu + \epsilon & \text{if the gene is } \textit{variable}, \text{ or multigenic} \end{cases}$$

n is the number of consecutive *persistent* genes previously encountered and any *variable* gene resets its value to 0. The p constant is used to penalize the inclusion of *persistent* genes in an RGP whereas the ν constant promotes the inclusion of *variable* genes. The default value of ν was set to 1 whereas it is 3 for p. Indeed, we want to penalize the insertion of several consecutive persistent genes that is rare in GIs while allowing isolated persistent genes. This penalty value is thus exponential according to the number of consecutive persistent genes (n). To ensure that the algorithm provides identical results independently of the reading direction of the contig, the ϵ constant is used and set to $1/\infty$. A gene family is considered as multigenic if it contains duplicated genes in more than d% of the genomes in the pangenome graph. The default value of d was set to 5% to not consider rare events of duplication that could correspond to assembly artifacts. In the case of circular sequences, if at least one gene had a score of 0, the algorithm continues after the end of the sequence to reevaluate gene scores from the beginning until reaching a gene score of 0 or reaching the last gene that had a score of 0 at the first pass.

2.1.2 Detection of RGPs and score updates

After all genes have been associated to a score s_g, RGPs are detected using the following algorithm on each contig.

- Step 1: initialization of a new RGP
  - If no gene on the contig has a $s_g \geq s_{min}$, stop here.
  - Select the gene g with highest score s_g (in case of equality, the gene closest to the end of the contig is selected).

Downloaded from https://academic.oup.com/bioinformatics/article/36/Supplement_2/i651/6055938 by Genoscope CNS user on 05 August 2021

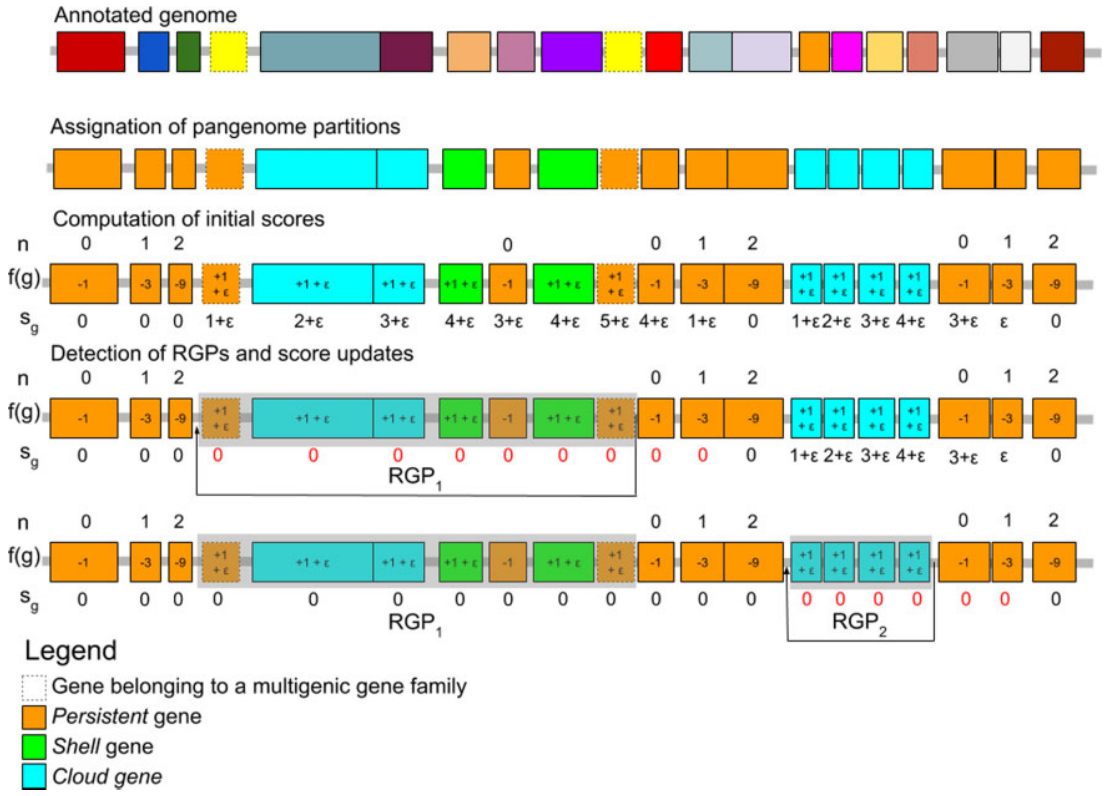


Fig. 1. Overview of the RGP detection method. The different steps of the RGP detection method are presented. Boxes represent protein-coding genes and their color is indicative of their gene families for the annotated genome and their pangenome partition for the other genome representations. Dashed boxes indicate genes belonging to multigenic gene families. In this example, two RGP are detected. RGP_1 has a score of 5, and RGP_2 a score of 4. *n* values indicate for each gene the number of downstream consecutive genes classified as persistent. *f*(*g*) values indicate the result of a function that is used to compute the gene score s_g. In this example, the default values for *p* and *v* parameters are used and are 3 and 1, respectively

  - A new RGP that ends at the selected gene *g* is created and the score s_g is assigned to the RGP.
- Step 2: extraction of the RGP
  - Add previous genes to the RGP until reaching a gene with s_g=0.
  - Set all the s_g of the RGP genes to 0.
  - Save the RGP if its length in nucleotides is $\geq l_{min}$
- Step 3: score updates
  - Recompute s_g scores from the gene selected at step 1 until reaching a gene with s_g=0.
  - Go to step 1.

This algorithm results in the prediction of RGPs that correspond to ordered sets of genes. A minimal gene score criterion s_{min} is used as a threshold to consider an RGP. Its default value is set to 4. In addition to s_{min}, a minimal length in nucleotides of an RGP l_{min} is defined and set to 3 000 by default.

2.1.3 Grouping RGPs into spots

From a set of genomes with predicted RGPs, spots are determined by comparing their *persistent* flanking genes. At both ends of each RGP, we select the *c* consecutive genes that are *persistent* and not multigenic. These genes are ordered according to their distance from the RGP and then converted into their corresponding gene family. Thus, it makes the algorithm independent of the reading direction. The borders of an RGP are thus defined as a pair of ordered sets of gene families. A graph $G(V, E)$ is built where each node v represents the borders of an RGP and each edge indicates that they share similar sets of gene families. Two borders v_i and v_j are similar if their first *e* gene families are identical or if their ordered sets overlap by at least *o* families. If all borders of two compared RGPs match, we add an edge between their corresponding nodes. Once the graph is built, all connected components are extracted and corresponds to the spots. Then, the list of associated RGPs is retrieved for each spot. The default values of *c*, *e* and *o* are 3, 1 and 2, respectively. Spots are associated to multiple metrics, such as the numbers of RGPs, gene families and different sets of gene families that compose the RGPs. RGPs that do not have *c* consecutive *persistent* genes on both ends are not considered for spot prediction. Either they are not complete as they end at contig borders or they are plasmids and thus do not have a *persistent* context.

2.2 Benchmark protocol

To assess the reliability of GI detection by panRGP, we used two previously published reference datasets (Langille *et al.*, 2008) that were recently updated (Bertelli *et al.*, 2019). The C-dataset is made of 104 genomes among 53 species from which GIs have been automatically predicted using a comparative genomic method based on IslandPick (Langille *et al.*, 2008). The L-dataset contains six genomes whose GIs have been curated. While it does not cover as much microbial diversity, it contains literature-curated GIs rather than automatically detected ones, which makes it a much more reliable source of information. Genomes of the L-dataset are also present in the C-dataset but it should be noted that the GIs of C-dataset only partly cover those of the L-dataset (Bertelli *et al.*, 2019). For each dataset and in each genome, 'positive regions' correspond to regions that are potential GIs and 'negative regions' to those that are not considered to be GIs.

A prerequisite for the execution of panRGP is the computation of pangenomes for each studied species. All available NCBI RefSeq genomes (downloaded the November 21, 2019) (Haft *et al.*, 2018) were used as PPanGGOLiN input to obtain a partionned pangenome graph. However, only species with at least 15 RefSeq genomes

Downloaded from https://academic.oup.com/bioinformatics/article/36/Supplement_2/i651/6055938 by Genoscope CNS user on 05 August 2021

could be analyzed due to the statistical constraint of the PPanGGOLiN method. Among the 54 species present in the GI datasets, 14 of them did not meet this condition. Furthermore, *Prochlorococcus marinus* was not considered in the analysis as this group of organisms does not seem to be a relevant single species at the genomic level (Parks *et al.*, 2018). An additional three had to be removed as their assemblies did not match the version, which were originally used to predict their GIs. A total of 81 genomes from 36 species from the reference datasets could be analyzed. The list of RefSeq genomes with their identifiers that were used to build pangenomes is available from https://github.com/axbazin/panrgp_supdata. The panRGP results were obtained with the PPanGGOLiN software version 1.1.72 with default parameters.

The results of panRGP were compared to other tools on the same reference datasets. We included tools that were found to correctly predict GIs in a recent study (Bertelli *et al.*, 2019): Islandpath-dimob (Bertelli and Brinkman, 2018), SigiCFR (Waack *et al.*, 2006), SigiHMM (Waack *et al.*, 2006), Alien Hunter (Vernikos and Parkhill, 2006), PredictBias (Pundhir *et al.*, 2008), ZislandExplorer (Wei *et al.*, 2017) and IslandViewer4 (Bertelli *et al.*, 2017), which combines results from Islander (Hudson *et al.*, 2015), Islandpath-dimob, SigiHMM and IslandPick (Langille *et al.*, 2008). In addition, three recent tools for GI detection were included in the benchmark: GI-cluster (Lu and Leong, 2018), XenoGI (Bush *et al.*, 2018) and IslandCafe (Jani and Azad, 2019). GI-Cluster relies on sequence composition and functional annotation to cluster regions. XenoGI uses bidirectional best hits with the information of a phylogenetic tree to identify gene families that arose from the same HGT. IslandCafe uses sequence composition and functional annotation with a custom database of hidden Markov models including genes that are often associated with HGT. All these tools were run with default parameters. XenoGI was used only with the L-dataset as it requires a manual selection of related genomes with a phylogeny. We selected four or five genomes from closely related species and compared them with mashtree (Katz *et al.*, 2019) to obtain the phylogenetic tree for each analyzed species. The selected genomes and the trees that were used for XenoGI are available from https://github.com/axbazin/panrgp_supdata. For Islandviewer and PredictBias, the predicted GIs were downloaded from their respective websites. Sofware versions, or commit numbers, and the mode of installation are provided in https://github.com/axbazin/panrgp_supdata.

To evaluate these tools, we compared their predictions with the positive and negative regions of the two datasets using the protocol described in Bertelli *et al.* (2019). The predicted regions that correspond to positive regions are considered as true positives (TP) and those that correspond to negative regions are considered as false positives (FP). The negative regions that were not predicted are true negatives (TN) and the positive regions that were not predicted are false negatives (FN). We computed Matthew's correlation coefficient (MCC), *F1score*, accuracy, precision and recall as in Bertelli *et al.* (2019) to compare the prediction performance of the different tools.

$$
\begin{aligned}
recall &= \frac{TP}{TP+FN} \\
precision &= \frac{TP}{TP+FP} \\
accuracy &= \frac{TP+TN}{TP+FP+FN+TN} \\
F1score &= \frac{2TP}{2TP+FP+FN} \\
MCC &= \frac{TP \times TN - FP \times FN}{\sqrt{(TP+FP)(TP+FN)(TN+FP)(TN+FN)}}
\end{aligned}
$$

2.3 Preparation of the MAG dataset

In addition to the benchmark, 1416 *E.coli* MAGs that have been published in a recent metagenomic study (Pasolli *et al.*, 2019) were downloaded from https://opendata.lifebit.ai/table/SGB. These MAGs were annotated using Prokka 1.13 (Seemann, 2014) before being analyzed by panRGP to predict RGPs and spots. The analysis was made with version 1.1.72 of PPanGGOLiN using the –defrag parameter to link potential gene fragments to their native gene family.

3 Results and discussion

3.1 Software overview

The methods of panRGP for the detection of RGPs and spots have been implemented in the PPanGGOLiN pangenomic software suite (version $\geq$ 1.1.72) available through Github (https://github.com/labgem/PPanGGOLiN) under the CeCiLL 2.1 open-source license. It is coded in Python3 with embedded code in C and can be easily installed using bioconda (Grüning *et al.*, 2018). We have also written an extensive documentation of every possible output files and the different ways of using the software in the GitHub wiki (https://github.com/labgem/PPanGGOLiN/wiki).

An overview of the panRGP workflow is given in Figure 2. The whole workflow can be run through the command 'ppanggolin panrgp'. First, the PPanGGOLiN partitioned pangenome graph is built from a set of input genomes chosen by the user. The genomes are expected to be from the same species and can be provided as gff3/gbff annotation files or as fasta sequences. In the latter case, genomes are annotated using the procedure described in Gautreau *et al.* (2020). A clustering step using MMseqs2 (Steinegger and Söding, 2017) is then executed if gene families are not provided as input by the user. In that case, families are built with 80% amino acid identity and 80% coverage on both query and target proteins for the sequence alignments and with the greedy set cover algorithm for the clustering (default PPanGGOLiN parameters). Afterwards, the graph is constructed and partitioned in *persistent*, *shell* and *cloud* families. Finally, RGPs and spots are predicted using the methods described above. The pangenome and all analysis results are stored in an HDF5 file. Each step of the workflow has a dedicated subcommand in the software suite. The user can then adapt parameters or (re-)execute only a part of the workflow. These subcommands take as input either the raw original files or an HDF5 file representing the pangenome.

When a reference pangenome is available, a new genome of the same species can be used as input to predict RGPs using the algorithm described in Section 2.1. This is done by the 'align' subcommand that maps the genes of the genomes to the gene families of the pangenome using the MMseqs2 software. The 'align' subcommand can also be used with protein sequences as input. In this case, they will be aligned to the pangenome gene families and the related RGPs and spots will be extracted, thus providing contextual information for proteins of interest.

The panRGP workflow ends by providing different output files. Tab-separated values files contain a summary of the predicted RGPs and spots. Figures can be drawn to represent all RGPs of a spot using the genoplotR library (Guy *et al.*, 2010). Furthermore, a subgraph of the pangenome graph (in a GEXF format) can be extracted to represent all the gene families found in a spot with their genomic organization (see Subsection 3.3). It can then be visualized with the Gephi software (Bastian *et al.*, 2009) by applying a layout algorithm, such as ForceAtlas2 (Jacomy *et al.*, 2014).

3.2 Benchmark results

To evaluate the panRGP method, we ran a benchmark as described in Section 2 in comparison with 10 other tools for GI prediction and on 2 different reference datasets. Those methods can be classified in two types: comparative-based and composition-based methods. IslandViewer4 is a hybrid method as it combines both approaches. It aggregates results from SigiHMM, which is a composition-based method, and IslandPick, which is a comparative-based method. Other methods like IslandCafe and GI-Cluster use additional functional information.

The C-dataset contains GIs on 81 genomes that were automatically predicted using a comparative genomic method. Results for the different methods are presented in Table 1. The panRGP

Downloaded from https://academic.oup.com/bioinformatics/article/36/Supplement_2/i651/6055938 by Genoscope CNS user on 05 August 2021

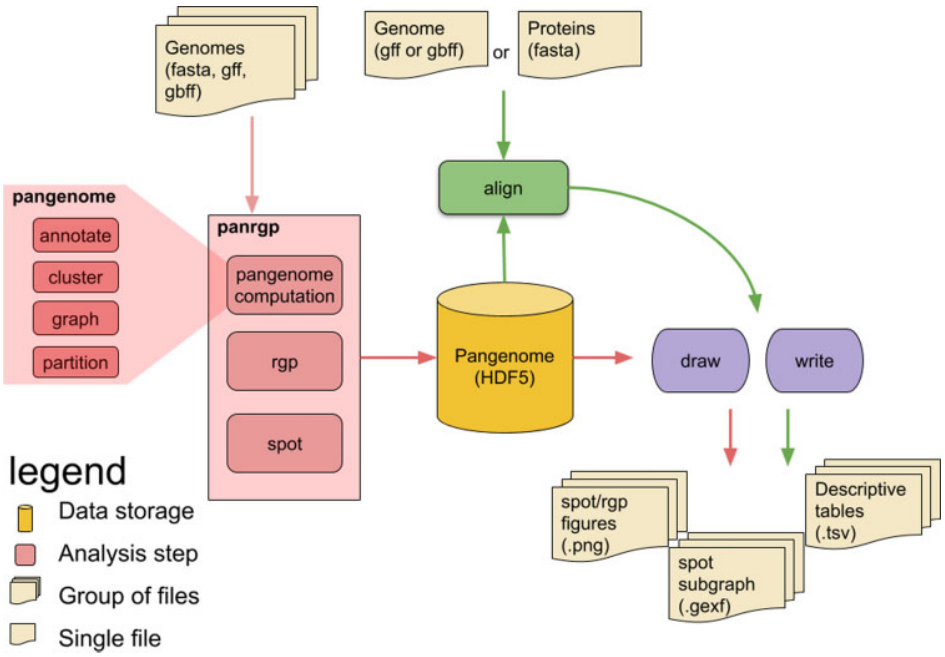


Fig. 2. Overview of the panRGP workflow. Each rounded box represents one of the possible commands of the software. The panRGP workflow computes the pangenome using the PPanGGOLiN method, predicts the RGPs and gathers them into spots. The pangenome and results are saved in an HDF5 file. This file can be used as an input with the 'align' command (i) to compare a new genome to the pangenome and predict the RGPs (ii) to align protein sequences to the pangenome families to extract related RGPs and spots. A summary of results in tsv text files and figures can be obtained using respectively 'write' and 'draw' commands, respectively

Table 1. Benchmark results on the C-dataset.

Tool	MCC	*F*1 score	Accuracy	Precision	Recall	Approach
panRGP	**0.778**	0.809	**0.924**	**0.949**	0.764	Comparative
IslandViewer4	0.762	**0.820**	0.911	0.908	**0.788**	Hybrid
IslandPath-DIMOB	0.523	0.570	0.781	0.891	0.477	Composition
GI-Cluster	0.483	0.592	0.780	0.674	0.633	Composition/Functional
SigiCRF	0.450	0.492	0.803	0.909	0.398	Composition
PredictBias	0.393	0.528	0.739	0.593	0.633	Composition
IslandCafe	0.377	0.444	0.761	0.769	0.355	Composition/Functional
AlienHunter	0.364	0.526	0.754	0.586	0.551	Composition
SigiHMM	0.338	0.455	0.756	0.655	0.376	Composition
ZislandExplorer	0.194	0.260	0.705	0.690	0.201	Composition

The best values for each metric are indicated with a bold font.

method gives the best results in terms of MCC, accuracy and precision whereas IslandViewer4 produces better results regarding *F1score* and recall. Overall, computed metrics for IslandViewer4 and panRGP are very close. Methods based on sequence composition do not perform as well. Some have a good precision (e.g. SigiCFR, IslandPath-DIMOB) but their recall and accuracy values are lower than comparative genomics based methods. It should be noted that the C-dataset was generated using comparative genomics, which may explain the good performance of IslandViewer4 and panRGP, which are the only tools using comparative genomics.

The L-dataset contains curated GIs on six genomes that were expertized. Results for the different methods are presented in Table 2. The benchmark was carried out on the same methods plus XenoGI, which is a comparative-based method that uses a phylogenetic tree to detect insertion events from HGT. As with the C-dataset, panRGP performs best in terms of MCC, accuracy and precision, as well as the *F1score*. Regarding the recall, XenoGI provides the best results. IslandViewer4 performs well but the metrics drop somewhat in comparison to the C-dataset. IslandCafe and GI-cluster which both rely on composition and functional annotation fare much better on this dataset. Overall, composition-based methods perform worse than comparative-based ones.

This benchmark shows that comparative-based methods are the most reliable to predict GIs in comparison to composition-based methods. Surprisingly, IslandViewer4 that combines both approaches does not perform better than panRGP or XenoGI, which only rely on comparative genomics. We can assume that the tools based on comparative genomics are more reliable when they use a larger number of genomes representing a greater diversity within the studied species. This may explain why panRGP comes on top in this study. Indeed, the reference pangenome graph that is used by panRGP can contain information from hundreds to thousands of genomes. On the other hand, IslandViewer4 and XenoGI are limited to few genomes. IslandViewer4 uses up to 12 genomes (6 by default). For XenoGI, authors indicate the use of 500 gigabytes (GB) of memory and a run time of 20 h on 50 threads for 40 strains, thus limiting its use on larger datasets. Conversely, panRGP can use far

Downloaded from https://academic.oup.com/bioinformatics/article/36/Supplement_2/i651/6055938 by Genoscope CNS user on 05 August 2021

Table 2. Benchmark results on the L-dataset.

Tool	MCC	*F*1 score	Accuracy	Precision	Recall	Approach
panRGP	**0.879**	**0.932**	**0.931**	**1.000**	0.884	Comparative
XenoGI	0.829	0.917	0.905	0.935	**0.924**	Comparative
IslandViewer4	0.684	0.791	0.817	0.998	0.669	Hybrid
IslandCafe	0.606	0.715	0.752	**1.000**	0.574	Composition/Functional
GI-Cluster	0.589	0.743	0.761	0.870	0.714	Composition/Functional
PredictBias	0.587	0.805	0.788	0.856	0.771	Composition
IslandPath-DIMOB	0.527	0.636	0.702	0.998	0.479	Composition
SigiCRF	0.424	0.520	0.687	0.993	0.434	Composition
AlienHunter	0.398	0.642	0.705	0.753	0.570	Composition
SigiHMM	0.268	0.444	0.591	0.817	0.325	Composition
ZislandExplorer	0.163	0.278	0.513	0.833	0.180	Composition

The best values for each metric are indicated with a bold font.

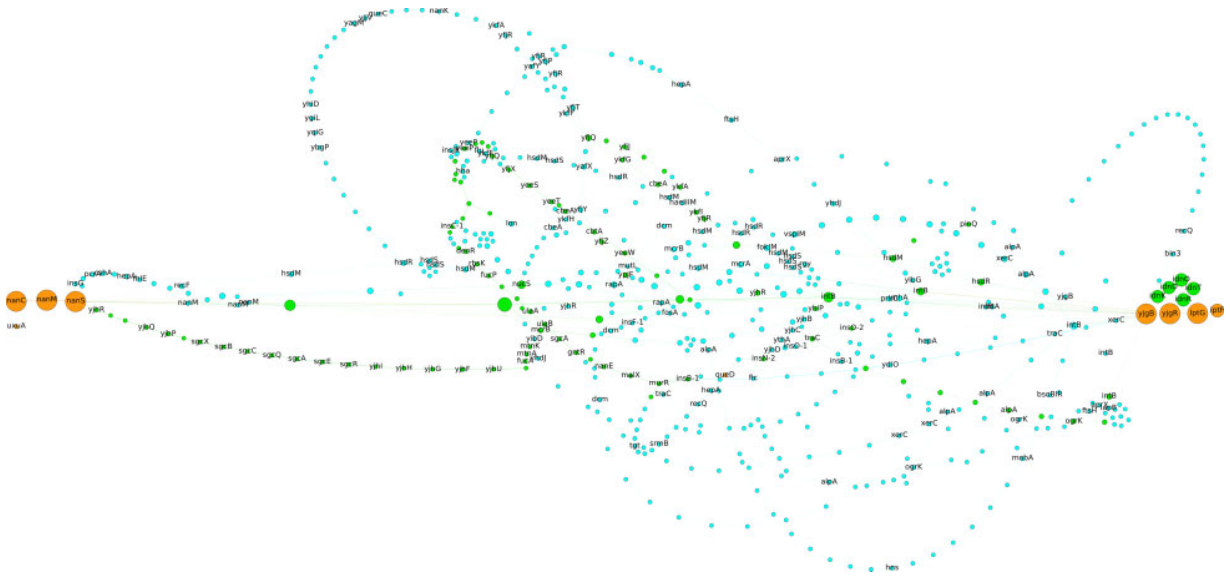


Fig. 3. Pangenome subgraph of the *leuX* hotspot in MAGs of *E.coli*. This figure illustrates the genetic diversity of the *leuX* hotspot both in terms of gene content and genome organization. Each node is a gene family and an edge is drawn between two families that share neighboring genes. Colors for *persistent*, *shell* and *cloud* pangenomes partitions are orange, green and blue, respectively. The size of the nodes is proportional to the number of occurrences in the spot. For each family, the most represented gene name is indicated. The tRNA *leuX* and the *fim* operon are not represented as they are not members of spot predicted by panRGP. *leuX* is located next to *yjgB* and the *fim* operon is just before *nanC*. This visualization was produced using Gephi (Bastian *et al.*, 2009) and the ForceAtlas2 layout algorithm (Jacomy *et al.*, 2014)

more than hundreds of genomes without requiring extensive computational resources. For the complete workflow of panRGP including the pangenome construction and RGP/spot prediction on *E.coli* genomes, it takes 3 min and 1.2 GB of memory to analyze 40 strains, 45 min and 14 GB of memory for 1000 strains and 7 h 38 min and 307 GB of memory for 10000 strains using 16 threads of an Intel Xeon CPU E5-2699v3. Most of the time is dedicated to genome annotation and pangenome partitioning.

3.3 Application of panRGP on a pangenome built from MAGs

To illustrate the potential of panRGP on MAGs, we studied the genomic context of a previously described hotspot in *E.coli* (Lescat *et al.*, 2009) using a pangenome constructed from MAG sequences from a recently published metagenome dataset (Pasolli *et al.*, 2019). The pangenome was built using 1 413 MAGs. It is made of 43 741 gene families including 5 111 342 genes. Those families are partitioned into 3 724 *persistent*, 2 490 *shell* and 37 618 *cloud* gene families. The *persistent* genome is very similar to the one that was already computed on a pangenome built from GenBank genomes (3 706 families) (Gautreau *et al.*, 2020) indicating that the *persistent* was well retrieved even though the MAGs are much more fragmented and incomplete than genomes from isolates. As a consequence, pangenome partitions obtained on these MAGs can be a reliable source of information to predict GIs with panRGP. A total of 47 692 regions were predicted by panRGP among which 18 030 are in-between *persistent* genes and could be used to predict spots of insertion with the algorithm previously described. Those 18 030 RGPs have been grouped into 294 spots of insertion. A list of all spots with descriptive metrics is available from https://github.com/axbazin/panrgp_supdata.

Among all the predicted spots, we focused our analysis on the *leuX* tRNA hotspot as it is one of the most diverse region of *E.coli* and involved in pathogenicity (Blum *et al.*, 1994; Touchon *et al.*, 2009). This spot was described as being in-between two core genes, namely *uxuA* and *abr* (previously named *yjgB*), in a comparative analysis of 14 genomes (Lescat *et al.*, 2009). To retrieve this spot from panRGP results, we used the protein sequences of both of those genes from *E.coli* K-12 MG1655 [P24215 and P27250 proteins from UniProt (UniProt Consortium, 2019)] and aligned them to the pangenome gene families (subcommand 'align' see Section 2). We find that only one panRGP spot is associated with both genes and corresponds to the spot number 10. It gathers 131 RGPs that are

Downloaded from https://academic.oup.com/bioinformatics/article/36/Supplement_2/i651/6055938 by Genoscope CNS user on 05 August 2021

represented by 79 different sets of gene families. The size of these RGPs is between 5 and 91 genes, with an average of 19 genes. There are a total of 585 different gene families. Among all predicted spots, it is the third most diverse in gene family content, confirming that it is one of the most dynamic regions of the *E.coli* genome.

Figure 3 shows the pangenome subgraph of this spot in a compact representation with the indication of gene names that are most often associated with each family. The spot borders were formerly assumed to be *ahr* and *uxuA* genes (Lescat *et al.*, 2009). While we agree that *ahr* borders the spot, the *nan* operon composed of the *nanS*, *nanM* and *nanC* genes (previously named *yjhS*, *yjhT* and *yjhA*) is the most common border predicted by panRGP instead of *uxuA*. Indeed, *nan* genes are *persistent* and thus present in most *E.coli* genomes from the gut microbiome. The punctual deletion of the fimbrial operon *fim* in few *E.coli* strains [e.g. 55989 and O42 strains, see Fig. 4 in Lescat *et al.* (2009)], which is located between the *nan* operon and *uxuA* in the other strains, misled the authors in determining the hotspot frontiers as few genomes were available when their work was published.

The spot detection method illustrated here is inspired from Oliveira *et al.* (2017) with, however, a fundamental difference: our method is not centered on a pivot genome but is applied on the whole pangenome without any reference. Furthermore, it allows for variations in terms of gene content and organization in the definition of the spot borders. The *leuX* hotspot is a great example showing that spot borders can vary throughout the evolution of a species. The panRGP method provides an exhaustive list of spot associated with several metrics (e.g. numbers of RGPs, gene families and different sets of families). Those results can be the starting point of studies on the dynamics of GIs within and between species.

4 Conclusion

We presented an original method that can identify RGPs on thousands of genomes and analyze them together to detect spots of insertion. Indeed, panRGP uses a partitioned pangenome graph of gene families that makes comparative-based approach to predict GIs more efficient. Indeed, our method is much more scalable on large datasets than already published tools, which rely on time-consuming pairwise sequence comparisons. We showed that panRGP results are highly reliable when compared to a dataset of curated GIs. We introduced a novel algorithm for the detection of spots of insertion, which was illustrated in the context of the analysis of an *E.coli* hotspot using MAGs from the human gut. Overall, we believe that panRGP provides an original approach to detect GIs to study their diversity and dynamics in a species of interest. Its ability to predict GIs and spots among thousands of genomes makes it an ideal approach for large-scale studies.

The tool is freely available and easily installable as part of the PPanGGOLiN software suite. It is also integrated in the MicroScope platform with a dedicated web page for result analysis and exploration of prokaryotic genomes (Vallenet *et al.*, 2019). An improvement of panRGP could be to analyze conserved alternative paths within RGPs using the pangenome graph structure. This could allow to automatically identify functional modules, i.e. set of genes involved in the same biological process akin to what was described in Lescat *et al.* (2009) and Touchon *et al.* (2009).

Acknowledgements

Valentin Sabatet for initial work and proof of concept on studying plastic regions using pangenome partitions. Mark Stam and Mathieu Dubois for insightful discussions. Mylène Beuvin for her artistic sense.

Funding

This research was supported in part by the Phare PhD program of the French Alternative Energies and Atomic Energy Commission (CEA) (to A.B.); and the Irtelis PhD program of the CEA (to G.G.).

Conflict of Interest: none declared.

References

Bastian,M. *et al.* (2009) Gephi: an open source software for exploring and manipulating networks. Icwsm, **8**, 361–362

Bertelli,C. and Brinkman,F.S. (2018) Improved genomic island predictions with IslandPath-DIMOB. *Bioinformatics*, **34**, 2161–2167.

Bertelli,C. *et al.*; Simon Fraser University Research Computing Group. (2017) IslandViewer 4: expanded prediction of genomic islands for larger-scale datasets. *Nucleic Acids Res.*, **45**, W30–W35.

Bertelli,C. *et al.* (2019) Microbial genomic island discovery, visualization and analysis. *Brief. Bioinformatics*, **20**, 1685–1698.

Blum,G. *et al.* (1994) Excision of large DNA regions termed pathogenicity islands from tRNA-specific loci in the chromosome of an *Escherichia coli* wild-type pathogen. *Infect. Immun.*, **62**, 606–614.

Bush,E.C. *et al.* (2018) xenoGI: reconstructing the history of genomic island insertions in clades of closely related bacteria. *BMC Bioinformatics*, **19**, 32.

Chan,A.P. *et al.* (2015) A novel method of consensus pan-chromosome assembly and large-scale comparative analysis reveal the highly flexible pan-genome of *Acinetobacter baumannii*. *Genome Biol.*, **16**, 143.

Dobrindt,U. *et al.* (2004) Genomic islands in pathogenic and environmental microorganisms. *Nat. Rev. Microbiol.*, **2**, 414–424.

Fouts,D.E. *et al.* (2012) PanOCT: automated clustering of orthologs using conserved gene neighborhood for pan-genomic analysis of bacterial strains and closely related species. *Nucleic Acids Res.*, **40**, e172.

Gautreau,G. *et al.* (2020) PPanGGOLiN: depicting microbial diversity via a partitioned pangenome graph. *PLoS Comput. Biol.*, **16**, e1007732.

Grüning,B. *et al.*; The Bioconda Team. (2018) Bioconda: sustainable and comprehensive software distribution for the life sciences. *Nat. Methods*, **15**, 475–476.

Guy,L. *et al.* (2010) genoPlotR: comparative gene and genome visualization in R. *Bioinformatics*, **26**, 2334–2335.

Hacker,J. and Carniel,E. (2001) Ecological fitness, genomic islands and bacterial pathogenicity. *EMBO Rep.*, **2**, 376–381.

Hacker,J. and Kaper,J.B. (2000) Pathogenicity islands and the evolution of microbes. *Annu. Rev. Microbiol.*, **54**, 641–679.

Haft,D.H. *et al.* (2018) RefSeq: an update on prokaryotic genome annotation and curation. *Nucleic Acids Res.*, **46**, D851–D860.

Hudson,C.M. *et al.* (2015) Islander: a database of precisely mapped genomic islands in tRNA and tmRNA genes. *Nucleic Acids Res.*, **43**, D48–D53.

Jacomy,M. *et al.* (2014) ForceAtlas2, a continuous graph layout algorithm for handy network visualization designed for the Gephi software. *PLoS One*, **9**, e98679.

Jani,M. and Azad,R.K. (2019) IslandCafe: compositional anomaly and feature enrichment assessment for delineation of genomic islands. *G3 (Bethesda)*, **9**, 3273–3285.

Katz,L. *et al.* (2019) Mashtree: a rapid comparison of whole genome sequence files. *J. Open Source Softw.*, **4**, 1762.

Kittichotirat,W. *et al.* (2011) Identification of the pangenome and its components in 14 distinct *Aggregatibacter actinomycetemcomitans* strains by comparative genomic analysis. *PLoS One*, **6**, e22420.

Langille,M.G. *et al.* (2008) Evaluation of genomic island predictors using a comparative genomics approach. *BMC Bioinformatics*, **9**, 329.

Lawrence,J.G. and Ochman,H. (1997) Amelioration of bacterial genomes: rates of change and exchange. *J. Mol. Evol.*, **44**, 383–397.

Lescat,M. *et al.* (2009) A module located at a chromosomal integration hot spot is responsible for the multidrug resistance of a reference strain from *Escherichia coli* clonal group A. *Antimicrob. Agents Chemother.*, **53**, 2283–2288.

Lu,B. and Leong,H.W. (2016) Computational methods for predicting genomic islands in microbial genomes. *Comput. Struct. Biotechnol. J.*, **14**, 200–206.

Lu,B. and Leong,H.W. (2018) GI-Cluster: detecting genomic islands via consensus clustering on multiple features. *J. Bioinf. Comput. Biol.*, **16**, 1840010.

Mathee,K. *et al.* (2008) Dynamics of *Pseudomonas aeruginosa* genome evolution. *Proc. Natl. Acad. Sci. USA*, **105**, 3100–3105.

Niehus,R. *et al.* (2015) Migration and horizontal gene transfer divide microbial genomes into multiple niches. *Nat. Commun.*, **6**, 8924.

Ochman,H. *et al.* (2000) Lateral gene transfer and the nature of bacterial innovation. *Nature*, **405**, 299–304.

Ogier,J.-C. *et al.* (2010) Units of plasticity in bacterial genomes: new insight from the comparative genomics of two bacteria interacting with invertebrates, Photorhabdus and Xenorhabdus. *BMC Genomics*, **11**, 568.

Oliveira,P.H. *et al.* (2017) The chromosomal organization of horizontal gene transfer in bacteria. *Nat. Commun.*, **8**, 841.

Downloaded from https://academic.oup.com/bioinformatics/article/36/Supplement_2/i651/6055938 by Genoscope CNS user on 05 August 2021

Page,A.J. *et al.* (2015) Roary: rapid large-scale prokaryote pan genome analysis. *Bioinformatics*, **31**, 3691–3693.
Parks,D.H. *et al.* (2018) A standardized bacterial taxonomy based on genome phylogeny substantially revises the tree of life. *Nat. Biotechnol.*, **36**, 996–1004.
Pasolli,E. *et al.* (2019) Extensive unexplored human microbiome diversity revealed by over 150,000 genomes from metagenomes spanning age, geography, and lifestyle. *Cell*, **176**, 649–662.
Pundhir,S. *et al.* (2008) PredictBias: a server for the identification of genomic and pathogenicity islands in prokaryotes. *In Silico Biol.*, **8**, 223–234.
Rodriguez-Valera,F. and Ussery,D.W. (2012) Is the pan-genome also a pan-selectome? *F1000Res.*, **1**, 16.
Sayers,E.W. *et al.* (2019) GenBank. *Nucleic Acids Res.*, **47**, D94–D99.
Seemann,T. (2014) Prokka: rapid prokaryotic genome annotation. *Bioinformatics*, **30**, 2068–2069.
Snipen,L. and Liland,K.H. (2015) Micropan: an R-package for microbial pan-genomics. *BMC Bioinformatics*, **16**, 79.
Steinegger,M. and Söding,J. (2017) MMseqs2 enables sensitive protein sequence searching for the analysis of massive data sets. *Nat. Biotechnol.*, **35**, 1026–1028.
Tettelin,H. *et al.* (2005). Genome analysis of multiple pathogenic isolates of *Streptococcus agalactiae*: implications for the microbial "pan-genome". *Proc. Natl. Acad. Sci. USA*, **102**, 13950–13955.
Thomas,C.M. and Nielsen,K.M. (2005) Mechanisms of, and barriers to, horizontal gene transfer between bacteria. *Nat. Rev. Microbiol.*, **3**, 711–721.
Touchon,M. *et al.* (2009) Organised genome dynamics in the *Escherichia coli* species results in highly diverse adaptive paths. *PLoS Genet.*, **5**, e1000344.
Treangen,T.J. and Rocha,E.P. (2011) Horizontal transfer, not duplication, drives the expansion of protein families in prokaryotes. *PLoS Genet.*, **7**, e1001284.
UniProt Consortium (2019) UniProt: a worldwide hub of protein knowledge. *Nucleic Acids Res.*, **47**, D506–D515.
Vallenet,D. *et al.* (2009) MicroScope: a platform for microbial genome annotation and comparative genomics. *Database*, **2009**, bap021.
Vallenet,D. *et al.* (2019) MicroScope: an integrated platform for the annotation and exploration of microbial gene functions through genomic, pangenomic and metabolic comparative analysis. *Nucleic Acids Res.*, **48**, D579–D589.
Vernikos,G.S. and Parkhill,J. (2006) Interpolated variable order motifs for identification of horizontally acquired DNA: revisiting the Salmonella pathogenicity islands. *Bioinformatics*, **22**, 2196–2203.
Waack,S. *et al.* (2006) Score-based prediction of genomic islands in prokaryotic genomes using hidden Markov models. *BMC Bioinformatics*, **7**, 142.
Wei,W. *et al.* (2017) Zisland Explorer: detect genomic islands by combining homogeneity and heterogeneity properties. *Brief. Bioinform.*, **18**, 357–366.
Zhu,D. *et al.* (2019) Comparative analysis reveals the Genomic Islands in *Pasteurella multocida* population genetics: on Symbiosis and adaptability. *BMC Genomics*, **20**, 63.

Downloaded from https://academic.oup.com/bioinformatics/article/36/Supplement_2/i651/6055938 by Genoscope CNS user on 05 August 2021

7.3 Conclusion

panRGP a permis d'introduire une approche de génomique comparative capable de traiter des milliers de génomes pour identifier des RGP, et de regrouper ces RGP en spots d'insertion. C'est la première approche de génomique comparée capable de considérer autant de génomes dans le calcul en lui-même des RGP. C'est aussi la première approche qui a permis de reconstruire des spots d'insertions à cette échelle, même s'il est probable que l'approche de Oliveira et al., 2017 puisse, elle aussi, considérer autant de génomes.

L'ensemble a été inclus dans la suite logicielle PPanGGOLiN, ce qui facilite la distribution et l'usage de celle-ci, dans le contexte d'une analyse pangénomique. Néanmoins, il n'est pas possible de l'appliquer avec d'autres approches de partitionnement, puisqu'il est impossible d'importer un partitionnement dans PPanGGOLiN aujourd'hui. Le logiciel inclut aussi plusieurs fonctions qui permettent de dessiner les spots, ou d'en extraire le sous-graphe pour l'analyser, ainsi que des méthodes qui permettent de comparer des séquences protéiques à un pangénome pour en extraire les RGP et les spots dans lesquels ces protéines ont des homologues.

L'approche panRGP est aussi disponible au travers de la plateforme MicroScope mais dans une version moins élaborée de l'algorithme de prédiction de RGP. Cette fonctionnalité dans la plateforme MicroScope ainsi que d'autres sont décrites dans une publication pour laquelle je suis co-auteur (Vallenet et al., 2020).

Les RGP identifiées par panRGP dans un même spot se recoupent parfois partiellement ou totalement, et avec suffisamment de génomes, on voit se dessiner des chemins ou des blocs de gènes alternatifs dans un même spot d'insertion. Identifier ainsi ces blocs de gènes qui s'intègrent ensemble permettrait d'identifier automatiquement des groupes de gènes acquis et conservés ensemble, et donc potentiellement impliqués dans les mêmes fonctions. Le développement d'une approche, nommée panModule et permettant cela, est décrit dans le chapitre suivant.

Chapitre 8

panModule

8.1 Détection de modules conservés dans les régions variables des pangénomes

L'identification de modules génomiques conservés n'est pas un problème nouveau, mais peu de méthodes dédiées aux procaryotes existent et, encore moins, sont capables d'utiliser des milliers de génomes. Étudier ce problème dans le contexte d'un pangénome au niveau d'une espèce facilite grandement les choses : il y a moins de variabilité liée à l'environnement et aux contextes génomiques différents dans lesquels on peut retrouver des synténies conservés. De plus, l'étape coûteuse des comparaisons a déjà été réalisée lors de la construction du pangénome.

Plusieurs articles ont déjà souligné l'organisation modulaire des îlots génomiques et des spots d'insertion, notamment Lescat et al., 2009 sur le spot *leuX* illustré par la figure 8.1.

La méthode panModule introduit une approche qui permet d'identifier des modules conservés au sein d'un pangénome, en utilisant des milliers de génomes d'une même espèce. Cette approche a été testée contre des données expertisées de modules dans 12 génomes de *Escherichia coli*, puis son utilité a été illustrée dans le cadre de l'analyse d'une région de pathogénicité chez *Klebsiella pneumoniae* 1084.

La méthode panModule a été intégrée dans la suite logicielle PPanGGOLiN. Ses résultats peuvent ainsi être étudiés dans le contexte des autres analyses disponibles dans PPanGGOLiN, puisqu'on peut aisément identifier sur quelles RGP et dans quels spots d'insertion les modules sont retrouvés.

8.2 Article 3 : panModule : detecting conserved modules in the variable regions of a pangenome graph

panModule: detecting conserved modules in the variable regions of a pangenome graph

Adelme Bazin [1] , Claudine Medigue [1] , David Vallenet [1*†], Alexandra Calteau [1 *†]

[1]LABGeM, Génomique Métabolique, CEA, Genoscope, Institut François Jacob, Université d'Évry, Université Paris-Saclay, CNRS, Evry, France

Abstract

The recent years have seen the rise of pangenomes as comparative genomic tools to better understand the evolution of gene content among microbial genomes in close phylogenetic groups such as species. While the core or persistent genome is often well-known as it includes essential or ubiquitous genes, the variable genome is usually less characterized and includes many genes with unknown functions even among the most studied organisms. It gathers important genes for strain adaptation that are acquired by horizontal gene transfer. Here, we introduce panModule, an original method to identify conserved modules in pangenome graphs built from thousands of microbial genomes. These modules correspond to synteny blocks composed of consecutive genes that are conserved in a subset of the compared strains. Identifying conserved modules can provide insights on genes involved in the same functional processes, and as such is a very helpful tool to facilitate the understanding of genomic regions with complex evolutionary histories. The panModule method was benchmarked on a curated dataset of conserved modules in *Escherichia coli* genomes. Its use was illustrated through a study of a high pathogenicity island in *Klebsiella pneumoniae* that allowed a better understanding of this region. panModule is freely available and accessible through the PPanGGOLiN software suite (`https://github.com/labgem/PPanGGOLiN`).

1 Introduction

Lately, the data deluge provided by NGS has given access to over a million of prokaryotic genome sequences in public data banks, as well as a wealth of genomes reconstructed from environmental data, such as metagenome assembled genomes (MAGs) or single-cell assembled genomes (SAGs). Consequently,

*To whom correspondence should be addressed. Emails: acalteau@genoscope.cns.fr, vallenet@genoscope.cns.fr

†The authors wish it to be known that, in their opinion, the last two authors should be regarded as Joint Last Authors.

many bacterial species of interest now have hundreds to several thousands of genomes publicly available. It represents a fantastic opportunity to understand the evolution of prokaryotic genomes, and more specifically to study gene flow within a given species. However, processing such a huge amount of data for comparative genomics analyses is becoming a real challenge and requires new bioinformatics solutions. In particular, many new methods rely on the conceptual framework of the pangenome, which corresponds to the entire gene repertoire of a taxonomic group [1]. Multiple methods have been developed to study pangenome structures and to perform comparative studies of hundreds to thousands of genomes (e.g. Roary [2], PIRATE [3], Panaroo [4], PPanGGOLiN [5], PanACoTA [6]). A pangenome can be described by two components: the core genome, which contains genes shared by all (or almost all) individuals, and the variable genome, which gathers all other genes.

Studying the variable part of the pangenome is of major interest since it gathers genes of importance for adaptation of the strains, and particularly genes that have been acquired by Horizontal gene transfer (HGT). Indeed, as described for many years, HGT is a significant source of gene novelty [7] and is a major driver of genome evolution in bacterial species providing and maintaining diversity at the population level [8, 9]. This horizontal gene flow occurs through different well-known mechanisms involving mobile genetic elements and can spread genes between potentially very distant bacterial lineages [10]. Transferred genetic material can correspond to single genes as well as clusters of consecutive genes on the chromosome from one or more transfer events. These latter are commonly described as genomic islands (GIs) [11]. They may bring an evolutionary advantage allowing adaptations to new environments or providing new pathogenicity capacities, for instance [12]. Studying the evolution and functional impact of GIs on bacterial populations is of major interest for microbiologists, since they are widely distributed in pathogenic and environmental microorganisms. GIs tend to insert at specific sites of the genome, such as tRNA genes [13]. Some of those insertion sites, called hotspots, are more active than the rest of the genome in terms of acquisition rate of new elements and have a much more diverse gene content, even between closely related individuals [14]. Indeed, hotspots diversify by rapid gene turnover driven by homologous recombination and horizontal gene transfer.

In the framework of the study of several *Escherichia coli* genomes, two publications have described the structure of GIs at a given hotspot [15, 16]. They showed a patchy structure corresponding to a segmented organization of genes into modules that can be found independently in different genome loci. These modules correspond to synteny blocks composed of consecutive genes that are conserved in a subset of the compared strains and can be functionally linked. In the *pheV*-tRNA hotspot of *E. coli*, the presence/absence of modules was shown to be uncorrelated with either the phylogenetic group or the pathotype [15]. Several other studies have described the modular structures of GIs in various species, such as in *Klebsiella pneumoniae* [17] or in *Photorhabdus/Xenorhabdus* [18]. Many methods to identify conserved synteny regions in prokaryotic genomes have been developed through the years. However, most of

them do not scale higher than a few dozen genomes. To our knowledge, only two are designed for prokaryotes and have the ability to cope with more than a few hundred genomes, namely Gecko3 [19] and CSBFinder [20, 21]. While both of those methods provide an answer to the problem of finding conserved syntenies [22], they are not designed to work with pangenomes but rather on a more diverse taxonomic selection of genomes. Other approaches that do not rely on synteny conservation but on the cooccurrence or coevolution of genes in a pangenome to infer functional associations exist as well. Pantagruel [23] or Coinfinder [24] are recent examples designed for bacterial pangenomes. However, their goal is to infer associations or dissociations between pairs of genes, and they do not attempt to refine those relations into groups of colocalized genes directly.

In this article, we propose a new method called panModule to detect modules in GIs based on the pangenome graph representation of PPanGGOLiN [5]. In this graph, nodes represent gene families and edges represent genetic contiguity. In PPanGGOLiN, families are classified by a statistical model into a tri-partition scheme as introduced by [25]: (i) the persistent genome, which corresponds to genes that are present in most individuals of the studied clade, (ii) the shell genome, which groups genes that are conserved between some individuals of the group but not most and (iii) the cloud genome, which corresponds to genes that are rare within the population and found only in one or a few individuals. To predict modules, the panModule algorithm detects sets of co-occurring and colocalized genes in the variable part of the pangenome graph, composed of shell and cloud families. The predictions of panModule were evaluated at the species level using a curated dataset of modules in 12 genomes of *Escherichia coli* previously analyzed [15]. We then illustrate our approach by predicting modules in a set of complete *Klebsiella pneumoniae* genomes and comparing those predictions with formerly studied modules in a high pathogenicity island (HPI) of a strain of medical interest [17]. The panModule method is integrated in the PPanGGOLiN software suite (`https://github.com/labgem/PPanGGOLiN`). Modules can be predicted on pangenomes made of thousands of genomes and be analyzed along with the GIs and integration spot results from the previously published panRGP method [26].

2 Materials and Methods

2.1 Module detection algorithm

The panModule method uses the pangenome graph representation of PPanGGOLiN [5] in which nodes correspond to homologous gene families (classified into *persistent*, *shell* and *cloud* partitions) and edges represent genetic contiguity (two families are linked in the graph if they contain genes that are neighbors in the genomes). Modules are defined as non-overlapping sets of cooccurring and colocalized variable gene families (*i.e.*, *shell* and *cloud* families) that correspond to connected components in the pangenome graph. The graph algorithm

behind panModule is inspired from a previously published method [27] that merges information from two or more graphs in a multigraph and then detects common connected components. These components correspond to conserved synteny groups, where the input graphs are genomes. In panModule, we do not use a multigraph representation but directly the pangenome graph to detect synteny conservation and thus modules in the variable genome.

First, each annotated genome sequence (*i.e.* complete genome sequences or contigs) is read to obtain a genome graph, which is a linear graph of genes that is cyclic in case of complete sequences of circular plasmids or chromosomes. Then, the pangenome graph $G(V, E)$ is built from all genome graphs where V is a set of vertices corresponding to gene families and E is a set of edges representing genetic contiguity in the genome graphs between those gene families. An edge $e_{i,j}$ is added between each pair of gene families (v_i, v_j) if their corresponding genes are separated by less than t genes on a genome graph. For $t \geqslant 1$, it corresponds to a transitive closure applied to the genome graphs and allows to connect two families even if their genes are not directly adjacent. This can be useful in case a module in a genome is interrupted by a gene insertion (e.g. an insertion sequence) or has lost genes due to a deletion or pseudogenization event.

For each edge $e_{i,j}$, two Jaccard similarity coefficients are computed as follows: $J(v_i, e_{i,j}) = \frac{w_{e_{i,j}}}{w_{v_i}}$ and $J(v_j, e_{i,j}) = \frac{w_{e_{i,j}}}{w_{v_j}}$ where w_{v_i} and w_{v_j} are the number of genes associated with the families v_i and v_j, respectively, and $w_{e_{i,j}}$ is the number of pairs of genes used to create an edge $e_{i,j}$ between two nodes v_i and v_j. A threshold s is defined as being the minimal Jaccard similarity to consider an edge as belonging to a module. If $J(v_i, e_{i,j}) \geqslant s \wedge J(v_j, e_{i,j}) \geqslant s$, the edge is kept, otherwise it is removed from the graph. In addition, nodes corresponding to gene families that are present in less than m genomes are also removed.

After this filtering step, each connected component is extracted using a modified Breadth-First Search (BFS) algorithm. A connected component is considered as a predicted module if it contains at least 3 nodes made of the *shell*, *cloud* or multigenic *persistent* families according to PPanGGOLiN classification [5]. Indeed, modules containing non-multigenic *persistent* families are not considered because they are not part of the variable genome and generally correspond to long syntenic regions that are conserved in almost all compared genomes without any or only a few rearrangement events.

2.2 Reference dataset and genome collections

To evaluate panModule predictions, we used a reference dataset of modules based on the expert annotation of 12 *E. coli* complete genomes originally published in [15]. These 12 strains are from different phylogroups (*i.e.* A, B1, B2 and D) and are commensal or ExPEC (Extra-intestinal Pathogenic *Escherichia coli*). Their genomes have been curated with the MicroScope platform [28] and their GIs have been divided into modules according to both synteny conservation and functional annotation of the genes. This dataset, named EcoliRef,

contains a total of 165 modules that are present in at least 2, and not more than 10 of the 12 genomes, for a total of 793 occurrences in 461 GIs. GIs and modules are described by their genomic coordinates on the 12 *E. coli* chromosome sequences and were classified into 7 functional categories (Supplementary Data file 'benchmark/reference_modules.tsv').

Module prediction was run on 4 collections of *E. coli* genomes. The first one corresponds to the 12 genomes of the EcoliRef reference dataset. The second one contains all 1671 *E. coli* genomes classified as 'Complete' or 'Chromosome' in NCBI RefSeq [29] (downloaded the 1st of March 2021 and listed in Supplementary Data file 'benchmark/EcoliComplete_genomes.list'). This dataset will be thereafter called EcoliComplete and includes the 12 genomes of EcoliRef. The third collection, named EcoliContigs, includes 1659 unfinished genomes plus the 12 genomes of EcoliRef. All *E. coli* genomes classified as 'Contigs' or 'Scaffold' in NCBI RefSeq (downloaded the 22nd of June 2021) were compared to all EcoliComplete genomes (apart from the 12 genomes of EcoliRef) with Mash (version 2.1.1, parameters: -s 5000, default for the others) [30] to iteratively pick out the closest genome in contigs (Supplementary Data file 'benchmark/EcoliContigs_genomes.list'). As such, we get an equivalent bias in genome composition between both datasets to compare them properly and thus evaluate the impact of using genomes in contigs for module prediction. The last collection, named EcoliMAGs, contains 1 416 MAGs classified as '*Escherichia coli*' from the study of Pasolli *et al.* [31] plus the 12 genomes of EcoliRef. MAGs were annotated using bakta (version 1.0.3, default parameters) [32]. As this dataset is much smaller than NCBI RefSeq, we chose not to apply the same filters as for the EcoliContigs dataset and just kept all MAGs. Therefore, we have here a collection of incomplete and fragmented *E. coli* genomes whose diversity is different from the two previous ones. Indeed, it contains potentially fewer pathogenic strains because MAGs were obtained from metagenomic samples not involving patients with an *E. coli* infections.

To illustrate the use of panModule on another species, a dataset containing 566 complete genomes classified as 's__Klebsiella pneumoniae' in GTDB (release 06-RS202) [33] was downloaded on the 3rd of May 2021 from GenBank and NCBI RefSeq (listed in Supplementary Data file 'Klebsiella_pneumoniae/Klebsiella_pneumoniae_genomes.list'). Then the modules were predicted using the method described in 2.1 with parameters $t = 4$, $m = 2$ and $s = 0.86$. The GIs of interest were identified by searching for those with positions overlapping with KPHPI208 in *Klebsiella pneumoniae* 1084. Annotations considered in the analysis are those from the downloaded file of NCBI RefSeq.

Genomic region illustrations were obtained using an online version of the CGView software [34] available at `https://beta.proksee.ca`.

2.3 Module prediction and benchmark procedure

The panModule method was run on the four *E. coli* genome collections using PPanGGOLiN software version 1.2.0 with default parameters to obtain homologous gene families and pangenome graphs. Module prediction was evaluated

at the genome level using the reference modules from the 12 genomes of the EcoliRef dataset. In a GI of a given genome, a set of genes is considered to be part of a predicted module if their corresponding families are members of the same module. The benchmark uses genomic positions and processes GIs one by one with the following assumptions: (i) pairs of genes located between the genomic positions of a reference module are considered as positive relations (ii) pairs of genes that do not belong to the same reference module but are in the same GI are considered as negative relations. Thus, pairs of genes in a GI that are in the same predicted module and the same reference module are True Positives (TP). Pairs of genes that are in different predicted modules or not in a predicted module but in the same reference module are False Negatives (FN). Pairs of genes that are in the same predicted module but in different reference modules or not in a reference module are False Positives (FP). Pairs of genes that are in different predicted modules or not in a predicted module and in different reference modules or not in a reference module are True Negatives (TN).

To evaluate module prediction, Matthew's correlation coefficient (MCC), F1-score, accuracy, precision and recall values were computed for the 4 *E. coli* genome collections as follows:

$$
\begin{aligned}
recall &= \frac{TP}{TP+FN} \\
precision &= \frac{TP}{TP+FP} \\
accuracy &= \frac{TP+TN}{TP+FP+FN+TN} \\
F1score &= \frac{2TP}{2TP+FP+FN} \\
MCC &= \frac{TP*TN-FP*FN}{\sqrt{(TP+FP)(TP+FN)(TN+FP)(TN+FN)}}
\end{aligned}
$$

In order to determine the best values for the s, m and t parameters of panModule, we evaluated the predictions on the EcoliRef dataset using a set of realistic value combinations and chose the one that gives the best MCC (Supplementary Data file 'benchmark/EcoliScope_benchmark_metrics.tsv'). We then used this set of parameters on the other genome collections for comparison.

3 Results & Discussion

3.1 Benchmark results

To evaluate panModule, we ran the benchmark as described in the Materials and Methods section to see how it performed against a curated set of functional modules. The best set of parameters was estimated on the EcoliRef dataset, and applied on the other datasets as such: $m = 2$, $t = 4$ and $s = 0.86$. A summary

of the benchmark results for each dataset with these parameters is available in Table 1. It is possible that other parameters yield better results, which we will discuss with the case of the EcoliMAGs dataset.

The figure 1 represents the modules predicted with the different datasets on a plastic region of the genome of *Escherichia coli* 536 with the set of parameters previously mentioned.

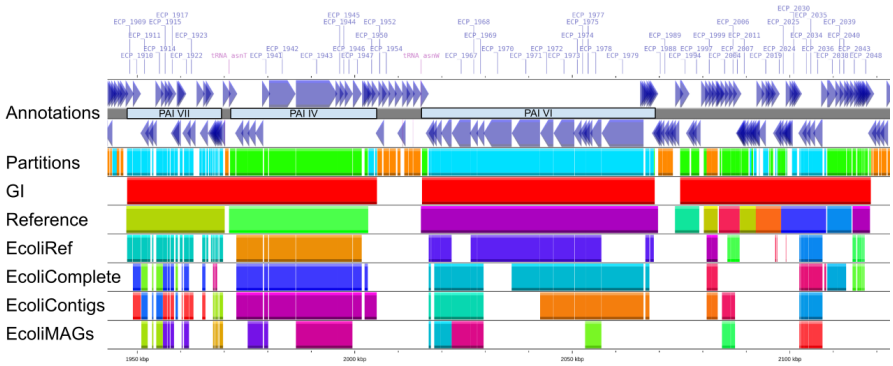


Figure 1: Modularized plastic region of *Escherichia coli* 536

Each track in the figure is a layer of information about genomic features in a region of the chromosome of *E. coli* 536 (accession CP000247.1). The 1st and 2nd tracks indicate gene positions and orientation. In between them are indicated the known pathogenicity islands in the region, respectively PAI-VII, PAI-IV and PAI-VI [35]. The 3rd track indicates for each gene its pangenome partition from the EcoliRef dataset by a color code: orange for *persistent*, green for *shell* and cyan for *cloud*. The 4th track indicates GI positions. The 5th track indicates reference modules. Tracks 6 to 9 show the module predictions using the different datasets (EcoliRef, EcoliComplete, EcoliContigs and EcoliMAGs). Colors for tracks 5 to 9 are selected randomly, and genes belonging to the same modules are colored identically in each track.

Overall, the modules detected by our approach in the EcoliRef dataset (*i.e.* the same set of genomes that were used for the expert annotation of modules) compare favorably to the reference modules with MCC and F1 score values of 0.63 and 0.66, respectively. The precision and accuracy values (0.96 and 0.85, respectively) are particularly high. This indicates that most of the genes belonging to a predicted module are found together in a reference module (True Positives) and that few genes are wrongly associated (False Positives). Similarly, most of the genes in a GI that do not belong to a same or any reference module are also not grouped into a module by our method (True Negatives). On the other hand, the recall value (0.51) is much lower, indicating that while we do recover proper modules a non-negligible number of the genes belonging to a reference module are not covered by our predictions (False Negatives). In summary, using the EcoliRef dataset, the modules predicted by panModule

Metric	EcoliRef	EcoliComplete	EcoliContigs	EcoliMAGs	EcoliMAGs ($s = 0.49$)
Genomes	12	1671	1671	1428	1428
Families	8867	57346	64318	44045	44045
Shell families	748	5542	5524	2410	2410
MCC	0.63	0.56	0.55	0.34	0.56
F1 score	0.66	0.59	0.60	0.31	0.60
Accuracy	0.85	0.83	0.83	0.76	0.83
Precision	0.96	0.93	0.89	0.90	0.90
Recall	0.51	0.43	0.45	0.19	0.45
Gene coverage	69.9%	64.6%	64.0%	39.2%	61.5%
Module coverage	79.9%	74.9%	73.8%	48.7%	68.0%
Predicted modules	219	1839	1516	1119	1485
Families in modules	1379	11558	9047	5473	11434
Shell in modules	61.9%	48.0%	42.2%	33.2%	54.8%

Table 1: panModule results and benchmark on the four *E. coli* genome collections

For each *E. coli* dataset, the number of genomes and pangenome gene families is given along with the number of families in the *shell*. Benchmark results are provided with two additional metrics (gene coverage and module coverage) corresponding to the percentage of genes and modules of the reference dataset that are associated to a predicted module. The benchmark was made using $s = 0.86$ except for the last column where $s = 0.49$. The number of predicted modules, their gene families and the percentage of shell families in modules are also provided.

are frequently included in or equal to the reference modules and rarely overlap multiple reference modules, but some reference modules may be missed or be incomplete. Those missing genes represent about 30% of the genes associated to a reference module, and about 20% of the reference modules are not covered by any prediction (see Table 1 and Figure 1 as an illustration).

Both EcoliComplete and EcoliContigs datasets show equivalent results to the EcoliRef dataset but with slightly lower metrics overall, with the recall being the most impacted (*i.e.* 0.43 and 0.45, respectively, versus 0.51 for the EcoliRef dataset). It is further amplified when looking at the EcoliMAGs dataset whose metrics are much lower, especially in terms of recall, where it reaches a staggering value of 0.19, meaning that most of the reference modules are not predicted. Figure 1 clearly displays the sparsity of predictions in that specific dataset. However, it still keeps an acceptable precision and accuracy with 0.90 and 0.76, respectively, indicating that predicted modules even with the most incomplete and fragmented genome datasets are often actual modules. Indeed, the method parameters estimated on the EcoliRef dataset are likely too stringent to analyze MAGs as many modules may be incomplete or split on several contigs. We looked for the Jaccard similarity (s) parameter providing the best MCC using the same values for t and m (Supplementary Data file 'benchmark/EcoliMAGs_benchmark_metrics.tsv'). The best value of s is 0.49, and gives a MCC value of 0.56 which is equivalent to the EcoliComplete and EcoliContigs datasets. Even using such a low s value, accuracy and precision remain high. Those results indicate that our method is applicable on very incomplete and fragmented genome datasets such as MAGs using a relaxed Jaccard similarity threshold.

This benchmark shows that panModule is able to correctly predict modules with a very good accuracy and precision even with incomplete genomes. It should be noticed that this validation is based only on a limited subset of curated data made of 12 *E. coli* genomes of which 6 are from the B2 phylogroup. Therefore, this reference dataset does not capture the overall diversity of *E. coli*. Nevertheless, panModule can modularize a large fraction of the *shell* pangenome of large and diverse genome datasets as well (e.g. 48% of the shell families are predicted in a module for the EcoliComplete dataset). It would have been interesting to validate the method on a larger amount of reference data and also on other species but we did not find such available resources.

3.2 Analysis of the KPHPI208 genomic island in *Klebsiella pneumoniae*

To illustrate the potential of panModule on other species, we chose to reanalyze the KPHPI208 GI of *Klebsiella pneumoniae* 1084. This 208-kb GI inserted at the *asn*-tRNA loci was described to be composed of 8 genomic modules (GM1 to GM8) using comparative genomics [17].

First, we used panRGP to predict GIs in *K. pneumoniae* 1084 from a pangenome made of 566 genomes. Two predicted GIs correspond to KPHPI208. The first detected GI starts at 1,744,478 bp and stops at 1,906,798 bp while the

second one starts at 1,920,068 bp and stops at 1,952,189 bp. The two GIs cover the majority of the region, with the remaining portion being persistent genes between the two islands. In the KPHPI208 region made of 135 genes, panModule predicted 9 modules including a total of 98 genes. An overall picture of the region is represented in Figure 2 and Table 2 summarizes the correspondences between the formerly published and the predicted modules. The genomic positions of each described GMs and predicted modules are given in Supplementary Data file 'Klebsiella_pneumoniae/Klebsiella_modules.tsv'.

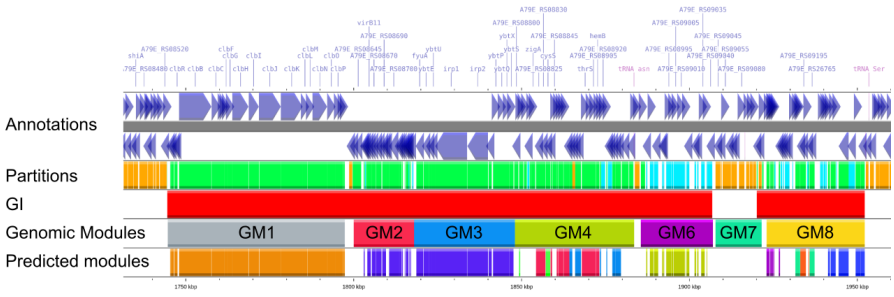


Figure 2: KPHPI208 genomic island in *Klebsiella pneumoniae* 1084
Each track in the figure is a layer of information about genomic features of the KPHPI208 genomic island. The 1st and 2nd tracks indicate gene positions and orientation. The 3rd track indicates for each gene its pangenome partition by a color code: orange for *persistent*, green for *shell* and cyan for *cloud*. The 4th track indicates GI predicted by panRGP. The 5th track indicates Genomic Modules (named GM1 to GM8). The 6th track indicates modules predicted by panModule.

Overall, our approach confirms the GMs that were functionally described by the authors, and provides novel insights for those that were uncharacterized. Only GM5 and GM7 were not predicted by panModule since they are composed of persistent genes. The predicted Module 12 is perfectly identical to the GM1 which codes for colibactin. Module 17 is extremely similar to GM6, only 2 genes which are likely unrelated to microcin biosynthesis were excluded from panModule predictions. Finding both GM2 (VirB) and GM3 (yersiniabactin) in the same predicted Module 13 indicates that they are conserved together in most *K. pneumoniae* genomes and suggests that they were exchanged together in the evolution of this species. Only two genes involved in the VirB secretion system are absent from the predicted module because they are not found conserved with other VirB system genes in a large number of genomes.

Among GMs that were described as unknown, GM4 and GM8 are composed of multiple modules predicted by panModule. Regarding GM4, we noticed that the genes that are in the same modules have similar annotations. No clear function could be inferred from Module 14. However, Module 15 contains mainly genes encoding enzymes and as such may be a metabolic pathway whereas Mod-

Genomic Modules	panModule prediction	Functional annotation	genes in modules / all genes	Pangenome module occurrence
GM1	Module 12	Colibactin	21 / 21	46
GM2	Module 13	VirB secretion system	15 / 24	322
GM3		Yersiniabactin	11 / 12	
GM4	Module 14	Unknown	22 / 34	85
	Module 15	Unknown enzymes		148
	Module 16	ABC transporter		81
GM5	None	Unknown	0 / 1	NA
GM6	Module 17	Microcin	15 / 17	34
GM7	None	Unknown	0 / 12	NA
GM8	Module 18	Unknown	18 / 31	85
	Module 19	Unknown		100
	Module 20	Unknown		66
	Module 21	Unknown		83

Table 2: Comparison between genomic modules of the KPHPI208 genomic island and panModule predictions
Genomic modules (GM) corresponds to the modules that were characterized in the original publication. The panModule prediction column contains the correspondences between the predicted modules and their GMs. Functional annotation column provides a brief summary of the module functions based on the gene annotation. Genes in modules / all genes column indicates how many genes are classified in a predicted module among all the genes in the original GM. Pangenome module occurrence column gives the number of genomes with the predicted module in the pangenome.

ule 16 gathers proteins that are mostly related to an ABC transporter. Those common annotations are a strong indication that both are proper functional modules. For GM8, its analysis was less straightforward because most of the gene functions are either unknown or very generic and do not appear to be related to each other. Although we do not have much information in terms of functional annotation, the detected modules can be used as a basis for experimental studies to determine the biological processes in which they are involved.

Looking more broadly at the pangenome level, a total of 315 out of 566 *K. pneumoniae* genomes have a predicted variable region in the same integration spot as the one of GM1-GM6 region of the 1084 strain. It is a highly variable region, as there are 128 different organizations among the 315 genomes, with 108 different combinations of modules. The KPHPI208 organization appears only once in the pangenome as such, but 13 other genomes include all of those modules as well. A dynamic visualization of those 128 organizations with their module composition was generated by PPanGGOLiN and is available in Supplementary Data file 'Klebsiella_pneumoniae/spot_19.html'.

4 Conclusion

We presented a novel method, named panModule, which groups genes into modules among thousands of genomes. Our approach uses a partitioned pangenome graph, which makes large-scale comparisons easier to compute. We benchmarked it against a curated set of *E. coli* genomes which were expertly annotated and whose GIs were divided into modules. We showed that panModule predictions were quite reliable regarding those annotations even for incomplete genomes. We illustrated the usefulness of our approach by revisiting the curated annotation of a genomic island in *K. pneumoniae* 1084. Overall, we believe that panModule provides an original approach to identify conserved modules in the variable regions of genomes, which may help to determine their function but also to better understand their complex evolutionary history.

The panModule method is freely available and easily installable as part of the PPanGGOLiN software suite, and can therefore be coupled with the other tools provided by the software, such as the analysis of pangenome partitions, the detection of GIs and spots of insertion. A potential improvement of the method presented here could be to include information from the species phylogeny in the computation of modules. Indeed, the calculation of Jaccard similarity could be weighted by a phylogenetic distance to favor the grouping of genes from distant strains into modules.

5 Data Availability

All mentioned Supplementary Data files and scripts to compute the benchmark are available at `https://github.com/axbazin/panmodule-supplementary`. The software is available at `https://github.com/labgem/PPanGGOLiN`.

6 Funding

This research was supported in part by the Phare PhD program of the French Alternative Energies and Atomic Energy Commission (CEA) for A.B.

References

[1] Hervé Tettelin, Vega Masignani, Michael J. Cieslewicz, Claudio Donati, Duccio Medini, Naomi L. Ward, Samuel V. Angiuoli, Jonathan Crabtree, Amanda L. Jones, A. Scott Durkin, Robert T. DeBoy, Tanja M. Davidsen, Marirosa Mora, Maria Scarselli, Immaculada Margarit y Ros, Jeremy D. Peterson, Christopher R. Hauser, Jaideep P. Sundaram, William C. Nelson, Ramana Madupu, Lauren M. Brinkac, Robert J. Dodson, Mary J. Rosovitz, Steven A. Sullivan, Sean C. Daugherty, Daniel H. Haft, Jeremy Selengut, Michelle L. Gwinn, Liwei Zhou, Nikhat Zafar, Hoda Khouri, Diana Radune, George Dimitrov, Kisha Watkins, Kevin J. B. O'Connor, Shannon Smith, Teresa R. Utterback, Owen White, Craig E. Rubens, Guido Grandi, Lawrence C. Madoff, Dennis L. Kasper, John L. Telford, Michael R. Wessels, Rino Rappuoli, and Claire M. Fraser. Genome analysis of multiple pathogenic isolates of streptococcus agalactiae: Implications for the microbial "pan-genome". *Proceedings of the National Academy of Sciences*, 102 (39):13950–13955, 2005. ISSN 0027-8424. doi: 10.1073/pnas.0506758102.

[2] Andrew J Page, Carla A Cummins, Martin Hunt, Vanessa K Wong, Sandra Reuter, Matthew TG Holden, Maria Fookes, Daniel Falush, Jacqueline A Keane, and Julian Parkhill. Roary: rapid large-scale prokaryote pan genome analysis. *Bioinformatics*, 31(22):3691–3693, 2015.

[3] Sion C Bayliss, Harry A Thorpe, Nicola M Coyle, Samuel K Sheppard, and Edward J Feil. Pirate: A fast and scalable pangenomics toolbox for clustering diverged orthologues in bacteria. *Gigascience*, 8(10):giz119, 2019.

[4] Gerry Tonkin-Hill, Neil MacAlasdair, Christopher Ruis, Aaron Weimann, Gal Horesh, John A Lees, Rebecca A Gladstone, Stephanie Lo, Christopher Beaudoin, R Andres Floto, Simon D.W. Frost, Jukka Corander, Stephen D. Bently, and Julian Parkhill. Producing polished prokaryotic pangenomes with the panaroo pipeline. *Genome biology*, 21(1):1–21, 2020.

[5] Guillaume Gautreau, Adelme Bazin, Mathieu Gachet, Rémi Planel, Laura Burlot, Mathieu Dubois, Amandine Perrin, Claudine Médigue, Alexandra Calteau, Stéphane Cruveiller, Catherine Matias, Christophe Ambroise, Eduardo P. C. Rocha, and David Vallenet. Ppanggolin: Depicting microbial diversity via a partitioned pangenome graph. *PLOS Computational Biology*, 16(3):1–27, 03 2020. doi: 10.1371/journal.pcbi.1007732.

[6] Amandine Perrin and Eduardo PC Rocha. Panacota: A modular tool for massive microbial comparative genomics. *NAR genomics and bioinformatics*, 3(1):lqaa106, 2021.

[7] Todd J Treangen and Eduardo PC Rocha. Horizontal transfer, not duplication, drives the expansion of protein families in prokaryotes. *PLoS genetics*, 7(1):e1001284, 2011.

[8] Rene Niehus, Sara Mitri, Alexander G Fletcher, and Kevin R Foster. Migration and horizontal gene transfer divide microbial genomes into multiple niches. *Nature communications*, 6(1):1–9, 2015.

[9] Howard Ochman, Jeffrey G Lawrence, and Eduardo A Groisman. Lateral gene transfer and the nature of bacterial innovation. *nature*, 405(6784): 299–304, 2000.

[10] Christopher M Thomas and Kaare M Nielsen. Mechanisms of, and barriers to, horizontal gene transfer between bacteria. *Nature reviews microbiology*, 3(9):711–721, 2005.

[11] Jörg Hacker and James B Kaper. Pathogenicity islands and the evolution of microbes. *Annual Reviews in Microbiology*, 54(1):641–679, 2000.

[12] Jörg Hacker and Elisabeth Carniel. Ecological fitness, genomic islands and bacterial pathogenicity. *EMBO reports*, 2(5):376–381, 2001.

[13] Ulrich Dobrindt, Bianca Hochhut, Ute Hentschel, and Jörg Hacker. Genomic islands in pathogenic and environmental microorganisms. *Nature Reviews Microbiology*, 2(5):414–424, 2004.

[14] Pedro H Oliveira, Marie Touchon, Jean Cury, and Eduardo PC Rocha. The chromosomal organization of horizontal gene transfer in bacteria. *Nature communications*, 8(1):1–11, 2017.

[15] Marie Touchon, Claire Hoede, Olivier Tenaillon, Valérie Barbe, Simon Baeriswyl, Philippe Bidet, Edouard Bingen, Stéphane Bonacorsi, Christiane Bouchier, Odile Bouvet, Alexandra Calteau, Hélène Chiapello, Olivier Clermont, Stéphane Cruveiller, Antoine Danchin, Médéric Diard, Carole Dossat, Meriem El Karoui, Eric Frapy, Louis Garry, Jean Marc Ghigo, Anne Marie Gilles, James Johnson, Chantal Le Bouguénec, Mathilde Lescat, Sophie Mangenot, Vanessa Martinez-Jéhanne, Ivan Matic, Xavier Nassif, Sophie Oztas, Marie Agnès Petit, Christophe Pichon, Zoé Rouy, Claude Saint Ruf, Dominique Schneider, Jérôme Tourret, Benoit Vacherie, David Vallenet, Claudine Médigue, Eduardo P. C. Rocha, and Erick Denamur. Organised genome dynamics in the escherichia coli species results in highly diverse adaptive paths. *PLOS Genetics*, 5(1):1–25, 01 2009. doi: 10.1371/journal.pgen.1000344.

[16] Mathilde Lescat, Alexandra Calteau, Claire Hoede, Valérie Barbe, Marie Touchon, Eduardo Rocha, Olivier Tenaillon, Claudine Médigue, James R Johnson, and Erick Denamur. A module located at a chromosomal integration hot spot is responsible for the multidrug resistance of a reference strain from escherichia coli clonal group a. *Antimicrobial agents and chemotherapy*, 53(6):2283–2288, 2009.

[17] Yi-Chyi Lai, Ann-Chi Lin, Ming-Ko Chiang, Yu-Han Dai, Chih-Chieh Hsu, Min-Chi Lu, Chun-Yi Liau, and Ying-Tsong Chen. Genotoxic klebsiella pneumoniae in taiwan. *PloS one*, 9(5):e96292, 2014.

[18] Jean-Claude Ogier, Alexandra Calteau, Steve Forst, Heidi Goodrich-Blair, David Roche, Zoé Rouy, Garret Suen, Robert Zumbihl, Alain Givaudan, Patrick Tailliez, Claudine Médigue, and Sophie Gaudriault. Units of plasticity in bacterial genomes: new insight from the comparative genomics of two bacteria interacting with invertebrates, photorhabdus and xenorhabdus. *BMC genomics*, 11(1):1–21, 2010.

[19] Sascha Winter, Katharina Jahn, Stefanie Wehner, Leon Kuchenbecker, Manja Marz, Jens Stoye, and Sebastian Böcker. Finding approximate gene clusters with gecko 3. *Nucleic acids research*, 44(20):9600–9610, 2016.

[20] Dina Svetlitsky, Tal Dagan, Vered Chalifa-Caspi, and Michal Ziv-Ukelson. Csbfinder: discovery of colinear syntenic blocks across thousands of prokaryotic genomes. *Bioinformatics*, 35(10):1634–1643, 2019.

[21] Dina Svetlitsky, Tal Dagan, and Michal Ziv-Ukelson. Discovery of multi-operon colinear syntenic blocks in microbial genomes. *Bioinformatics*, 36 (Supplement_1):i21–i29, 2020.

[22] Cristina G Ghiurcuta and Bernard ME Moret. Evaluating synteny for improved comparative studies. *Bioinformatics*, 30(12):i9–i18, 2014.

[23] Florent Lassalle, Philippe Veber, Elita Jauneikaite, and Xavier Didelot. Automated reconstruction of all gene histories in large bacterial pangenome datasets and search for co-evolved gene modules with pantagruel. *BioRxiv*, page 586495, 2019.

[24] Fiona Jane Whelan, Martin Rusilowicz, and James Oscar McInerney. Coinfinder: detecting significant associations and dissociations in pangenomes. *Microbial genomics*, 6(3), 2020.

[25] Eugene V. Koonin and Yuri I. Wolf. Genomics of bacteria and archaea: the emerging dynamic view of the prokaryotic world. *Nucleic Acids Research*, 36(21):6688–6719, 10 2008. ISSN 0305-1048. doi: 10.1093/nar/gkn668.

[26] Adelme Bazin, Guillaume Gautreau, Claudine Médigue, David Vallenet, and Alexandra Calteau. panRGP: a pangenome-based method to predict genomic islands and explore their diversity. *Bioinformatics*, 36

(Supplement_2):i651–i658, 12 2020. ISSN 1367-4803. doi: 10.1093/bioinformatics/btaa792.

[27] Frédéric Boyer, Anne Morgat, Laurent Labarre, Joël Pothier, and Alain Viari. Syntons, metabolons and interactons: an exact graph-theoretical approach for exploring neighbourhood between genomic and functional data. *Bioinformatics*, 21(23):4209–4215, 2005.

[28] David Vallenet, Alexandra Calteau, Mathieu Dubois, Paul Amours, Adelme Bazin, Mylène Beuvin, Laura Burlot, Xavier Bussell, Stéphanie Fouteau, Guillaume Gautreau, Aurélie Lajus, Jordan Langlois, Rémi Planel, David Roche, Johan Rollin, Zoe Rouy, Valentin Sabatet, and Claudine Médigue. MicroScope: an integrated platform for the annotation and exploration of microbial gene functions through genomic, pangenomic and metabolic comparative analysis. *Nucleic Acids Research*, 48(D1):D579–D589, 10 2019. ISSN 0305-1048. doi: 10.1093/nar/gkz926.

[29] Wenjun Li, Kathleen R O'Neill, Daniel H Haft, Michael DiCuccio, Vyacheslav Chetvernin, Azat Badretdin, George Coulouris, Farideh Chitsaz, Myra K Derbyshire, A Scott Durkin, Noreen R Gonzales, Marc Gwadz, Christopher J Lanczycki, James S Song, Narmada Thanki, Jiyao Wang, Roxanne A Yamashita, Mingzhang Yang, Chanjuan Zheng, Aron Marchler-Bauer, and Françoise Thibaud-Nissen. RefSeq: expanding the Prokaryotic Genome Annotation Pipeline reach with protein family model curation. *Nucleic Acids Research*, 49(D1):D1020–D1028, 12 2020. ISSN 0305-1048. doi: 10.1093/nar/gkaa1105.

[30] Brian D Ondov, Todd J Treangen, Páll Melsted, Adam B Mallonee, Nicholas H Bergman, Sergey Koren, and Adam M Phillippy. Mash: fast genome and metagenome distance estimation using minhash. *Genome biology*, 17(1):1–14, 2016.

[31] Edoardo Pasolli, Francesco Asnicar, Serena Manara, Moreno Zolfo, Nicolai Karcher, Federica Armanini, Francesco Beghini, Paolo Manghi, Adrian Tett, Paolo Ghensi, Maria Carmen Collado, Benjamin L. Rice, Casey DuLong, Xochitl C. Morgan, Christopher D. Golden, Christopher Quince, Curtis Huttenhower, and Nicola Segata. Extensive unexplored human microbiome diversity revealed by over 150,000 genomes from metagenomes spanning age, geography, and lifestyle. *Cell*, 176(3):649–662.e20, 2019. ISSN 0092-8674. doi: https://doi.org/10.1016/j.cell.2019.01.001.

[32] Oliver Schwengers, Lukas Jelonek, Marius Dieckmann, Sebastian Beyvers, Jochen Blom, and Alexander Goesmann. Bakta: Rapid & standardized annotation of bacterial genomes via alignment-free sequence identification. *bioRxiv*, 2021.

[33] Donovan H Parks, Maria Chuvochina, Christian Rinke, Aaron J Mussig, Pierre-Alain Chaumeil, and Philip Hugenholtz. Gtdb: an ongoing census of

bacterial and archaeal diversity through a phylogenetically consistent, rank normalized and complete genome-based taxonomy. *Nucleic Acids Research*, 2021.

[34] Jason R Grant and Paul Stothard. The cgview server: a comparative genomics tool for circular genomes. *Nucleic acids research*, 36(suppl_2): W181–W184, 2008.

[35] Elzbieta Brzuszkiewicz, Holger Brüggemann, Heiko Liesegang, Melanie Emmerth, Tobias Ölschläger, Gábor Nagy, Kaj Albermann, Christian Wagner, Carmen Buchrieser, Levente Emődy, Gerhard Gottschalk, Jörg Hacker, and Ulrich Dobrindt. How to become a uropathogen: Comparative genomic analysis of extraintestinal pathogenic escherichia coli strains. *Proceedings of the National Academy of Sciences*, 103(34):12879–12884, 2006. ISSN 0027-8424. doi: 10.1073/pnas.0603038103.

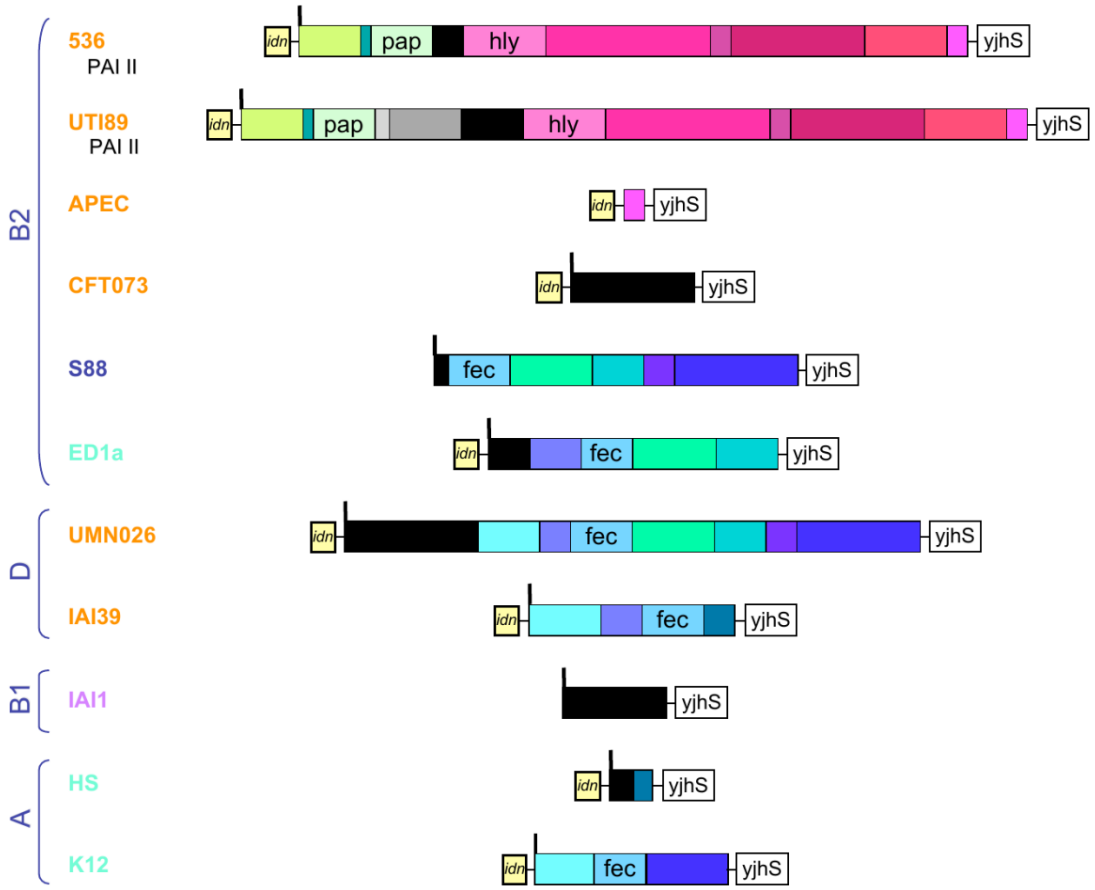


FIGURE 8.1 – Schéma du spot *leuX* dans différentes souches de *Escherichia coli*
Cette figure illustre l'organisation modulaire du spot *leuX* chez E. coli. Les rectangles de la figure représentent des modules de gènes qui correspondent à des groupes de gènes qui sont supposés être acquis ensemble dans différentes souches. Figure réalisée par Alexandra Calteau.

8.3 Conclusion

La méthode panModule permet d'identifier des modules conservés à l'échelle d'une espèce, et ceux-ci ont une correspondance importante avec des modules fonctionnels qui ont été expertisés grâce à leur annotation fonctionnelle. C'est le premier outil de son genre, dans le sens où il est le seul à utiliser l'information de colocalisation apportée par le graphe de pangénome, et à rester exclusivement dans le contexte du pangénome. Ignorer ainsi l'information provenant d'autres espèces, contrairement aux autres outils, présente l'avantage de n'étudier les gènes que dans

'un seul' contexte, en quelque sorte, qui est le contexte de l'espèce que le pangénome représente. Même si le fait qu'un bloc de gènes soit conservé dans plusieurs contextes génomiques serait une information beaucoup plus forte pour leur association fonctionnelle, un même système moléculaire retrouvé dans des contextes différents peut être très variable. Ainsi, une variabilité entre des modules d'espèces différentes ne serait pas forcément signe que les modules sont mal construits, mais peut être au contraire une indication de la variabilité du système. On retrouve la prise en compte de cette notion variabilité au sein de systèmes biologiques, notamment, dans les sous-systèmes dans SEED (Overbeek et al., 2005) mais aussi les gènes non obligatoires dans les systèmes de MacSyFinder (Abby et al., 2014).

Comparer ainsi les modules et, plus largement, les pangénomes de plusieurs espèces entre elles permettrait ainsi d'étudier les dynamiques d'échanges inter-espèces ainsi que les variabilités fonctionnelles à différents niveaux taxonomiques. Des travaux seront conduits avec cet objectif dans le cadre de la thèse de Jérôme Arnoux au LABGeM (2021-2024).

Un prélude de ces comparaisons à large échelle est la constitution d'une base de données de pangénomes qui utilise des milliers, voir des millions de génomes, pour permettre ensuite des analyses conjointes. C'est là un des sujets du prochain chapitre de cette thèse qui sera dédié à la constitution de cette base de données appelée panGBank.

Chapitre 9

panGBank

9.1 Préambule et historique

Pour générer les pangénomes des différents articles présentés jusqu'alors, il a fallu concevoir un ensemble de méthodes et de programmes automatisés pour télécharger et assurer la persistance des données, regrouper les génomes en espèces et calculer les pangénomes, pour ensuite en extraire les composantes intéressantes pour nos différentes études. D'autres chercheurs ont souhaité s'intéresser aux génomes générés dans le contexte de certains des articles présentés dans cette thèse. Néanmoins, la mise à disposition et le partage de ce type de données est loin d'être aisé : chaque pangénome est une petite base de données en soit, avec des milliers, voire des millions de relations, de séquences de gènes et de résultats de calculs des différentes méthodes présentées précédemment. Ainsi est né le besoin de panGBank : une base de données publique de pangénomes générés par PPanGGOLiN dont le contenu en génomes peut graduellement inclure ceux nouvellement mis à disposition dans des bases de données publiques comme GenBank.

Le projet a commencé avant mon arrivée au laboratoire, il a été initié en juillet 2018, et se conclura probablement après mon départ (prévu pour février 2022). Il a eu un nombre conséquent (à mes yeux) de protagonistes, qui ont conçu, modifié, amélioré, réécrit les différentes parties de panGBank. Le projet a ainsi inclus les ingénieurs Mathieu Gachet, Laura Burlot et Rémi Planel, l'ingénieur de recherche Mathieu Dubois, les chercheurs Alexandra Calteau et David Vallenet, les alternants Paul Amours et Jérôme Arnoux, et les doctorants Guillaume Gautreau, Jérome Arnoux (qui débute sa thèse) et moi-même. Une part de cette section peut se retrouver, racontée différemment, dans le rapport d'alternance de Paul Amours, que j'ai co-encadré.

Je n'ai donc pas travaillé seul, loin de là. Les différents éléments que je vais présenter dans cette section rendent compte de l'état actuel de panGBank, néan-

moins celui-ci a connu de (très) nombreuses modifications avant d'en arriver là. Je vais rendre compte de l'ensemble du projet. Les protagonistes ayant beaucoup changé, le NCBI (National Center for Biotechnology Information, une agence américaine, dépositaire de beaucoup de données publique) étant très friand de changements intempestifs non documentés sur leur politique de partage de données, et les différents outils utilisés ayant la désagréable habitude de ne soudainement plus fonctionner correctement lorsqu'il y a trop de génomes/chemins/fichiers impliqués, j'ai eu le plaisir de toucher à toutes les parties du projet, qu'elles relèvent de la technique ou de la recherche.

9.2 Introduction

L'essor de la pangénomique a mené à la mise à disposition de ressources de pangénomes. Par conséquent, il existe déjà plusieurs bases de données de pangénomes accessibles via une interface web. Elles ont toutes plusieurs avantages ou limitations, que j'ai tenté de résumer dans le tableau 9.1.

Globalement, les limitations de toutes ces plateformes sont qu'aucune ne peut faire des pangénomes qui ont un nombre de génomes supérieur à quelques centaines, et que la plupart utilisent un paradigme de core/accessoire qui est très limitant dès que l'on a beaucoup de génomes. Cela n'est pas une limitation dans leur cas étant donné qu'ils ont justement peu de génomes dans leurs pangénomes, mais cela le serait dans le cas d'une base de données qui inclurait la majorité de la diversité des génomes disponibles publiquement.

Toutes les bases de données mentionnées ont globalement les mêmes défauts :

- Aucune ne permet d'explorer toute la diversité d'une base de données publique de génomes comme GenBank, ou d'autres de taille équivalente.
- Aucune ne propose d'API (Application Programming Interface) qui permettrait d'automatiser certaines analyses ou la récupération de certains fichiers pour des analyses à l'échelle de plus d'un pangénome.
- Aucune ne réalise un partitionnement statistique.

L'objectif de panGBank est de répondre à ces trois faiblesses, en proposant une base de données de pangénomes qui serait mise à jour régulièrement avec l'évolution du contenu des banques de génomes. Cette ressource disposerait d'une API avec laquelle les développeurs pourraient interagir programmatiquement, mais également qui serait accessible via une interface web pour les bioanalystes non-développeurs. Finalement, les pangénomes calculés seraient partitionnés statistiquement par PPanGGOLiN, et leurs îlots génomiques, spots d'insertions et modules automatiquement détectés par les différentes méthodes disponibles dans la suite PPanGGOLiN.

Outil	limites	avantages	citations
EDGAR (Blom et al., 2009)	• Peu ergonomique • Paradigme core / accessoire • Génomes fixes (24317 génomes)	• Nombreuses fonctionnalités • Choix des génomes	399
MicroScope (Vallenet et al., 2017)	• Paradigme core / accessoire • Limité à 300 génomes • Génomes fixes (5893 génomes publics)	• Choix des génomes • Lien avec le reste de la plateforme • Courbes de raréfaction	166
panWeb (Pantoja et al., 2017)	• Beaucoup de bugs • formats standards (gbff) non supportés • Pas de base de données de génomes	• Choix des génomes • Choix des paramètres • Courbes de raréfaction	16
panX (Ding et al., 2018)	• Génomes fixes et peu de diversité • Paradigme core / accessoire • Limité à 500 génomes	• Ergonomique • Phylogénies espèce et gènes • Métadata sur les génomes • Open source	146
PGAweb (X. Chen et al., 2018)	• Paradigme core / accessoire • Très limité en nombre de génomes (une dizaine/centaine)	• Courbe de raréfaction • Phylogénie espèce • Repose sur PGAP, qui est open source	15
panGFR-HM (Chaudhari et al., 2018)	• Inaccessible		4

Table 9.1 – Plateformes web pour les pangénomes bactériens

9.3 Workflow de panGBank

9.3.1 Téléchargement des génomes

Dans l'état actuel des choses, l'étape de téléchargement est gérée par BioMaJ (Filangi et al., 2008), un framework python pour synchroniser et traiter de multiples banques de données. À chaque itération, Biomaj télécharge l'ensemble des génomes de GenBank qu'il n'a pas déjà automatiquement, au format fasta, puis lance un traitement sur tous ces génomes qui permet de produire des métriques de qualité des assemblages. Les métriques de qualité sont les suivantes :

- Nombre de contigs : nombre de séquences contiguës dans l'assemblage d'un génome
- N50 : taille du plus petit contig nécessaire pour couvrir 50% du génome avec l'ensemble des contigs de taille plus grande ou égale.
- L90 : nombre de contigs nécessaires pour couvrir 90% du génome

9.3.2 Contrôle qualité

Actuellement, les génomes ayant un $L90 \leq 200$ et un nombre de contigs ≤ 1000 sont conservés. De plus, les génomes sont tous associés à une ontologie de statut au niveau de GenBank, et ceux ayant les statuts "multi-isolate project" ou "partial" ne sont pas considérés. Pour les statuts "partial", il s'agit de portions de génomes souvent très petites et non de génomes complets, ce qui pourrait nuire aux résultats de l'analyse de pangénome. Pour les "multi-isolate project", il s'agit majoritairement de projets de surveillance sanitaire (comme ceux conduits par la FDA, Food and Drug Administration, aux États-Unis). Ces projets consistent à séquencer absolument tout ce qui peut être lié à un problème sanitaire. Ils incluent donc beaucoup de souches de mêmes espèces, souvent déjà très étudiées comme *Salmonella enterica* ou *Escherichia coli*, et où il n'y a que peu de diversité génétique entre les souches. Ignorer ces souches nous permet d'éviter d'ajouter des centaines de milliers de génomes extrêmement proches dans les pangénomes qui sont déjà composés de milliers de génomes. De plus, la diversité qu'ils représentent est beaucoup plus limitée que lorsque les souches viennent de projets différents, et les inclure biaiserait la composition des pangénomes. PPanGGOLiN en théorie pourrait calculer des pangénomes avec des centaines de milliers de génomes, mais les ressources informatiques dont nous disposons ne pourraient assumer les millions de génomes qui seront probablement accessibles d'ici à un ou deux ans.

9.3.3 Assignation taxonomique

Nous avons cherché à assigner tous les génomes précédemment sélectionnés à des espèces identifiables, lorsque c'était possible. Pour cela, nous avons utilisé la taxonomie de GTDB (Parks et al., 2018) comme référence. Dans GTDB, chaque espèce dispose d'un génome de référence qui la représente. Celui-ci est choisi compte tenu de l'historique de l'espèce s'il y en a un, à la qualité de son assemblage et à sa centralité au regard des autres génomes de l'espèce. Une manière rapide d'assigner taxonomiquement un génome à une espèce est donc de comparer nos génomes aux génomes de références de GTDB.

Parmi les outils que j'ai précédemment mentionnés dans la section 3.4.3, nous avons retenu Mash comme outil de comparaison, car celui-ci même s'il n'est pas le plus rapide ou le plus précis permet de réaliser exactement les opérations que l'on souhaitait. Nous avons ensuite essayé d'établir le meilleur seuil à utiliser pour déterminer quelle était la distance Mash maximale en dessous de laquelle un génome serait trop différent pour appartenir à l'espèce représentée par son génome de référence. Pour cela, nous avons utilisé la base de GTDB elle-même. Nous avons téléchargé tous les génomes de GTDB, et comparé tous les génomes de GTDB avec tous les génomes de référence de GTDB en utilisant Mash. Ensuite, nous avons mesuré les similitudes et les différences entre la taxonomie des génomes dans GTDB, et l'assignation que l'on réalise avec Mash.

La figure 9.1 illustre le niveau de cohérence taxonomique compte tenu de la distance Mash dans cette analyse. Nous avons conclu que nous utiliserions une distance Mash de 0.05 pour l'assignation. Cette mesure a été retenue pour deux raisons : premièrement il n'y avait que quelques centaines de mauvaises assignations sur presque 200 000 génomes et toutes les mauvaises assignations se faisaient sur une espèce du même genre. Deuxièmement, ce seuil de distance Mash correspond approximativement à la valeur d'ANI qui avait été identifiée dans l'article de Konstantinidis et al., 2005 et qui a été largement utilisée.

Dans le workflow de panGBank, nous comparons nos génomes avec les génomes de références de GTDB, avec un seuil de distance Mash maximal de 0.05, pour assigner ces génomes à des espèces. Après cette opération, tous les génomes ne sont pas forcément associés à une espèce. On va alors essayer de regrouper ensemble des génomes non assignés, pour voir si certains se ressemblent suffisamment. Pour cela, nous allons comparer tous ces génomes entre eux avec Mash, en utilisant le même seuil. Ensuite, on construit un graphe $G(V, E)$ où les nœuds V sont les génomes et les arêtes E les relations de similarité identifiées par Mash. On applique alors l'algorithme de Louvain (Blondel et al., 2008) sur ce graphe pour identifier des communautés que l'on va présumer être des espèces.

Ensuite, pour chaque espèce qui dispose de 15 génomes ou plus (valeur modulable compte tenu de nos besoins), le workflow va calculer un pangénome.

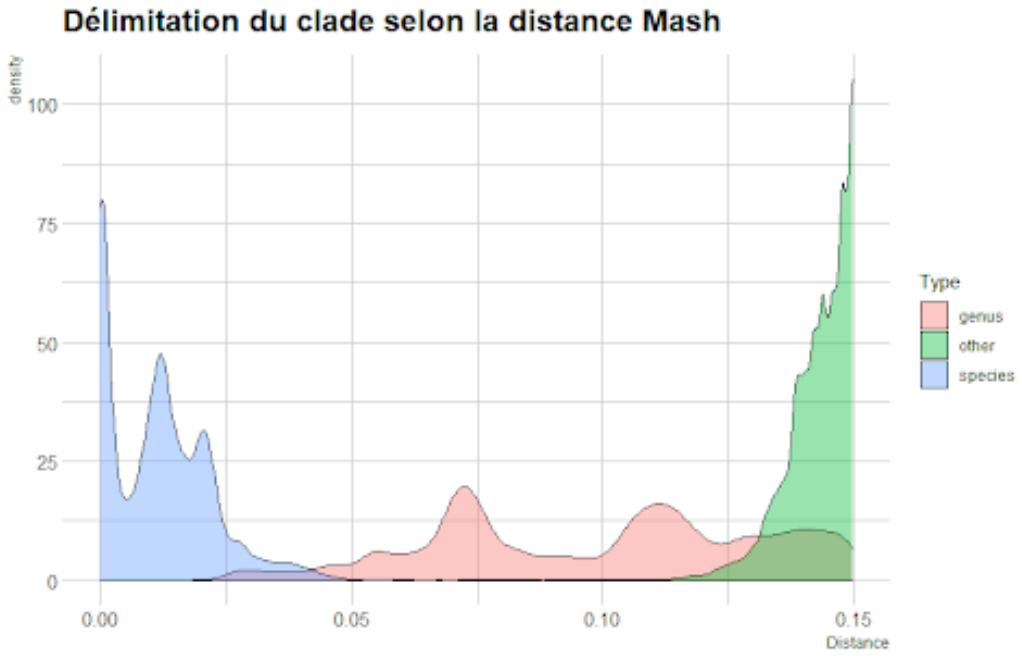


FIGURE 9.1 – Comparaison des génomes de GTDB avec Mash
Graphe de densité illustrant le résultat de la comparaison des génomes de GTDB avec Mash. Pour chaque comparaison, on renvoie le niveau taxonomique commun le plus bas entre le génome de référence et l'assignation taxonomique du génome comparé. Image générée par Paul Amours.

9.3.4 Contenu et résultats de panGBank

Le contenu actuel de panGBank est ainsi dépendant de deux bases de données : la source des génomes et la source de la taxonomie. La version actuelle, dont je vais commenter les résultats, est issue de la version de GenBank du 17 août 2021, et de la version de GTDB 06-RS202 du 27 avril 2021.

Le tableau 9.2 indique le nombre de génomes restant après chacune des étapes du workflow. La majorité des génomes filtrés sont ceux qui sont marqués comme "multi-isolate project". Les différences entre les étapes d'assignation taxonomique et de construction des pangénomes sont liées au fait que l'on ne reconstruit des pangénomes que pour les espèces ayant plus de 15 génomes. Ainsi, pour l'écrasante majorité des espèces non présentes dans GTDB, on ne dispose pas d'assez de génomes pour reconstruire un pangénome, puisque parmi les 23 787 clusters de génomes réalisés avec Louvain, uniquement 20 d'entre eux ont plus de 15 génomes.

Téléchargement	Contrôle qualité	Assignation taxonomique	Pangénomes
998 284	313 284	GTDB : 283 562 Louvain : 29 722	GTDB : 196 630 Louvain : 316

TABLE 9.2 – Nombre de génomes restant après chaque étape de panGBank

C'est vrai aussi pour la plupart des espèces de GTDB, car parmi les 47 893 espèces, on ne reconstruit que 1346 pangénomes. Nous avons donc un total de 1366 pangénomes qui regroupent 196 946 génomes.

Le pangénome avec le plus de *persistent* est *Streptomyces mirabilis* avec un total de 7 781 familles, suivi de près par plusieurs autres espèces de Streptomyces et de Bradyrhizobium. Ce sont des bactéries du sol et associées aux plantes. Le génome de la souche de référence de *S. mirabilis* fait 11.29Mb et a un taux de GC de 70 %, qui sont des valeurs très hautes pour une bactérie. La bactérie elle-même est remarquable par plusieurs aspects : elle est productrice d'antibiotique (EL-SAYED, 2012) et est hautement résistante aux métaux lourds, elle produit des métabolites secondaires associés notamment au nickel (A. SCHMIDT et al., 2009).

Le pangénome avec le moins de *persistent* est celui de *Aminicenans sakinawicola*, qui appartient au phylum des Acidobacteriota (Phylum GTDB, Candidatus Aminicenans au NCBI) avec le nombre incroyable de 4 familles. C'est, heureusement, la seule avec un nombre aussi extrême, car au vu de cette valeur irréaliste il y a probablement eu un problème lié à la diversité de l'espèce elle-même, au clustering des familles, ou au partitionnement. Le génome représentatif de l'espèce dans GTDB n'est pas un "vrai" génome, mais un concaténât de 24 expériences de single cell, dont les génomes ont au maximum 97 % d'ANI. Il a été publié dans le contexte d'un article dédié au séquençage et à l'analyse de génomes issus de phylum majoritairement inexplorés (RINKE et al., 2013).

De manière plus générale, la diversité taxonomique de panGBank représentée dans la figure 9.2 est fortement biaisée par certains groupes : presque la moitié des pangénomes sont issus de protéobactéries, dont 80 % de gamma-protéobacteries. Sans grande surprise, la majorité des groupes que l'on retrouve sont ceux qui sont notoirement connus pour, soit leurs pathogènes (Enterobacteriaceae, Vibrionaceae, Staphyloccaceae), soit leur intérêt pour l'agroalimentaire (Lactobacillaceae, Rhizobiaceae), soit leur capacité à produire des molécules comme des antibiotiques (les Actinobacteriota en général). De plus, le règne des Archées est extrêmement mal représenté avec seulement 20 pangénomes, qui comprennent 487 génomes. La diversité génomique des Archées reste encore très peu connue comparée à celle des bactéries.

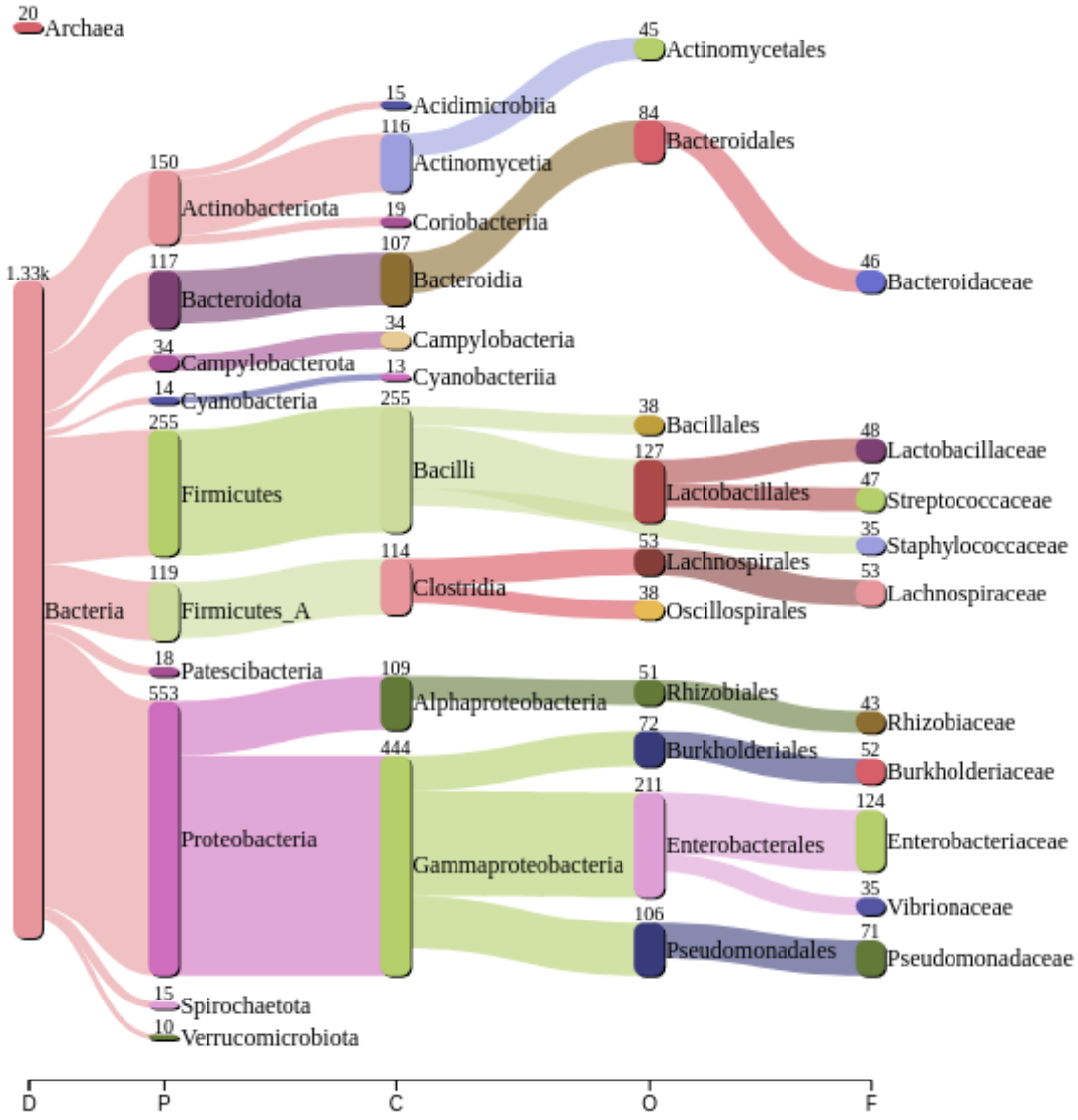


FIGURE 9.2 – Nombre de pangénomes par taxons dans panGBank
Figure illustrant la diversité taxonomique des pangénomes dans panGBank. Le principe de ce type de représentation est de ne montrer que les n (ici 10) taxons les plus représentés par niveau taxonomique (5 niveaux ici, du règne à la famille). Figure générée avec Pavian (BREITWIESER et al., 2020).

9.4 Futur de panGBank

En l'état, panGBank peut être interrogé par nos soins, quand certains pangénomes nous intéressent ou pour certains articles. L'article de PPanGGOLiN inclut

notamment les résultats d'une ancienne version de panGBank. Les développements nécessaires pour finaliser le travail sont en grande partie informatiques, il s'agit de créer une interface web et une API pour interroger, filtrer et télécharger les pangénomes (ou des parties des pangénomes).

Quatrième partie

Conclusions

Chapitre 10

Conclusions et perspectives

10.1 Conclusions sur ce travail de thèse

Dans cette thèse, les objectifs étaient à l'origine de développer des méthodes autour de PPanGGOLiN pour détecter et caractériser les régions variables des pangénomes, puis d'analyser et comparer celles-ci au regard de leurs capacités métaboliques.

Le travail a été de finaliser PPanGGOLiN lui-même conjointement avec les autres auteurs de l'article, ce qui a pris presque un an de cette thèse et dont les résultats ont été présentés dans le chapitre 6. L'approche de partitionnement statistique des graphes de pangénome permet de calculer le génome *persistent*, le génome *shell* et le génome *cloud*. Le *persistent* s'est avéré beaucoup plus stable face au biais d'échantillonnage et à la fragmentation des données génomiques que n'importe quelle autre méthode alors disponible.

Dans un second temps, le chapitre 7 présente une méthode originale pour la détection d'îlots génomiques qui peut utiliser des informations issues de milliers de génomes grâce au partitionnement. C'est la première méthode de génomique comparative capable ainsi de passer à l'échelle sur des milliers de génomes, et elle s'est positionnée très honorablement en comparaison aux autres approches existantes. panRGP introduit aussi la première approche de génomique comparative permettant une analyse globale des sites d'insertion d'une espèce qui est accessible au travers d'un logiciel et ne repose en aucune manière sur le choix d'un génome pivot, ou de référence. Cette méthode a été publiée à l'occasion de la conférence "the 19th European Conference on Computational Biology" (ECCB2020) dans Bioinformatics. Cela m'a permis de présenter mon travail pendant 10 minutes durant la conférence. J'ai aussi eu l'occasion de présenter panRGP pendant 15 minutes dans le contexte des Journées Ouvertes en Biologie, Informatique et Mathématiques (JOBIM2021), lors d'une session 'highlights' qui permettait de présenter un

article paru lors de l'année 2020.

Finalement, le chapitre 8 introduit une approche qui permet de découper le génome accessoire d'un pangénome en modules conservés, et qui se compare favorablement à un jeu de données expertisé chez *E. coli*. C'est une approche originale en soi, car il n'existe aucune méthode équivalente qui permet de découper des blocs de gènes uniques non chevauchants à l'échelle d'un pangénome fait de centaines, voire de milliers de génomes. La méthode panModule permet ainsi d'obtenir une forme d'annotation relationnelle pour des gènes dont on ne dispose pas forcément d'autres informations, ce qui peut faciliter les travaux d'analyses. Ces informations peuvent être très pratiques lorsqu'on travaille sur des organismes peu étudiés, et phylogénétiquement éloignés de ceux qui le sont un peu plus. Cette méthode est actuellement disponible en preprint sur biorxiv.

Les analyses et comparaisons de pangénomes ont été très sporadiques et seront approfondies dans le cadre du travail d'un autre étudiant en thèse, Jérôme Arnoux, qui prendra aussi la suite du développement de PPanGGOLiN.

J'ai eu l'occasion de présenter les travaux menés pendant cette thèse de manière plus globale à deux reprises. En premier lieu lors de la conférence JOBIM2019 à Nantes, où j'ai présenté un poster, et lors de la conférence "8th Congress of European Microbiologists" (FEMS19) à Glasgow ou j'ai pu présenter le même poster que JOBIM2019 ainsi que réaliser une présentation de 5 minutes.

J'ai eu l'occasion de mettre en application panRGP et PPanGGOLiN lors d'une analyse d'un pangénome de *Nesseria meningitidis*, dans le contexte de NeMeSys 2.0, un projet de construction d'une banque de mutants non essentiels, et qui a mené à un article dont je suis co-auteur (Muir et al., 2020).

Une chance que j'ai eue lors de cette thèse est de voir d'autres utiliser le fruit de mon travail et construire leur recherche autour de ces outils pour faire parfois des choses que je n'avais pas imaginées ou considérées. Je pense notamment au travail de Pfeifer et al., 2021 sur des familles de phages dont les graphes de pangénomes ont été construits avec PPanGGOLiN. Je salue par la même occasion le travail de Flores Ramos et al., 2021, dont les auteurs ont été parmi les premiers à utiliser les parties de PPanGGOLiN qui manipulent les spots et les RGP, et qui ont donc eu à subir certains désagréments liés à des imperfections, maintenant en partie dépassées, de l'outil. Alors que j'écris ces lignes, PPanGGOLiN a été cité 30 fois, et panRGP 9 fois (nombres obtenus sur Google Scholar).

PPanGGOLiN n'est pas seulement utilisé dans le contexte de travaux donnant des publications, on m'a aussi rapporté des usages en routine dans une équipe de surveillance sanitaire, dans des compagnies de biotechnologie, ou encore dans un groupe réalisant des prestations en bioinformatique. Je suis honoré d'avoir pu contribuer ainsi à la création d'un outil et de méthodes qui ont pu être utiles à certains.

10.2 Perspectives sur les méthodes développées

10.2.1 PPanGGOLiN et réserves sur la méthode de partitionnement

Partitionner un pangénome avec une approche statistique est, je pense, une nécessité. Les données que l'on manipule ont souvent des origines très variées et leur qualité l'est aussi. Malgré les quelques outils existants pour mesurer la qualité, ceux-ci restent souvent imprécis, car ils ne se basent que sur la présence de quelques dizaines de familles attendues chez une bactérie (ou plus généralement, un procaryote). De plus, malgré les tentatives d'uniformiser la phylogénie, les groupes étudiés peuvent aussi avoir une diversité variable d'un groupe à l'autre. Utiliser ainsi une méthode de partitionnement statistique permet d'offrir une garantie, celle d'obtenir un partitionnement qui correspond à ses données d'entrée et à leur variabilité intrinsèque. PPanGGOLiN n'est peut-être pas l'approche idéale, mais elle a le mérite de répondre pleinement à cette problématique, avec l'élégance de prendre en compte le voisinage des familles en plus de leur profil de présence.

L'usage de modèles mixés de Bernoulli semble très adapté à la détection du *persistent* ou des sous-parties du *shell* qui sont effectivement présentes dans un nombre conséquent de génomes. C'est d'ailleurs sur ce point que la méthode a été justifiée dans l'article. Cela n'est néanmoins peut-être pas le cas pour les gènes plus rares où le modèle de Bernoulli qui leur correspond le plus est simplement un vecteur composé exclusivement d'absence. À cause de cela, la frontière entre le *shell* peu présent et le *cloud* est extrêmement floue, et je ne pense pas qu'elle ait une quelconque réalité biologique.

Idéalement, le *cloud* devrait correspondre aux gènes qui sont très récemment acquis et qui n'ont donc pas eu le temps d'être sujet à la sélection purificatrice que la majorité des génomes procaryotes subissent. Le *shell*, lui, devrait correspondre aux gènes qui ont effectivement eu le temps de subir une pression évolutive qui les auraient contre-sélectionné s'ils étaient inutiles. Nous n'avons pas formellement testé cela entre les gènes du *cloud* et ceux du *shell*. Cependant, je ne m'attends pas à ce qu'il y ait une telle dichotomie avec un partitionnement qui se base exclusivement sur la présence et l'organisation. Une méthode de partitionnement de pangénome qui viserait à séparer distinctement le *shell* et le *cloud* devrait donc, à mon avis, se fonder sur la sélection que lesdits gènes ont (ou n'ont pas) déjà subi depuis leur intégration dans les génomes. Cela serait peut-être plus adapté que le type d'approche que nous utilisons actuellement dans PPanGGOLiN, ou celles qui utilisent la fréquence comme mOTUpan(Buck et al., 2021) ou micropan(Snipen et al., 2015).

10.2.2 panRGP, et l'usage d'un score arbitraire

panRGP est une approche de génomique comparative qui utilise la partition d'un pangénome comme proxy d'une comparaison et, grâce à cela, peut s'appliquer sur des dizaines de milliers de génomes. Néanmoins, la détection en elle-même des RGP utilise un ensemble de scores et de fonctions relativement arbitraires dans leur construction. Notre volonté était de retrouver des ensembles de gènes variables (*shell* et *cloud*) d'une taille suffisante pour être un peu plus qu'une unité monocistronique et interrompu par peu de gènes *persistent*. On s'attend à ce que ces ensembles soient minoritaires à l'échelle de chaque génome, c'est-à-dire que les génomes sont tous très majoritairement composés de *persistent*. L'article de panRGP le montre. Ces scores fonctionnent excellemment sur des génomes connus dont les espèces ont été assez bien définies et dont on possède un nombre conséquent de génomes.

Les résultats de panRGP, notamment dans le calcul du score, sont néanmoins très dépendants des résultats de partitionnement du pangénome. Si l'on sort du contexte de l'espèce, et de l'attente que nous avions d'avoir une majorité de gènes *persistent*, il est possible que ces scores donnent des régions extrêmement larges, voire l'ensemble du génome s'il y a vraiment peu de *persistent*. Les régions ainsi détectées ne seraient donc probablement pas des îlots génomiques ou d'anciens îlots et ne correspondraient pas à grand-chose de biologiquement ou d'évolutivement sensé.

La méthode panRGP est donc probablement convenable à l'échelle d'une espèce telle que GTDB les définies, mais peut-être pas à des échelles taxonomiques plus larges. Pour cela, je pense que des approches, comme celle de xenoGI (Bush et al., 2018), sont excellentes et peut-être plus adaptées à la problématique, car chaque bloc de gènes est placé dans un arbre phylogénétique. On peut donc plus aisément identifier les portions nouvellement acquises des autres simplement en se référant à la position dans l'arbre où les gènes ont intégré les génomes. Malheureusement xenoGI lui-même va difficilement au-delà de 50 génomes pouvant être analysés simultanément, ce qui est très limitant aujourd'hui. Une méthode idéale devrait réaliser un placement phylogénétique de blocs de gènes et être capable de fonctionner sur des milliers de génomes de n'importe quelle diversité, mais une telle approche n'existe pas encore à ma connaissance.

10.2.3 panModule, des modules conservés mais à quelle échelle ?

La méthode panModule permet d'identifier des blocs de gènes qui sont conservés ou perdus ensemble dans un groupe de génomes d'intérêt. La mesure choisie est basée sur un coefficient de similarité de Jaccard entre les familles de gènes qui composent un module. Néanmoins, un problème récurrent dans les jeux de

données de génomes, et notamment ceux issus des banques de données publiques, est l'échantillonnage. On retrouve beaucoup plus de génomes d'intérêt clinique ou industriel, dont certains peuvent être très proches, voire clonaux, bien loin de la diversité que l'on retrouverait avec un échantillonnage uniforme des individus d'une espèce.

Dans le contexte de panModule, chaque présence ou absence d'une famille en relation avec une autre a le même poids dans le calcul de la similarité de Jaccard. On pourrait imaginer au contraire prendre en compte la diversité que représente chaque génome dans le groupe étudié : si plusieurs génomes sont très proches (clonaux), ceux-ci pourraient avoir un poids minoré face à des génomes qui seraient plus différents des autres.

Hors du contexte de panModule, plusieurs méthodes cherchent à prendre en compte la phylogénie ou la diversité que peuvent représenter les génomes, de différentes manières. Le logiciel dRep (Olm et al., 2017) notamment inclut une approche permettant de choisir des génomes dont l'ensemble formerait un échantillon plus homogène. Coinfinder (Whelan et al., 2020), que j'ai déjà mentionné précédemment, peut rapporter pour chacune des associations ou dissociations détectées la mesure de dispersion D des traits au regard d'un arbre phylogénétique (Fritz et al., 2010).

10.3 Perspectives sur la génomique comparative

Toutes les méthodes auxquelles j'ai contribué sont des approches de génomiques comparatives qui sont dites "reference-free". Elles ne se comparent pas à un (ou des) génomes de référence, ce qui permet d'outrepasser beaucoup de biais. L'immense avantage de ces méthodes est de pouvoir s'appliquer sur n'importe quels organismes, même si leurs génomes n'ont jamais été étudiés.

Contrairement à ce que l'on pourrait penser naïvement, l'essor de ces méthodes est resté poussif en comparaison de la masse de données générées ces 15 dernières années. Encore aujourd'hui en bioinformatique, énormément d'approches se basent sur des comparaisons à une ou des références. De nombreuses analyses sont conduites sur plusieurs génomes simultanément, mais totalement indépendamment, là où l'on pourrait grandement gagner à analyser ces génomes conjointement. Il est néanmoins plus simple d'utiliser des méthodes que l'on maîtrise et qui donnent des résultats dont on connaît (ou dont on croit connaître) les biais éventuels, que de chercher constamment l'amélioration, ce qui est une prise de risque, pour tout ce que l'on fait. Cette inertie est inhérente à notre fonctionnement.

Je peux citer comme exemple les recherches de variants qui sont globalement toujours basées sur des références et conduites séparément sur plusieurs génomes avant d'être réunies. Des méthodes pour dépasser ce modèle commencent tout

juste à émerger (je pense notamment à Pandora (Colquhoun et al., 2021), qui est une approche utilisant des graphes de pangénomes de séquences).

Un autre exemple qui me tient beaucoup à cœur est la détection de gènes dans les génomes procaryotes, que l'on appelle annotation syntaxique. Certains auteurs utilisent aussi les termes "annotation structurale" (structural annotation, en anglais), mais je trouve cela terriblement maladroit sachant que la bioinformatique structurale, qui inclut des étapes d'"annotation des structures" tridimensionnelles des protéines existe. Malgré le fait que de très nombreuses analyses requièrent cette détection de gènes, il n'existe en comparaison du besoin qu'extrêmement peu d'outils qui la réalise. La liste exhaustive (dans la limite de mes connaissances) des outils d'annotation syntaxique aujourd'hui utilisables est la suivante : Glimmer3 (Delcher et al., 2007), GenemarkS2+ (Lomsadze et al., 2018), Balrog (Sommer et al., 2021), Prodigal (Hyatt et al., 2010), Amigene (Bocs et al., 2003), Zcurve (Guo et al., 2003), MetaGeneAnnotator (Noguchi et al., 2008). Tous ces outils sont dédiés à l'analyse d'un unique ensemble de séquences d'un génome ou d'un métagénome, hors de tout contexte, et jamais en comparaison avec d'autres génomes qui lui ressemblerait. Quelques pipelines d'annotations (PGAP (Tatusova et al., 2016) ou bakta (Schwengers et al., 2021), par exemple) rajoutent une étape de réannotation, en comparant des séquences de protéines connues à certaines zones du génome, et c'est ce qui se rapproche le plus d'une annotation syntaxique fondée sur la génomique comparative.

Aujourd'hui, l'immense majorité des analyses de génomique procaryote se font avec plusieurs génomes souvent phylogénétiquement proches. Je pense qu'on gagnerait grandement à analyser ces génomes ensemble en prenant en compte le fait que la majorité des gènes sont sous sélection positive. Par conséquent, on s'attend à ce que les gènes notamment soient des portions plus conservées que tout ce qui n'est pas fonctionnel entre des génomes très proches. Un unique outil d'annotation syntaxique dédié spécifiquement à la correction de la prédiction des codons d'initiation des gènes nommé StartLink+ (sur biorxiv) (Gemayel et al., 2020) utilise en quelque sorte cela, mais le faire pour la détection de gènes me semble potentiellement aussi approprié. Certaines ébauches en cours de développement comme ggCaller (non publiée, https://github.com/samhorsfield96/ggCaller) sont peut-être le début de cela.

Il y a donc encore beaucoup de possibilités pour utiliser la génomique comparative pour apprendre des données que l'on collecte massivement et que l'on agglomère dans des bases de données. Proposer des alternatives méthodologiques, même si elles sont moins performantes, est une démarche très constructive, car cela permet aussi de mieux comprendre les limites et les biais des outils utilisés en routine. L'inertie est confortable, mais dans un cadre scientifique, cela peut dangereusement mener au dogme.

Remerciements

Merci à toutes les personnes qui ont pu m'aider et me supporter ces trois dernières années. Ce fut une période différente de ce que j'avais imaginé, mais néanmoins appréciable par bien des aspects, mais surtout un : l'aspect humain de la recherche scientifique.

En premier lieu, merci à David et Alex, qui m'ont fait confiance pour me prendre en thèse puis accompagné pour ces quelques années. Travailler à vos côtés fut enrichissant, pas seulement scientifiquement. Merci aux membres du LAGBeM passés et présents de manière général. C'est un bel environnement de travail avec de belles personnes, restez comme vous êtes ! J'ai fait de belles rencontres pendant ces années et certaines je l'espère seront pérennes.

Mes hommages à Paul qui a eu à subir mon encadrement pendant un an. J'espère que tu en gardes un bon souvenir, je te souhaite le meilleur pour la suite tu le mérites !

Je souhaite bien du courage aux doctorants du Genoscope passés et présents que j'ai eus la chance de rencontrer.

Merci aussi aux scientifiques qui acceptent d'échanger avec les apprenants de leur domaine. Ces échanges sont souvent enrichissants et peuvent permettre de donner d'autres perspectives et d'autres orientations que lesdits apprenants n'envisage pas forcément. Merci donc de prendre le temps de discuter après un séminaire, au pied d'un poster, en visioconférence, ou autour d'une boisson dans ces événements sociaux liés au monde académique, dans le partage et sans jugement.

J'ai eu la chance de voir plusieurs personnes prendre la suite des travaux auquel j'ai participé. Bon courage à vous ! Si ça se passe mal, dites-vous que c'est ma faute ;)

Merci aux membres du comité de thèse d'avoir accompagné ce travail, et aux membres du jury d'avoir accepté de le juger en présence, "comme à l'ancienne".

Merci à mes amis pour leur indéfectible soutien. Merci à mes parents de me soutenir même si les choses que je fais peuvent sembler un peu étrange ! Sans vous je ne serais jamais allé aussi loin, cette thèse est aussi la vôtre. Merci.

Bibliographie

Abby, Sophie S et al. (2014). "MacSyFinder : a program to mine genomes for molecular systems with an application to CRISPR-Cas systems". In : *PloS one* 9.10, e110726 (cf. pages 93, 173).

Agrawal, Rakesh, Ramakrishnan Srikant et al. (1994). "Fast algorithms for mining association rules". In : *Proc. 20th int. conf. very large data bases, VLDB*. Tome 1215. Citeseer, pages 487-499 (cf. page 97).

Altman, Tomer et al. (2013). "A systematic comparison of the MetaCyc and KEGG pathway databases". In : *BMC bioinformatics* 14.1, pages 1-15 (cf. page 91).

Altschul, Stephen F et al. (1990). "Basic local alignment search tool". In : *Journal of molecular biology* 215.3, pages 403-410 (cf. pages 45, 47).

Ambroise, Christophe, Mo Dang et Gérard Govaert (1997). "Clustering of spatial data by the EM algorithm". In : *geoENV I—Geostatistics for environmental applications*. Springer, pages 493-504 (cf. page 108).

Anand, Swadha et al. (2020). "FunGeCo : a web-based tool for estimation of functional potential of bacterial genomes and microbiomes using gene context information". In : *Bioinformatics* 36.8, pages 2575-2577 (cf. page 93).

Aramaki, Takuya et al. (2020). "KofamKOALA : KEGG ortholog assignment based on profile HMM and adaptive score threshold". In : *Bioinformatics* 36.7, pages 2251-2252 (cf. page 93).

Argos, Patrick et al. (1984). "Similarity in gene organization and homology between proteins of animal picomaviruses and a plant comovirus suggest common ancestry of these virus families". In : *Nucleic Acids Research* 12.18, pages 7251-7267 (cf. page 45).

Arnaud, Martha et al. (2005). *Curator's Guide to Pathway/Genome Databases* (cf. page 91).

Aziz, Ramy K et al. (2008). "The RAST Server : rapid annotations using subsystems technology". In : *BMC genomics* 9.1, pages 1-15 (cf. page 51).

Bailly-Bechet, Marc, Massimo Vergassola et Eduardo Rocha (2007). "Causes for the intriguing presence of tRNAs in phages". In : *Genome research* 17.10, pages 1486-1495 (cf. page 72).

Baker, Daniel N et Ben Langmead (2019). "Dashing : fast and accurate genomic distances with HyperLogLog". In : *Genome biology* 20.1, pages 1-12 (cf. page 57).

Barka, Essaid Ait et al. (2016). "Taxonomy, physiology, and natural products of Actinobacteria". In : *Microbiology and Molecular Biology Reviews* 80.1, pages 1-43 (cf. page 14).

Barker, Daniel et Mark Pagel (2005). "Predicting functional gene links from phylogenetic-statistical analyses of whole genomes". In : *PLoS computational biology* 1.1, e3 (cf. page 101).

Bayliss, Sion C et al. (2019). "PIRATE : A fast and scalable pangenomics toolbox for clustering diverged orthologues in bacteria". In : *Gigascience* 8.10, giz119 (cf. pages 68, 138).

Beck, Christian, Henning Knoop et Ralf Steuer (2018). "Modules of co-occurrence in the cyanobacterial pan-genome reveal functional associations between groups of ortholog genes". In : *PLoS genetics* 14.3, e1007239 (cf. page 94).

Benedict, Matthew N et al. (2014). "ITEP : an integrated toolkit for exploration of microbial pan-genomes". In : *BMC genomics* 15.1, pages 1-11 (cf. page 68).

Bergeron, Anne, Sylvie Corteel et Mathieu Raffinot (2002). "The algorithmic of gene teams". In : *International Workshop on Algorithms in Bioinformatics*. Springer, pages 464-476 (cf. page 97).

Bergey, D.H. et al. (1923). *Bergey's Manual of Determinative Bacteriology, 1st edition*. The Williams et Wilkins Co, Baltimore (cf. page 55).

Bertelli, Claire, Keith E Tilley et Fiona SL Brinkman (2019). "Microbial genomic island discovery, visualization and analysis". In : *Briefings in bioinformatics* 20.5, pages 1685-1698 (cf. page 74).

Bertelli, Claire et al. (2017). "IslandViewer 4 : expanded prediction of genomic islands for larger-scale datasets". In : *Nucleic acids research* 45.W1, W30-W35 (cf. pages 78, 80).

Blin, Kai et al. (2019). "antiSMASH 5.0 : updates to the secondary metabolite genome mining pipeline". In : *Nucleic acids research* 47.W1, W81-W87 (cf. page 93).

Blom, Jochen et al. (2009). "EDGAR : a software framework for the comparative analysis of prokaryotic genomes". In : *BMC bioinformatics* 10.1, pages 1-14 (cf. pages 69, 177).

Blondel, Vincent D et al. (2008). "Fast unfolding of communities in large networks". In : *Journal of statistical mechanics : theory and experiment* 2008.10, P10008 (cf. pages 42, 179).

Bocs, Stephanie et al. (2003). "AMIGene : annotation of microbial genes". In : *Nucleic acids research* 31.13, pages 3723-3726 (cf. page 192).

Boyer, Frédéric et al. (2005). "Syntons, metabolons and interactons : an exact graph-theoretical approach for exploring neighbourhood between genomic and functional data". In : *Bioinformatics* 21.23, pages 4209-4215 (cf. page 98).

Breitwieser, Florian P et Steven L Salzberg (2020). "Pavian : interactive analysis of metagenomics data for microbiome studies and pathogen identification". In : *Bioinformatics* 36.4, pages 1303-1304 (cf. page 182).

Brown, C Titus et Luiz Irber (2016). "sourmash : a library for MinHash sketching of DNA". In : *Journal of Open Source Software* 1.5, page 27 (cf. page 57).

Brutlag, Douglas L et al. (1993). "BLAZE™ : An implementation of the Smith-Waterman sequence comparison algorithm on a massively parallel computer". In : *Computers & chemistry* 17.2, pages 203-207 (cf. page 45).

Brynildsrud, Ola et al. (2016). "Rapid scoring of genes in microbial pan-genome-wide association studies with Scoary". In : *Genome biology* 17.1, pages 1-9 (cf. page 101).

Buchfink, Benjamin, Klaus Reuter et Hajk-Georg Drost (2021). "Sensitive protein alignments at tree-of-life scale using DIAMOND". In : *Nature methods* 18.4, pages 366-368 (cf. page 45).

Buchfink, Benjamin, Chao Xie et Daniel H Huson (2015). "Fast and sensitive protein alignment using DIAMOND". In : *Nature methods* 12.1, pages 59-60 (cf. page 45).

Buchrieser, Carmen et al. (1998). "The high-pathogenicity island of Yersinia pseudotuberculosis can be inserted into any of the three chromosomal asn tRNA genes". In : *Molecular microbiology* 30.5, pages 965-978 (cf. page 73).

Buck, Moritz, Maliheh Mehrshad et Stefan Bertilsson (2021). "mOTUpan : a robust Bayesian approach to leverage metagenome assembled genomes for core-genome estimation". In : *bioRxiv* (cf. pages 68, 138, 189).

Bush, Eliot C et al. (2018). "xenoGI : reconstructing the history of genomic island insertions in clades of closely related bacteria". In : *BMC bioinformatics* 19.1, pages 1-11 (cf. pages 78, 190).

Caspi, Ron et al. (2020). "The MetaCyc database of metabolic pathways and enzymes-a 2019 update". In : *Nucleic acids research* 48.D1, pages D445-D453 (cf. page 91).

Chaudhari, Narendrakumar M et al. (2018). "PanGFR-HM : a dynamic web resource for pan-genomic and functional profiling of human microbiome with comparative features". In : *Frontiers in microbiology* 9, page 2322 (cf. page 177).

Chen, Jingchun et Bo Yuan (2006). "Detecting functional modules in the yeast protein–protein interaction network". In : *Bioinformatics* 22.18, pages 2283-2290 (cf. page 89).

Chen, Xinyu et al. (2018). "PGAweb : a web server for bacterial pan-genome analysis". In : *Frontiers in microbiology* 9, page 1910 (cf. pages 69, 177).

Chiapello, Hélène et al. (2005). "Systematic determination of the mosaic structure of bacterial genomes : species backbone versus strain-specific loops". In : *BMC bioinformatics* 6.1, pages 1-10 (cf. page 76).

Chiapello, Hélène et al. (2008). "MOSAIC : an online database dedicated to the comparative genomics of bacterial strains at the intra-species level". In : *BMC bioinformatics* 9.1, pages 1-9 (cf. page 76).

Cobián Güemes, Ana Georgina et al. (2016). "Viruses as winners in the game of life". In : *Annual Review of Virology* 3, pages 197-214 (cf. page 32).

Colquhoun, Rachel M et al. (2021). "Pandora : nucleotide-resolution bacterial pan-genomics with reference graphs". In : *Genome biology* 22.1, pages 1-30 (cf. page 192).

Contreras-Moreira, Bruno et Pablo Vinuesa (2013). "GET_HOMOLOGUES, a versatile software package for scalable and robust microbial pangenome analysis". In : *Applied and environmental microbiology* 79.24, pages 7696-7701 (cf. page 68).

Couvin, David et al. (2018). "CRISPRCasFinder, an update of CRISRFinder, includes a portable version, enhanced performance and integrates search for Cas proteins". In : *Nucleic acids research* 46.W1, W246-W251 (cf. page 93).

Cury, Jean et al. (2020). "Identifying conjugative plasmids and integrative conjugative elements with CONJscan". In : *Horizontal Gene Transfer*. Springer, pages 265-283 (cf. page 93).

Darling, Aaron CE et al. (2004). "Mauve : multiple alignment of conserved genomic sequence with rearrangements". In : *Genome research* 14.7, pages 1394-1403 (cf. pages 77, 78).

Darling, Aaron E, István Miklós et Mark A Ragan (2008). "Dynamics of genome rearrangement in bacterial populations". In : *PLoS genetics* 4.7, e1000128 (cf. page 27).

Delaye, Luis et al. (2008). "The origin of a novel gene through overprinting in Escherichia coli". In : *BMC Evolutionary Biology* 8.1, pages 1-10 (cf. page 37).

Delcher, Arthur L et al. (2007). "Identifying bacterial genes and endosymbiont DNA with Glimmer". In : *Bioinformatics* 23.6, pages 673-679 (cf. page 192).

Deniélou, Yves-Pol et al. (2011). "Bacterial syntenies : an exact approach with gene quorum". In : *BMC bioinformatics* 12.1, pages 1-15 (cf. page 98).

Dhillon, Bhavjinder K et al. (2013). "IslandViewer update : improved genomic island discovery and visualization". In : *Nucleic acids research* 41.W1, W129-W132 (cf. page 78).

Dhillon, Bhavjinder K et al. (2015). "IslandViewer 3 : more flexible, interactive genomic island discovery, visualization and analysis". In : *Nucleic acids research* 43.W1, W104-W108 (cf. page 78).

Ding, Wei, Franz Baumdicker et Richard A Neher (2018). "panX : pan-genome analysis and exploration". In : *Nucleic acids research* 46.1, e5-e5 (cf. page 177).

Dittrich, Marcus T et al. (2008). "Identifying functional modules in protein–protein interaction networks : an integrated exact approach". In : *Bioinformatics* 24.13, pages i223-i231 (cf. page 89).

Dorward, David W, Claude F Garon et Ralph C Judd (1989). "Export and intercellular transfer of DNA via membrane blebs of Neisseria gonorrhoeae". In : *Journal of bacteriology* 171.5, pages 2499-2505 (cf. page 35).

Dubey, Gyanendra P et Sigal Ben-Yehuda (2011). "Intercellular nanotubes mediate bacterial communication". In : *Cell* 144.4, pages 590-600 (cf. page 35).

Dubnau, David et Melanie Blokesch (2019). "Mechanisms of DNA uptake by naturally competent bacteria". In : *Annual review of genetics* 53, pages 217-237 (cf. page 29).

Durant, Éloi et al. (2021). "Panache : a Web Browser-Based Viewer for Linearized Pangenomes". In : *bioRxiv* (cf. page 68).

Dutkowski, Janusz et Jerzy Tiuryn (2007). "Identification of functional modules from conserved ancestral protein–protein interactions". In : *Bioinformatics* 23.13, pages i149-i158 (cf. page 89).

Dy, Ron L et al. (2014). "Remarkable mechanisms in microbes to resist phage infections". In : *Annual review of virology* 1, pages 307-331 (cf. page 33).

Edgar, Robert C (2010). "Search and clustering orders of magnitude faster than BLAST". In : *Bioinformatics* 26.19, pages 2460-2461 (cf. page 52).

Eisen, Jonathan A et al. (2000). "Evidence for symmetric chromosomal inversions around the replication origin in bacteria". In : *Genome biology* 1.6, pages 1-9 (cf. page 27).

Eisenstark, A (1977). "Genetic recombination in bacteria". In : *Annual review of genetics* 11.1, pages 369-396 (cf. pages 28, 29).

Engelen, Stefan et al. (2012). "Distinct co-evolution patterns of genes associated to DNA polymerase III DnaE and PolC". In : *BMC genomics* 13.1, pages 1-15 (cf. page 94).

Erez, Zohar et al. (2017). "Communication between viruses guides lysis–lysogeny decisions". In : *Nature* 541.7638, pages 488-493 (cf. page 33).

Ferry, Tristan et al. (2021). "Safety of tedizolid as suppressive antimicrobial therapy for patients with complex implant-associated bone and joint infection due to multidrug-resistant Gram-positive pathogens : results from the TediSAT cohort study". In : *Open Forum Infectious Diseases*. Tome 8. 7. Oxford University Press US, ofab351 (cf. page 34).

Filangi, Olivier et al. (2008). "BioMAJ : a flexible framework for databanks synchronization and processing". In : *Bioinformatics* 24.16, pages 1823-1825 (cf. page 178).

FITCH, Walter M (1966). "An improved method of testing for evolutionary homology". In : *Journal of molecular biology* 16.1, pages 9-16 (cf. page 43).

FLEMING, Alexander (1929). "On the antibacterial action of cultures of a penicillium, with special reference to their use in the isolation of B. influenzae". In : *British journal of experimental pathology* 10.3, page 226 (cf. page 13).

FLORES RAMOS, Stephany et al. (2021). "Genomic Stability and Genetic Defense Systems in Dolosigranulum pigrum, a Candidate Beneficial Bacterium from the Human Microbiome". In : *mSystems* 6.5, e00425-21 (cf. pages 82, 188).

FOUTS, Derrick E et al. (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, e172-e172 (cf. page 68).

FRANCESCHINI, Andrea et al. (2012). "STRING v9. 1 : protein-protein interaction networks, with increased coverage and integration". In : *Nucleic acids research* 41.D1, pages D808-D815 (cf. page 96).

FRANCESCHINI, Andrea et al. (2016). "SVD-phy : improved prediction of protein functional associations through singular value decomposition of phylogenetic profiles". In : *Bioinformatics* 32.7, pages 1085-1087 (cf. page 94).

FRASER, Claire M et al. (1995). "The minimal gene complement of Mycoplasma genitalium". In : *Science* 270.5235, pages 397-404 (cf. page 45).

FRITZ, Susanne A et Andy PURVIS (2010). "Selectivity in mammalian extinction risk and threat types : a new measure of phylogenetic signal strength in binary traits". In : *Conservation Biology* 24.4, pages 1042-1051 (cf. page 191).

FROMONT-RACINE, Micheline, Jean-Christophe RAIN et Pierre LEGRAIN (1997). "Toward a functional analysis of the yeast genome through exhaustive two-hybrid screens". In : *Nature genetics* 16.3, pages 277-282 (cf. page 88).

FU, Limin et al. (2012). "CD-HIT : accelerated for clustering the next-generation sequencing data". In : *Bioinformatics* 28.23, pages 3150-3152 (cf. page 48).

FUJISAWA, T et A EISENSTARK (1973). "Bi-directional chromosomal replication in Salmonella typhimurium". In : *Journal of bacteriology* 115.1, pages 168-176 (cf. page 24).

GALPERIN, Michael Y et al. (2021). "COG database update : focus on microbial diversity, model organisms, and widespread pathogens". In : *Nucleic acids research* 49.D1, pages D274-D281 (cf. page 47).

GAUTREAU, Guillaume (2020). "Conceptualisation et exploitation d'un graphe de pangénome partitionné comme représentation compacte de la diversité du répertoire génique des espèces procaryotes". Thèse de doctorat (cf. pages 62, 109).

GEMAYEL, Karl, Alexandre LOMSADZE et Mark BORODOVSKY (2020). "Start-Link+ : Prediction of Gene Starts in Prokaryotic Genomes by an Algorithm Integrating Independent Sources of Evidence". In : *bioRxiv* (cf. page 192).

GIBBONS, Henry S et al. (2011). "Genomic signatures of strain selection and enhancement in Bacillus atrophaeus var. globigii, a historical biowarfare simulant". In : *PLoS One* 6.3, e17836 (cf. page 32).

GLAZKO, Galina V et Arcady R MUSHEGIAN (2004). "Detection of evolutionarily stable fragments of cellular pathways by hierarchical clustering of phyletic patterns". In : *Genome biology* 5.5, pages 1-13 (cf. page 94).

GRIFFITH, Fred (1928). "The significance of pneumococcal types". In : *Epidemiology & Infection* 27.2, pages 113-159 (cf. page 29).

GROISMAN, Eduardo A, Anne-Béatrice BLANC-POTARD et Keiichi UCHIYA (1999). "Pathogenicity islands and the evolution of Salmonella virulence". In : *Pathogenicity islands and other mobile virulence elements*, pages 127-150 (cf. pages 74, 79).

GUO, Feng-Biao, Hong-Yu OU et Chun-Ting ZHANG (2003). "ZCURVE : a new system for recognizing protein-coding genes in bacterial and archaeal genomes". In : *Nucleic acids research* 31.6, pages 1780-1789 (cf. page 192).

HACKER, Jörg et Elisabeth CARNIEL (2001). "Ecological fitness, genomic islands and bacterial pathogenicity". In : *EMBO reports* 2.5, pages 376-381 (cf. pages 72, 73).

HACKER, Jörg et James B KAPER (2000). "Pathogenicity islands and the evolution of microbes". In : *Annual Reviews in Microbiology* 54.1, pages 641-679 (cf. page 71).

HACKER, Jörg et al. (1990). "Deletions of chromosomal regions coding for fimbriae and hemolysins occur in vitro and in vivo in various extra intestinal Escherichia coli isolates". In : *Microbial pathogenesis* 8.3, pages 213-225 (cf. page 74).

HAUSER, Maria, Christian E MAYER et Johannes SÖDING (2013). "kClust : fast and sensitive clustering of large protein sequence databases". In : *BMC bioinformatics* 14.1, pages 1-12 (cf. page 52).

HAUSER, Maria, Martin STEINEGGER et Johannes SÖDING (2016). "MMseqs software suite for fast and deep clustering and searching of large protein sequence sets". In : *Bioinformatics* 32.9, pages 1323-1330 (cf. pages 52, 53).

HE, Xin et Michael H GOLDWASSER (2004). "Identifying conserved gene clusters in the presence of orthologous groups". In : *Proceedings of the eighth annual international conference on Resaerch in computational molecular biology*, pages 272-280 (cf. page 97).

HENNIG, André, Jörg BERNHARDT et Kay NIESELT (2015). "Pan-Tetris : an interactive visualisation for Pan-genomes". In : *BMC bioinformatics* 16.11, pages 1-11 (cf. page 68).

Hogg, Justin S et al. (2007). "Characterization and modeling of the Haemophilus influenzae core and supragenomes based on the complete genomic sequences of Rd and 12 clinical nontypeable strains". In : *Genome biology* 8.6, pages 1-18 (cf. pages 64, 67).

Hsiao, William et al. (2003). "IslandPath : aiding detection of genomic islands in prokaryotes". In : *Bioinformatics* 19.3, pages 418-420 (cf. page 75).

Hsiao, William W L et al. (2005). "Evidence of a large novel gene pool associated with prokaryotic genomic islands". In : *PLoS genetics* 1.5, e62 (cf. page 75).

Huang, Ying et al. (2010). "CD-HIT Suite : a web server for clustering and comparing biological sequences". In : *Bioinformatics* 26.5, pages 680-682 (cf. page 48).

Huynen, Martijn et al. (2000). "Predicting protein function by genomic context : quantitative evaluation and qualitative inferences". In : *Genome research* 10.8, pages 1204-1210 (cf. page 94).

Hyatt, Doug et al. (2010). "Prodigal : prokaryotic gene recognition and translation initiation site identification". In : *BMC bioinformatics* 11.1, pages 1-11 (cf. page 192).

Jain, Chirag et al. (2018). "High throughput ANI analysis of 90K prokaryotic genomes reveals clear species boundaries". In : *Nature communications* 9.1, pages 1-8 (cf. pages 57-60).

Jani, Mehul et Rajeev K Azad (2019). "IslandCafe : compositional anomaly and feature enrichment assessment for delineation of genomic islands". In : *G3 : Genes, Genomes, Genetics* 9.10, pages 3273-3285 (cf. page 76).

Jensen, Lars J et al. (2009). "STRING 8—a global view on proteins and their functional interactions in 630 organisms". In : *Nucleic acids research* 37.suppl_1, pages D412-D416 (cf. page 96).

Jensen, Lars Juhl et al. (2007). "eggNOG : automated construction and annotation of orthologous groups of genes". In : *Nucleic acids research* 36.suppl_1, pages D250-D254 (cf. pages 48, 67).

Jinek, Martin et al. (2012). "A programmable dual-RNA–guided DNA endonuclease in adaptive bacterial immunity". In : *science* 337.6096, pages 816-821 (cf. page 33).

Johnson, Christopher M et Alan D Grossman (2015). "Integrative and conjugative elements (ICEs) : what they do and how they work". In : *Annual review of genetics* 49, pages 577-601 (cf. page 30).

Kanehisa, Minoru, Yoko Sato et Kanae Morishima (2016). "BlastKOALA and GhostKOALA : KEGG tools for functional characterization of genome and metagenome sequences". In : *Journal of molecular biology* 428.4, pages 726-731 (cf. page 93).

Kanehisa, Minoru et al. (2021). "KEGG : integrating viruses and cellular organisms". In : *Nucleic acids research* 49.D1, pages D545-D551 (cf. page 91).

KAPER, James B, Jay L MELLIES et James P NATARO (1999). "Pathogenicity islands and other mobile genetic elements of diarrheagenic Escherichia coli". In : *Pathogenicity islands and other mobile virulence elements*, pages 33-58 (cf. page 74).

KARAOLIS, David KR et al. (1999). "A bacteriophage encoding a pathogenicity island, a type-IV pilus and a phage receptor in cholera bacteria". In : *Nature* 399.6734, pages 375-379 (cf. page 73).

KARLIN, Samuel (1998). "Global dinucleotide signatures and analysis of genomic heterogeneity". In : *Current opinion in microbiology* 1.5, pages 598-610 (cf. page 75).

— (2001). "Detecting anomalous gene clusters and pathogenicity islands in diverse bacterial genomes". In : *Trends in microbiology* 9.7, pages 335-343 (cf. page 74).

KIEŁBASA, Szymon M et al. (2011). "Adaptive seeds tame genomic sequence comparison". In : *Genome research* 21.3, pages 487-493 (cf. page 45).

KONSTANTINIDIS, Konstantinos T et James M TIEDJE (2005). "Genomic insights that advance the species definition for prokaryotes". In : *Proceedings of the National Academy of Sciences* 102.7, pages 2567-2572 (cf. pages 56, 179).

KOONIN, Eugene V et Yuri I WOLF (2008). "Genomics of bacteria and archaea : the emerging dynamic view of the prokaryotic world". In : *Nucleic acids research* 36.21, pages 6688-6719 (cf. page 67).

KRISTENSEN, David M, Xixu CAI et Arcady MUSHEGIAN (2011). "Evolutionarily conserved orthologous families in phages are relatively rare in their prokaryotic hosts". In : *Journal of Bacteriology* 193.8, pages 1806-1814 (cf. page 48).

KRISTENSEN, David M et al. (2010). "A low-polynomial algorithm for assembling clusters of orthologous groups from intergenomic symmetric best matches". In : *Bioinformatics* 26.12, pages 1481-1487 (cf. page 47).

KULP, Adam et Meta J KUEHN (2010). "Biological functions and biogenesis of secreted bacterial outer membrane vesicles". In : *Annual review of microbiology* 64, pages 163-184 (cf. page 35).

LAING, Chad et al. (2010). "Pan-genome sequence analysis using Panseq : an online tool for the rapid analysis of core and accessory genomic regions". In : *BMC bioinformatics* 11.1, pages 1-14 (cf. page 68).

LANG, Andrew S, Alexander B WESTBYE et J Thomas BEATTY (2017). "The distribution, evolution, and roles of gene transfer agents in prokaryotic genetic exchange". In : *Annual review of virology* 4, pages 87-104 (cf. page 34).

LANG, Andrew S, Olga ZHAXYBAYEVA et J Thomas BEATTY (2012). "Gene transfer agents : phage-like elements of genetic exchange". In : *Nature Reviews Microbiology* 10.7, pages 472-482 (cf. page 35).

Langille, Morgan GI et Fiona SL Brinkman (2009). "IslandViewer : an integrated interface for computational identification and visualization of genomic islands". In : *Bioinformatics* 25.5, pages 664-665 (cf. page 78).

Langille, Morgan GI, William WL Hsiao et Fiona SL Brinkman (2008). "Evaluation of genomic island predictors using a comparative genomics approach". In : *BMC bioinformatics* 9.1, pages 1-10 (cf. pages 75, 78).

Lassalle, Florent et al. (2019). "Automated reconstruction of all gene histories in large bacterial pangenome datasets and search for co-evolved gene modules with Pantagruel". In : *BioRxiv*, page 586495 (cf. page 101).

Lathe III, Warren C, Berend Snel et Peer Bork (2000). "Gene context conservation of a higher order than operons". In : *Trends in biochemical sciences* 25.10, pages 474-479 (cf. pages 24, 91, 92).

Lescat, Mathilde et al. (2009). "A module located at a chromosomal integration hot spot is responsible for the multidrug resistance of a reference strain from Escherichia coli clonal group A". In : *Antimicrobial agents and chemotherapy* 53.6, pages 2283-2288 (cf. pages 72, 81, 83, 153).

Li, Li, Christian J Stoeckert et David S Roos (2003). "OrthoMCL : identification of ortholog groups for eukaryotic genomes". In : *Genome research* 13.9, pages 2178-2189 (cf. page 50).

Li, Weizhong et Adam Godzik (2006). "Cd-hit : a fast program for clustering and comparing large sets of protein or nucleotide sequences". In : *Bioinformatics* 22.13, pages 1658-1659 (cf. page 48).

Li, Weizhong, Lukasz Jaroszewski et Adam Godzik (2001). "Clustering of highly homologous sequences to reduce the size of large protein databases". In : *Bioinformatics* 17.3, pages 282-283 (cf. page 48).

Ling, Xu, Xin He et Dong Xin (2009). "Detecting gene clusters under evolutionary constraint in a large number of genomes". In : *Bioinformatics* 25.5, pages 571-577 (cf. page 97).

Lio, Pietro et Marina Vannucci (2000). "Finding pathogenicity islands and gene transfer events in genome data". In : *Bioinformatics* 16.10, pages 932-940 (cf. page 75).

Lomsadze, Alexandre et al. (2018). "Modeling leaderless transcription and atypical genes results in more accurate gene prediction in prokaryotes". In : *Genome research* 28.7, pages 1079-1089 (cf. page 192).

Lukjancenko, Oksana, Trudy M Wassenaar et David W Ussery (2010). "Comparison of 61 sequenced Escherichia coli genomes". In : *Microbial ecology* 60.4, pages 708-720 (cf. page 76).

Luo, Chengwei et al. (2011). "Genome sequencing of environmental Escherichia coli expands understanding of the ecology and speciation of the model bac-

terial species". In : *Proceedings of the National Academy of Sciences* 108.17, pages 7200-7205 (cf. page 58).

Lyons, Nicholas A et al. (2016). "A combinatorial kin discrimination system in Bacillus subtilis". In : *Current Biology* 26.6, pages 733-742 (cf. page 30).

Mackiewicz, Paweł et al. (2001). "Flip-flop around the origin and terminus of replication in prokaryotic genomes". In : *Genome biology* 2.12, pages 1-4 (cf. page 28).

Maddison, Wayne P et Richard G FitzJohn (2015). "The unsolved challenge to phylogenetic correlation tests for categorical characters". In : *Systematic biology* 64.1, pages 127-136 (cf. page 101).

Makarova, Kira S et al. (2007). "Clusters of orthologous genes for 41 archaeal genomes and implications for evolutionary genomics of archaea". In : *Biology direct* 2.1, pages 1-20 (cf. pages 48, 67).

Mandel, Mt (1969). "New approaches to bacterial taxonomy : perspective and prospects". In : *Annual review of microbiology* 23.1, pages 239-274 (cf. pages 55, 61).

Manolov, Alexander et al. (2020). "Genome Complexity Browser : Visualization and quantification of genome variability". In : *PLoS computational biology* 16.10, e1008222 (cf. page 81).

Mantri, Yogita et Kelly P Williams (2004). "Islander : a database of integrative islands in prokaryotic genomes, the associated integrases and their DNA site specificities". In : *Nucleic acids research* 32.suppl_1, pages D55-D58 (cf. page 75).

Mao, Xizeng et al. (2014). "DOOR 2.0 : presenting operons and their functions through dynamic and integrated views". In : *Nucleic acids research* 42.D1, pages D654-D659 (cf. page 91).

Marcotte, Edward M et al. (1999). "Detecting protein function and protein-protein interactions from genome sequences". In : *Science* 285.5428, pages 751-753 (cf. page 90).

Marmur, J, S Falkow et M Mandel (1963). "New approaches to bacterial taxonomy". In : *Annual Reviews in Microbiology* 17.1, pages 329-372 (cf. page 55).

Martinez-Murcia, AJ, S Benlloch et MD Collins (1992). "Phylogenetic interrelationships of members of the genera Aeromonas and Plesiomonas as determined by 16S ribosomal DNA sequencing : lack of congruence with results of DNA-DNA hybridizations". In : *International Journal of systematic and evolutionary microbiology* 42.3, pages 412-421 (cf. page 56).

Médigue, Claudine et al. (1991). "Evidence for horizontal gene transfer in Escherichia coli speciation". In : *Journal of molecular biology* 222.4, pages 851-856 (cf. page 75).

MEDINI, Duccio et al. (2005). "The microbial pan-genome". In : *Current opinion in genetics & development* 15.6, pages 589-594 (cf. page 63).

MERKL, Rainer (2004). "SIGI : score-based identification of genomic islands". In : *BMC bioinformatics* 5.1, pages 1-14 (cf. page 75).

MILO, Ron et Rob PHILLIPS (2015). *Cell biology by the numbers*. Garland Science (cf. page 21).

MUIR, Alastair et al. (2020). "Construction of a complete set of Neisseria meningitidis mutants and its use for the phenotypic profiling of this human pathogen". In : *Nature communications* 11.1, pages 1-13 (cf. page 188).

MULEY, Vijaykumar Yogesh et Akash RANJAN (2012). "Effect of reference genome selection on the performance of computational methods for genome-wide protein-protein interaction prediction". In : (cf. page 94).

MURRAY, Connor S, Yingnan GAO et Martin WU (2021). "Re-evaluating the evidence for a universal genetic boundary among microbial species". In : *Nature communications* 12.1, pages 1-7 (cf. page 61).

NAOR, Adit et Uri GOPHNA (2013). "Cell fusion and hybrids in Archaea : prospects for genome shuffling and accelerated strain development for biotechnology". In : *Bioengineered* 4.3, pages 126-129 (cf. page 35).

NEEDLEMAN, Saul B et Christian D WUNSCH (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, pages 443-453 (cf. pages 43, 44).

NOGUCHI, Hideki, Takeaki TANIGUCHI et Takehiko ITOH (2008). "MetaGeneAnnotator : detecting species-specific patterns of ribosomal binding site for precise gene prediction in anonymous prokaryotic and phage genomes". In : *DNA research* 15.6, pages 387-396 (cf. page 192).

O'BRIEN, Kevin P, Maido REMM et Erik LL SONNHAMMER (2005). "Inparanoid : a comprehensive database of eukaryotic orthologs". In : *Nucleic acids research* 33.suppl_1, pages D476-D480 (cf. page 50).

OGIER, Jean-Claude et al. (2010). "Units of plasticity in bacterial genomes : new insight from the comparative genomics of two bacteria interacting with invertebrates, Photorhabdus and Xenorhabdus". In : *BMC genomics* 11.1, pages 1-21 (cf. page 83).

OLIVEIRA, Pedro H et al. (2017). "The chromosomal organization of horizontal gene transfer in bacteria". In : *Nature communications* 8.1, pages 1-11 (cf. pages 81, 82, 141, 151).

OLM, Matthew R et al. (2017). "dRep : a tool for fast and accurate genomic comparisons that enables improved genome recovery from metagenomes through de-replication". In : *The ISME journal* 11.12, pages 2864-2868 (cf. page 191).

Olm, Matthew R et al. (2020). "Consistent metagenome-derived metrics verify and delineate bacterial species boundaries". In : *Msystems* 5.1, e00731-19 (cf. page 59).

Ondov, Brian D et al. (2016). "Mash : fast genome and metagenome distance estimation using MinHash". In : *Genome biology* 17.1, pages 1-14 (cf. page 57).

Onesime, Mbulayi, Zhenyu Yang et Qi Dai (2021). "Genomic Island Prediction via Chi-Square Test and Random Forest Algorithm". In : *Computational and Mathematical Methods in Medicine* 2021 (cf. page 76).

Orchard, Sandra et al. (2012). "Protein interaction data curation : the International Molecular Exchange (IMEx) consortium". In : *Nature methods* 9.4, pages 345-350 (cf. page 88).

Orchard, Sandra et al. (2014). "The MIntAct project—IntAct as a common curation platform for 11 molecular interaction databases". In : *Nucleic acids research* 42.D1, pages D358-D363 (cf. page 88).

Östlund, Gabriel et al. (2010). "InParanoid 7 : new algorithms and tools for eukaryotic orthology analysis". In : *Nucleic acids research* 38.suppl_1, pages D196-D203 (cf. page 50).

Ou, Hong-Yu et al. (2006). "A novel strategy for the identification of genomic islands by comparative analysis of the contents and contexts of tRNA sites in closely related bacteria". In : *Nucleic acids research* 34.1, e3-e3 (cf. pages 77, 81).

Ou, Hong-Yu et al. (2007). "MobilomeFINDER : web-based tools for in silico and experimental discovery of bacterial genomic islands". In : *Nucleic acids research* 35.suppl_2, W97-W104 (cf. page 78).

Overbeek, Ross et al. (1999). "The use of gene clusters to infer functional coupling". In : *Proceedings of the National Academy of Sciences* 96.6, pages 2896-2901 (cf. page 95).

Overbeek, Ross et al. (2005). "The subsystems approach to genome annotation and its use in the project to annotate 1000 genomes". In : *Nucleic acids research* 33.17, pages 5691-5702 (cf. pages 51, 173).

Page, Andrew J et al. (2015). "Roary : rapid large-scale prokaryote pan genome analysis". In : *Bioinformatics* 31.22, pages 3691-3693 (cf. pages 68, 138).

Pantoja, Yan et al. (2017). "PanWeb : a web interface for pan-genomic analysis". In : *PLoS One* 12.5, e0178154 (cf. pages 69, 177).

Parks, Donovan H et al. (2018). "A standardized bacterial taxonomy based on genome phylogeny substantially revises the tree of life". In : *Nature biotechnology* 36.10, pages 996-1004 (cf. pages 59, 61, 62, 179).

Pasek, Sophie et al. (2005). "Identification of genomic features using microsyntenies of domains : domain teams". In : *Genome research* 15.6, pages 867-874 (cf. page 97).

Passarge, Eberhard, Bernhard Horsthemke et Rosann A Farber (1999). "Incorrect use of the term synteny". In : *Nature genetics* 23.4, pages 387-387 (cf. page 95).

Pathmanathan, Jananan Sylvestre et al. (2018). "CompositeSearch : a generalized network approach for composite gene families detection". In : *Molecular biology and evolution* 35.1, pages 252-255 (cf. page 90).

Pazos, Florencio et Alfonso Valencia (2001). "Similarity of phylogenetic trees as indicator of protein–protein interaction". In : *Protein engineering* 14.9, pages 609-614 (cf. page 101).

Pedersen, Thomas Lin et al. (2017). "PanViz : interactive visualization of the structure of functionally annotated pangenomes". In : *Bioinformatics* 33.7, pages 1081-1082 (cf. page 68).

Pellegrini, Matteo et al. (1999). "Assigning protein functions by comparative genome analysis : protein phylogenetic profiles". In : *Proceedings of the National Academy of Sciences* 96.8, pages 4285-4288 (cf. page 94).

Perrin, Amandine et Eduardo PC Rocha (2021). "PanACoTA : A modular tool for massive microbial comparative genomics". In : *NAR genomics and bioinformatics* 3.1, lqaa106 (cf. page 68).

Pfeifer, Eugen et al. (2021). "Bacteria have numerous distinctive groups of phage–plasmids with conserved phage and variable plasmid gene repertoires". In : *Nucleic acids research* 49.5, pages 2655-2673 (cf. page 188).

Plaza Oñate, Florian et al. (2019). "MSPminer : abundance-based reconstitution of microbial pan-genomes from shotgun metagenomic data". In : *Bioinformatics* 35.9, pages 1544-1552 (cf. page 68).

Raghavan, Rahul et al. (2015). "Genome rearrangements can make and break small RNA genes". In : *Genome biology and evolution* 7.2, pages 557-566 (cf. page 37).

Rain, Jean-Christophe et al. (2001). "The protein–protein interaction map of Helicobacter pylori". In : *Nature* 409.6817, pages 211-215 (cf. page 88).

Remm, Maido, Christian EV Storm et Erik LL Sonnhammer (2001). "Automatic clustering of orthologs and in-paralogs from pairwise species comparisons". In : *Journal of molecular biology* 314.5, pages 1041-1052 (cf. page 49).

Rinke, Christian et al. (2013). "Insights into the phylogeny and coding potential of microbial dark matter". In : *Nature* 499.7459, pages 431-437 (cf. page 181).

Rocha, Eduardo PC et Antoine Danchin (2003). "Essentiality, not expressiveness, drives gene-strand bias in bacteria". In : *Nature genetics* 34.4, pages 377-378 (cf. page 24).

Rödelsperger, Christian et Christoph Dieterich (2007). "Two Graph-based Approaches for Finding Cross-species Conserved Gene Orders." In : *German Conference on Bioinformatics*, pages 163-173 (cf. page 97).

— (2010). "CYNTENATOR : progressive gene order alignment of 17 vertebrate genomes". In : *PloS one* 5.1, e8861 (cf. page 98).

Rodriguez-R, Luis M et al. (2021). "Reply to :"Re-evaluating the evidence for a universal genetic boundary among microbial species"". In : *Nature communications* 12.1, pages 1-7 (cf. page 61).

Sahl, Jason W et al. (2014). "The large-scale blast score ratio (LS-BSR) pipeline : a method to rapidly compare genetic content between bacterial genomes". In : *PeerJ* 2, e332 (cf. page 68).

Salazar, Alex N et Thomas Abeel (2018). "Approximate, simultaneous comparison of microbial genome architectures via syntenic anchoring of quiver representations". In : *Bioinformatics* 34.17, pages i732-i742 (cf. page 138).

Salwinski, Lukasz et al. (2004). "The database of interacting proteins : 2004 update". In : *Nucleic acids research* 32.suppl_1, pages D449-D451 (cf. page 88).

Santos-Zavaleta, Alberto et al. (2019). "RegulonDB v 10.5 : tackling challenges to unify classic and high throughput knowledge of gene regulation in E. coli K-12". In : *Nucleic acids research* 47.D1, pages D212-D220 (cf. page 90).

El-Sayed, Mohammed H (2012). "Di-(2-ethylhexyl) Phthalate, a major bioactive metabolite with antimicrobial and cytotoxic activity isolated from the culture filtrate of newly isolated soil Streptomyces (Streptomyces mirabilis strain NSQu-25)". In : *World Applied Sciences Journal* 20.9, pages 1202-1212 (cf. page 181).

Schmidt, André et al. (2009). "Heavy metal resistance to the extreme : Streptomyces strains from a former uranium mining area". In : *Geochemistry* 69, pages 35-44 (cf. page 181).

Schmidt, Thomas et Jens Stoye (2007). "Gecko and GhostFam". In : *Comparative Genomics*. Springer, pages 165-182 (cf. page 99).

Scholz, Matthias et al. (2016). "Strain-level microbial epidemiology and population genomics from shotgun metagenomics". In : *Nature methods* 13.5, pages 435-438 (cf. page 68).

Schwengers, Oliver et al. (2021). "Bakta : rapid and standardized annotation of bacterial genomes via alignment-free sequence identification". In : *Microbial Genomics* 7.11, 000685. issn : 2057-5858. doi : https://doi.org/10.1099/mgen.0.000685. url : https://www.microbiologyresearch.org/content/journal/mgen/10.1099/mgen.0.000685 (cf. page 192).

Shapiro, B Jesse et al. (2012). "Population genomics of early events in the ecological differentiation of bacteria". In : *science* 336.6077, pages 48-51 (cf. page 58).

Sharp, Paul M et al. (1989). "Chromosomal location and evolutionary rate variation in enterobacterial genes". In : *Science* 246.4931, pages 808-810 (cf. page 24).

SHIEH, Yu-Wei et al. (2015). "Operon structure and cotranslational subunit association direct protein assembly in bacteria". In : *Science* 350.6261, pages 678-680 (cf. page 90).

SHULGINA, Yekaterina et Sean R EDDY (nov. 2021). "A computational screen for alternative genetic codes in over 250,000 genomes". In : *eLife* 10. Sous la direction d'Eugene V KOONIN, e71402. ISSN : 2050-084X. DOI : 10.7554/eLife.71402. URL : https://doi.org/10.7554/eLife.71402 (cf. page 20).

ŠKUNCA, Nives et Christophe DESSIMOZ (2015). "Phylogenetic profiling : how much input data is enough ?" In : *PloS one* 10.2, e0114701 (cf. page 94).

SMITH, Temple F, Michael S WATERMAN et al. (1981). "Identification of common molecular subsequences". In : *Journal of molecular biology* 147.1, pages 195-197 (cf. pages 44, 45, 63).

SNEL, Berend, Peer BORK et Martijn HUYNEN (2000a). "Genome evolution : gene fusion versus gene fission". In : *Trends in genetics* 16.1, pages 9-11 (cf. page 90).

SNEL, Berend, Peer BORK et Martijn A HUYNEN (2002). "The identification of functional modules from the genomic association of genes". In : *Proceedings of the National Academy of Sciences* 99.9, pages 5890-5895 (cf. page 96).

SNEL, Berend et Martijn A HUYNEN (2004). "Quantifying modularity in the evolution of biomolecular systems". In : *Genome research* 14.3, pages 391-397 (cf. pages 87, 102).

SNEL, Berend et al. (2000b). "STRING : a web-server to retrieve and display the repeatedly occurring neighbourhood of a gene". In : *Nucleic acids research* 28.18, pages 3442-3444 (cf. page 96).

SNIPEN, Lars, Trygve ALMØY et David W USSERY (2009). "Microbial comparative pan-genomics using binomial mixture models". In : *BMC genomics* 10.1, pages 1-8 (cf. pages 65, 67, 107).

SNIPEN, Lars et Kristian Hovde LILAND (2015). "micropan : an R-package for microbial pan-genomics". In : *BMC bioinformatics* 16.1, pages 1-8 (cf. pages 68, 107, 189).

SOMMER, Markus J et Steven L SALZBERG (2021). "Balrog : A universal protein model for prokaryotic gene prediction". In : *PLoS computational biology* 17.2, e1008727 (cf. page 192).

SONNHAMMER, Erik LL et Gabriel ÖSTLUND (2015). "InParanoid 8 : orthology analysis between 273 proteomes, mostly eukaryotic". In : *Nucleic acids research* 43.D1, pages D234-D239 (cf. page 50).

STACKEBRANDT, EaBMG et Brett M GOEBEL (1994). "Taxonomic note : a place for DNA-DNA reassociation and 16S rRNA sequence analysis in the present species definition in bacteriology". In : *International journal of systematic and evolutionary microbiology* 44.4, pages 846-849 (cf. page 56).

STACKEBRANDT, Erko (2006). "Taxonomic parameters revisited : tarnished gold standards". In : *Microbiol. Today* 33, pages 152-155 (cf. page 56).

STANLEY, Sabrina Y et Karen L MAXWELL (2018). "Phage-encoded anti-CRISPR defenses". In : *Annual review of genetics* 52, pages 445-464 (cf. page 33).

STEINEGGER, Martin et Johannes SÖDING (2017). "MMseqs2 enables sensitive protein sequence searching for the analysis of massive data sets". In : *Nature biotechnology* 35.11, pages 1026-1028 (cf. pages 49, 52, 53, 137).

— (2018). "Clustering huge protein sequence sets in linear time". In : *Nature communications* 9.1, pages 1-8 (cf. pages 52, 53).

STEWART, Gregory J et Curtis A CARLSON (1986). "The biology of natural transformation". In : *Annual Reviews in Microbiology* 40.1, pages 211-231 (cf. page 29).

SVETLITSKY, Dina, Tal DAGAN et Michal ZIV-UKELSON (2020). "Discovery of multi-operon colinear syntenic blocks in microbial genomes". In : *Bioinformatics* 36.Supplement_1, pages i21-i29 (cf. page 100).

SVETLITSKY, Dina et al. (2019). "CSBFinder : discovery of colinear syntenic blocks across thousands of prokaryotic genomes". In : *Bioinformatics* 35.10, pages 1634-1643 (cf. page 100).

SZKLARCZYK, Damian et al. (2010). "The STRING database in 2011 : functional interaction networks of proteins, globally integrated and scored". In : *Nucleic acids research* 39.suppl_1, pages D561-D568 (cf. page 96).

SZKLARCZYK, Damian et al. (2015). "STRING v10 : protein–protein interaction networks, integrated over the tree of life". In : *Nucleic acids research* 43.D1, pages D447-D452 (cf. page 96).

SZKLARCZYK, Damian et al. (2016). "The STRING database in 2017 : quality-controlled protein–protein association networks, made broadly accessible". In : *Nucleic acids research*, gkw937 (cf. page 96).

SZKLARCZYK, Damian et al. (2021). "The STRING database in 2021 : customizable protein–protein networks, and functional characterization of user-uploaded gene/measurement sets". In : *Nucleic acids research* 49.D1, pages D605-D612 (cf. page 96).

SZÖLLŐSI, Gergely J et al. (2012). "Phylogenetic modeling of lateral gene transfer reconstructs the pattern and relative timing of speciations". In : *Proceedings of the national academy of sciences* 109.43, pages 17513-17518 (cf. page 36).

TAMAMES, Javier et al. (1997). "Conserved clusters of functionally related genes in two bacterial genomes". In : *Journal of molecular evolution* 44.1, pages 66-73 (cf. page 95).

TATUSOV, Roman L, Eugene V KOONIN et David J LIPMAN (1997). "A genomic perspective on protein families". In : *Science* 278.5338, pages 631-637 (cf. pages 46, 48).

TATUSOV, Roman L et al. (2003). "The COG database : an updated version includes eukaryotes". In : *BMC bioinformatics* 4.1, pages 1-14 (cf. page 47).

TATUSOVA, Tatiana et al. (juin 2016). "NCBI prokaryotic genome annotation pipeline". In : *Nucleic Acids Research* 44.14, pages 6614-6624. ISSN : 0305-1048. DOI : 10.1093/nar/gkw569. eprint : https://academic.oup.com/nar/article-pdf/44/14/6614/7629591/gkw569.pdf. URL : https://doi.org/10.1093/nar/gkw569 (cf. page 192).

TESSON, Florian et al. (2021). "Systematic and quantitative view of the antiviral arsenal of prokaryotes". In : *bioRxiv* (cf. page 93).

TETTELIN, Hervé et al. (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, pages 13950-13955 (cf. pages 63, 64, 67, 107).

TETTELIN, Hervé et al. (2008). "Comparative genomics : the bacterial pan-genome". In : *Current opinion in microbiology* 11.5, pages 472-477 (cf. pages 65, 66).

THORPE, Harry A et al. (2018). "Piggy : a rapid, large-scale pan-genome analysis tool for intergenic regions in bacteria". In : *Gigascience* 7.4, giy015 (cf. page 68).

TONKIN-HILL, Gerry et al. (2020). "Producing polished prokaryotic pangenomes with the Panaroo pipeline". In : *Genome biology* 21.1, pages 1-21 (cf. pages 68, 138).

TOUCHON, Marie et al. (2009). "Organised genome dynamics in the Escherichia coli species results in highly diverse adaptive paths". In : *PLoS genetics* 5.1, e1000344 (cf. pages 72, 81, 83).

TREANGEN, Todd J et Eduardo PC ROCHA (2011). "Horizontal transfer, not duplication, drives the expansion of protein families in prokaryotes". In : *PLoS genetics* 7.1, e1001284 (cf. page 36).

TRIA, Fernando DK et William F MARTIN (2021). "Gene duplications are at least 50 times less frequent than gene transfers in prokaryotic genomes". In : *Genome Biology and Evolution* (cf. page 36).

URHAN, Aysun et Thomas ABEEL (2021). "A comparative study of pan-genome methods for microbial organisms : Acinetobacter baumannii pan-genome reveals structural variation in antimicrobial resistance-carrying plasmids". In : *Microbial Genomics* 7.11, 000690. ISSN : 2057-5858. DOI : https://doi.org/10.1099/mgen.0.000690. URL : https://www.microbiologyresearch.org/content/journal/mgen/10.1099/mgen.0.000690 (cf. page 138).

VALENTINE, PETER J, NADJA B SHOEMAKER et ABIGAIL A SALYERS (1988). "Mobilization of Bacteroides plasmids by Bacteroides conjugal elements". In : *Journal of bacteriology* 170.3, pages 1319-1324 (cf. page 32).

VALLENET, David et al. (2017). "MicroScope in 2017 : an expanding and evolving integrated resource for community expertise of microbial genomes". In : *Nucleic acids research* 45.D1, pages D517-D528 (cf. page 177).

VALLENET, David et al. (2020). "MicroScope : an integrated platform for the annotation and exploration of microbial gene functions through genomic, pangenomic and metabolic comparative analysis". In : *Nucleic Acids Research* 48.D1, pages D579-D589 (cf. pages 69, 98, 151).

VAN DEN BELD, MJC et FAG REUBSAET (2012). "Differentiation between Shigella, enteroinvasive Escherichia coli (EIEC) and noninvasive Escherichia coli". In : *European journal of clinical microbiology & infectious diseases* 31.6, pages 899-904 (cf. page 59).

VAN DONGEN, Stijn Marinus (2000). "Graph clustering by flow simulation". Thèse de doctorat (cf. pages 42, 51).

VEESLER, David et Christian CAMBILLAU (2011). "A common evolutionary origin for tailed-bacteriophage functional modules and bacterial machineries". In : *Microbiology and Molecular Biology Reviews* 75.3, pages 423-433 (cf. page 33).

VERNIKOS, Georgios S et Julian PARKHILL (2006). "Interpolated variable order motifs for identification of horizontally acquired DNA : revisiting the Salmonella pathogenicity islands". In : *Bioinformatics* 22.18, pages 2196-2203 (cf. page 75).

— (2008). "Resolving the structural features of genomic islands : a machine learning approach". In : *Genome research* 18.2, pages 331-342 (cf. page 75).

VIEIRA, Gilles et al. (2011). "Core and panmetabolism in Escherichia coli". In : *Journal of bacteriology* 193.6, pages 1461-1472 (cf. page 103).

VIEIRA-SILVA, Sara et Eduardo PC ROCHA (2010). "The systemic imprint of growth and its uses in ecological (meta) genomics". In : *PLoS genetics* 6.1, e1000808 (cf. page 24).

VON MERING, Christian et al. (2002). "Comparative assessment of large-scale data sets of protein–protein interactions". In : *Nature* 417.6887, pages 399-403 (cf. page 103).

VON MERING, Christian et al. (2003). "STRING : a database of predicted functional associations between proteins". In : *Nucleic acids research* 31.1, pages 258-261 (cf. page 96).

VON MERING, Christian et al. (2005). "STRING : known and predicted protein–protein associations, integrated and transferred across organisms". In : *Nucleic acids research* 33.suppl_1, pages D433-D437 (cf. page 96).

VON MERING, Christian et al. (2007). "STRING 7—recent developments in the integration and prediction of protein interactions". In : *Nucleic acids research* 35.suppl_1, pages D358-D362 (cf. page 96).

WAACK, Stephan et al. (2006). "Score-based prediction of genomic islands in prokaryotic genomes using hidden Markov models". In : *BMC bioinformatics* 7.1, pages 1-12 (cf. page 75).

WATSON, Hl C et JC KENDREW (1961). "Comparison between the amino-acid sequences of sperm whale myoglobin and of human haemoglobin". In : *Nature* 190.4777, pages 670-672 (cf. page 43).

WAYNE, LG et al. (1987). "Report of the ad hoc committee on reconciliation of approaches to bacterial systematics". In : *International Journal of Systematic and Evolutionary Microbiology* 37.4, pages 463-464 (cf. pages 55, 57).

WEI, Wen et al. (2017). "Zisland Explorer : detect genomic islands by combining homogeneity and heterogeneity properties". In : *Briefings in bioinformatics* 18.3, pages 357-366 (cf. pages 75, 76).

WEST, Stuart A et Guy A COOPER (2016). "Division of labour in microorganisms : an evolutionary perspective". In : *Nature Reviews Microbiology* 14.11, pages 716-723 (cf. page 35).

WESTOVER, Benjamin P et al. (2005). "Operon prediction without a training set". In : *Bioinformatics* 21.7, pages 880-888 (cf. page 90).

WHELAN, Fiona Jane, Martin RUSILOWICZ et James Oscar MCINERNEY (2020). "Coinfinder : detecting significant associations and dissociations in pangenomes". In : *Microbial genomics* 6.3 (cf. pages 94, 191).

WILLENBROCK, Hanni et al. (2007). "Characterization of probiotic Escherichia coli isolates with a novel pan-genome microarray". In : *Genome biology* 8.12, pages 1-16 (cf. pages 65, 107).

WINTER, Sascha et al. (2016). "Finding approximate gene clusters with Gecko 3". In : *Nucleic acids research* 44.20, pages 9600-9610 (cf. page 99).

WINTHER, Rasmus G (2001). "Varieties of modules : kinds, levels, origins, and behaviors". In : *Journal of Experimental Zoology* 291.2, pages 116-129 (cf. page 87).

WOESE, Carl R et George E FOX (1977). "Phylogenetic structure of the prokaryotic domain : the primary kingdoms". In : *Proceedings of the National Academy of Sciences* 74.11, pages 5088-5090 (cf. page 56).

WOESE, Carl R. (2005). "Q A". In : *Current Biology* 15.4, R111-R112. ISSN : 0960-9822. DOI : https://doi.org/10.1016/j.cub.2005.02.003. URL : https://www.sciencedirect.com/science/article/pii/S0960982205001417 (cf. page 10).

WOLF, Yuri I et Eugene V KOONIN (2012). "A tight link between orthologs and bidirectional best hits in bacterial and archaeal genomes". In : *Genome biology and evolution* 4.12, pages 1286-1294 (cf. page 47).

YANAI, Itai, Adnan DERTI et Charles DELISI (2001). "Genes linked by fusion events are generally of the same functional category : a systematic analysis of

30 microbial genomes". In : *Proceedings of the National Academy of Sciences* 98.14, pages 7940-7945 (cf. page 90).

Yu, Sung-Huan, Jörg Vogel et Konrad U Förstner (2018). "ANNOgesic : a Swiss army knife for the RNA-seq based annotation of bacterial/archaeal genomes". In : *GigaScience* 7.9, giy096 (cf. page 90).

Zeng, Xinghuo et al. (2008). "OrthoCluster : a new tool for mining synteny blocks and applications in comparative genomics". In : *Proceedings of the 11th international conference on Extending database technology : Advances in database technology*, pages 656-667 (cf. page 97).

Zhao, XiaoFei (2019). "BinDash, software for fast genome distance estimation on a typical personal laptop". In : *Bioinformatics* 35.4, pages 671-673 (cf. page 57).

Zhao, Yongbing et al. (2012). "PGAP : pan-genomes analysis pipeline". In : *Bioinformatics* 28.3, pages 416-418 (cf. page 68).

Zhu, Dekang et al. (2019). "Comparative analysis reveals the Genomic Islands in Pasteurella multocida population genetics : on Symbiosis and adaptability". In : *BMC genomics* 20.1, pages 1-11 (cf. page 82).

université PARIS-SACLAY

ÉCOLE DOCTORALE
Structure et dynamique des systèmes vivants (SDSV)

Titre : Méthodes d'analyse comparative de la variabilité intraspécifique des pangénomes procaryotes
Mots clés : Bioinformatique, Pangénomique, Théorie des graphes, Génomique microbienne

Résumé : Ces dernières années ont vu l'émergence de technologies de séquençage permettant l'acquisition massive de données génomiques de millions de procaryotes, majoritairement des bactéries. Loin de ralentir ou de se stabiliser, la quantité de données génomiques récupérées chaque année est toujours plus importante.

Une approche clé pour la compréhension des génomes se fait au travers de leur comparaison : les différences comme les similitudes dans les phénotypes d'organismes étudiés peuvent se retrouver aussi dans leurs génomes. Néanmoins, étudier ainsi des milliers, voire des millions de génomes, est informatiquement coûteux car cela nécessite des milliards de comparaisons. Il faut donc développer de nouvelles méthodes, trouver de nouveaux paradigmes pour continuer d'analyser et de comprendre les organismes, leurs génomes et les écosystèmes dans lesquels ils sont retrouvés.

Pour faire face à ce défi, des méthodes dites de pangénomiques ont récemment émergé dans le but de compiler l'ensemble de la diversité génomique d'une espèce d'intérêt. À l'origine, les pangénomes ne permettaient de comparer que le contenu en gènes des génomes pour déterminer ce qui était commun et ce qui ne l'était pas. Néanmoins, l'information apportée par la présence (ou l'absence) d'un gène ne prend du sens que si elle est rapportée dans son contexte génomique, car une fonction dépend souvent d'un ensemble de gènes.

Le but de cette thèse a été de développer un modèle de graphe de pangénome pour pouvoir comparer à la fois les gènes et leur organisation dans des (dizaines de) milliers de génomes procaryotes. Ce modèle à permis de développer des méthodes utilisant ce graphe pour en extraire des informations clés. Ces méthodes concernent notamment la détection d'îlots génomiques qui sont dépositaires de la majorité des acquisitions de nouvelles capacités phénotypiques chez les procaryotes. Elles incluent également l'identification de modules conservés qui sont des groupes de gènes qui fonctionnent potentiellement ensemble pour apporter une même fonction. Toutes ces méthodes ont été regroupées dans une suite logicielle appelée PPanGGOLiN, qui permet ainsi de décrire la diversité génomique d'une espèce et d'étudier la dynamique de l'acquisition et des échanges de gènes entre ces organismes.

Title : Methods for comparative analysis of the intraspecific variability in prokaryotic pangenomes
Keywords : Bioinformatics, Pangenomics, Graph theory, Microbial genomics

Abstract : The last few years have seen the emergence of sequencing technologies allowing the massive acquisition of genomic data from millions of prokaryotes, mostly bacteria. Far from slowing down or stabilizing, the amount of genomic data recovered each year is continuously increasing.

A key approach to understanding genomes is through their comparison : differences as well as similarities in the phenotypes of the organisms to be studied can also be found in their genomes. However, studying thousands or even millions of genomes in this way is computationally expensive, as it requires billions of comparisons. It is therefore necessary to develop new methods, to find new paradigms to continue to analyze and understand the organisms, their genomes, and the ecosystems in which they are found.

To face this challenge, so-called pangenomic methods have recently emerged with the aim of compiling the entire genomic diversity of a species of interest. Originally, pangenomes only allowed comparison of the gene content of genomes to determine what was common and what was not. However, the information provided by the presence (or absence) of a gene is only meaningful if reported in its genomic context, as a function often depends on a set of genes.

The goal of this thesis was to develop a pangenome graph model to compare both genes and their organization in (tens of) thousands of prokaryotic genomes. This model has allowed the development of methods using this graph to extract key information. These methods include the detection of genomic islands that are the repository of most of the acquisition of new phenotypic abilities in prokaryotes. They also include the identification of conserved modules, which are groups of genes that potentially work together to provide the same function. All these methods have been grouped in a software suite called PPanGGOLiN, which allows to describe the genomic diversity of a species and to study the dynamics of gene acquisition and exchange between these organisms.

Université Paris-Saclay
Espace Technologique / Immeuble Discovery
Route de l'Orme aux Merisiers RD 128 / 91190 Saint-Aubin, France