Multi-Proteins Similarity-based sampling to select representative genomes from large databases

Rémi-Vinh Coudert[1,2,*], Jean-Philippe Charrier[2], Frédéric Jauffrit[2], Jean-Pierre Flandrois[1,*], Céline Brochier-Armanet[1,*]

[1] Université de Lyon, Université Lyon 1, CNRS, VetAgro Sup, Laboratoire de Biométrie et Biologie Evolutive, UMR5558, Villeurbanne, France

[2] Microbiology Research & Development, bioMérieux SA, 376 Chemin de l'Orme, 69280 Marcy-l'Étoile, France

*To whom correspondence should be addressed.

Abstract

Motivation: Genome sequence databases are growing exponentially, but with high redundancy and uneven data quality. For these reasons, selecting representative subsets of ~~available~~ genomes is an essential step for almost all studies. However, most current sampling approaches are biased and ~~often~~ unable to process large datasets in a reasonable time.

Results: Here we present MPS-Sampling (Multiple-Protein Similarity-based Sampling), a fast, scalable, and efficient method for selecting reliable representative samples of genomes from very large datasets. Using ~~single-copy protein families~~families of homologous proteins as input, MPS-Sampling delineates homogeneous groups of genomes through two successive clustering steps. Representative genomes are then selected within these groups according to predefined or user-defined priority criteria.

MPS-Sampling was applied to a ~~set~~ dataset of 48 ribosomal protein families from 178,203 bacterial genomes to generate representative genome ~~samples~~ sets of various size, corresponding to a sampling of 32.17% down to 0.3% of the complete dataset. An in-depth analysis shows that the selected genomes are both taxonomically and phylogenetically representative of the complete dataset, demonstrating the relevance of the approach.

Availability and Implementation: MPS-Sampling is an open-source software available under CeCILL v2.1 at https://github.com/rvcoudert/MPS_Sampling.

Contact: remi.coudert@univ-lyon1.fr, celine.brochier-armanet@univ lyon1.fr

Supplementary information: Supplementary Materials are available at Bioinformatics online.

Introduction

The burst of genome sequencing provides a wealth of data and an ever-increasing access to genetic information, including from uncultured organisms (Stephens *et al.*, 2015). However, available genomic data are growing in an unbalanced way, both in terms of quality, as most released genomes are in fact rough draft assemblies, and diversity, with the over-representation of a few taxa, reflecting socio-economic considerations (Land *et al.*, 2015) (Supplementary Material S1-S2). In this context, exhaustive analyses are complex, time-consuming, technically impossible and, in most cases, irrelevant. Therefore, most studies use subsamples of available genomes (Land *et al.*, 2015). Current sampling strategies are usually based on taxonomic, phylogenic, or genome similarity criteria, but they all have the same objective: finding the best balance between sample size and representativeness, and proceed in the same way: the grouping of genomes, then the selection of representatives at random (Garcia *et al.*, 2021) or according to *a priori* defined criteria (e.g., type strains, completeness) (Parks *et al.*, 2017; Chun *et al.*, 2018).

Taxonomy-based approaches group genomes according to their taxonomy (Chun *et al.*, 2018; Garcia *et al.*, 2021). Although easy to apply, these approaches have limitations. For instance, while they represent a significant part of the biodiversity (e.g. up to 60% in the study by Parks *et al.* (Parks *et al.*, 2020)), many genomes have incomplete or even no taxonomic assignment (Supplementary Material S2). Most of the time, these genomes are omitted, which can lead to important gaps. Furthermore, these approaches are sensitive to taxonomy errors (Chun *et al.*, 2018) and hampered by historic legacy (Lan and Reeves, 2002).

Phylogeny-based approaches seek to group genomes according to the information provided by a phylogenetic tree (e.g. tree topology, branch supports, patristic distances). Several tools have been developed, such as Treemmer (Menardo *et al.*, 2018), PhyCLIP (Han *et al.*, 2019), TreeCluster (Balaban *et al.*, 2019), or AncestralClust (Pipes and Nielsen, 2022). However, they often include manual curation steps, require taxonomic information, and cannot handle very large phylogenetic trees due to computational time and memory usage. Furthermore, the quality of trees decreases as the number and divergence of sequences and their divergence increase (Philippe *et al.*, 2017), which may impact on the relevance of the clustering.

Most genome similarity-based methods are based on the computation of Overall Genome Relatedness Indexes (OGRI) (Chun and Rainey, 2014), such as the average nucleotide identity (ANI) (Goris *et al.*, 2007) or the shared ratio of k-mers (Ondov *et al.*, 2016). While being in principle taxonomically and phylogenetically agnostic, these approaches have also some limits. First, they are constrained by the quadratic complexity as they require genome pairwise comparisons (Chun and Rainey, 2014). Second, the accuracy of OGRIs strongly decreases as the evolutionary divergence between genomes increases (i.e. above the species or genus levels) (Qin *et al.*, 2014).

Finally, several international consortia provide their own sets of representative genomes (Supplementary Material S3). However, these ready-to-use datasets are of limited interest as users have no control over sampling density, redundancy, or data update, and in most cases the genomes of undescribed organisms are not included.

Overall, current approaches for selecting representative genomes have important limitations and are not well suited to handle large collections of genomes. In our view, an ideal approach should be (i) able to process very large datasets with acceptable computational time, (ii) independent of taxonomy and phylogeny, (iii) scalable, (iv) reproducible, and (v) allow users to define their own criteria for the selection of representative genomes.

To address these needs, we ~~have~~ developed MPS-Sampling (Multi Protein Similarity-based Sampling), a fast, scalable, and reliable method for selecting representative genomes. MPS-Sampling

Commenté [CBA1]: A voir si cette reference est utile

Commenté [CRV2R1]: Pour moi, elle est utile car elle parle de la complexité quadratique lors de l'analyse bio-informatique de génomes.

Voici la citation du texte : "For example, saving all available bacterial and archaeal genome sequences onto even a smartphone is possible (4 megabases 612 000 is only 40 gigabases). However, pairwise comparison of all genomes will take an unimaginably long time (if each comparison takes 1 min, 12 000 612 000 comparisons will take 273 years)."

Commenté [CBA3R1]: OK

uses families of homologous proteins as input and returns genome samples of variable density that are representative of both the taxonomic and phylogenetic diversity of the original datasets.

Materials and Methods

Materials and Methods

1. MPS-Sampling workflow

MPS-Sampling is distributed as a Snakemake pipeline (Supplementary Material S4). Its workflow is illustrated in Figure 1, as well as in a flowchart (Supplementary Material S5) and in an entity relationship diagram (ERD) (Supplementary Material S6).

Input.

MPS-Sampling uses families of homologous proteins as input (e.g. ribosomal families, core protein families). MPS-Sampling can handle missing data, meaning that protein families are not expected to be present in all genomes. The order of input data (i.e. genomes or protein families) does not affect the sampling results.

Step 1: Construction of Lin-clusters

This step aims at identifying pairs of closely related sequences. More precisely, within each protein family, sequences are clustered using Linclust of the MMseqs2 suite (Steinegger and Söding, 2018). Linclust was chosen for its efficiency, very high specificity, high sensitivity, and near-linear complexity which allows to efficiently process very large datasets. Linclust identifies putative pairs of sequences through a kmer-based heuristic. The relevance of each pair is then evaluated by Linclust based on sequence alignment using three parameters: the alignment e-value (eValue), coverage (minCov), and sequence identity (minSeqID). Linclust uses confirmed links to build sequence clusters, hereafter called **Lin-clusters**, according to the greedy set cover algorithm (Slavík, 1996). Lin-clusters are numbered from largest to smallest; the Lin-cluster encompassing the largest number Lin-clusters are numbered from largest to smallest; the Lin-cluster encompassing the largest number of sequences being designated as 1 (step 1-2).

Step 2: Construction of elementary groups of genomes

This first clustering aims to gather close genomes into elementary groups of genomes (EGG). Genomes are assigned a list of Lin-clusters, based on those to which their sequences belong. These assignments constitute the labels of the genomes, which are stored in the **Lin-clustering matrix** (step 2-1). To ensure reproducibility, the Lin-clustering matrix is reordered by columns (protein families) and by rows (genomes) (step 2-2). Genomes with the same label are then grouped together in the same EGG (step 2-2). The Lin-clustering matrix is reduced by removing identical lines, leading to a smaller and non-redundant matrix of labels, called the **Lin-combination matrix** (step 2-3). Each line in the Lin-combination matrix corresponds to one EGG, containing at least one genome. This delineation is very strict, as genomes that differ by only one Lin-cluster are placed in separate EGGs. Within a given EGG, genomes are considered indistinguishable from this stage.

Step 3: Construction of pre-connected components (optional)

This optional step aims at reducing the computation time by pre-defining rough groups of EGGs, called **pre-connected components**, prior the final clustering ().

Commenté [CBA4]: Compte-tenu de la remarque du reviewer 2 qui dit que ce qu'on propose n'est pas original et que tout repose sur Linclust, je me demande si on n'aurait pas intérêt à prendre de la distance avec Linclust et a appeler les Lin-cluster autrement . A discuter de vive voix.

Commenté [CRV5R4]: À discuter.

Commenté [CBA6R4]: OK on voit ça après l'envoi de ton manuscrit de thèse

Commenté [CBA7]: Inutile?

Commenté [CBA8]: J'ai l'impression que la figure n'a pas été mise à jour en fonction des remarques faites sur la version précédente

Commenté [CRV9R8]: Si c'est bon, la pré-connection est bien présente.

a mis en forme : Couleur de police : Automatique

Commenté [CBA10]: Dans la figure, il faudrait indiquer les EEG. On fera ça après le dépôt de ton manuscrit de thèse

Commenté [CRV11R10]: On les avait enlevés de la Figure car ça faisait trop lourd.

Commenté [CBA12]: On peut supprimer le schéma correspondant à cette étape dans la Figure 1, en indiquant simplement Step 3
Pre-connexion (optionnal)

Commenté [CRV13R12]: Ok, j'ajouterai ça dans la Figure 1.

Commenté [CBA14R12]: Il faut le faire

Commenté [CRV15R12]: Oui justement, la pré-connection a été ajoutée.

Step 4: Construction of MPS-clusters

This second clustering aims at gathering close EGGs into groups, called **MPS-clusters**. The similarity between two EGG is measured by the **Dice index** (Dice, 1945) (Supplementary Material S8 ~~Supplementary Material S5~~). Dice indexes are computed between all pairs of EGGs (or between pairs of EGGs within pre-connected ~~components) and~~ components) and stored in the **similarity matrix** (or into the similarity sub-matrix of each pre-connected component) (step 4-1). To avoid quadratic complexity, the matrix is calculated per column (Supplementary Material S9 ~~Supplementary Material S6 - 710~~). Then, EGGs whose the pairs of EGGs with a Dice index is lower than a threshold, called **minimum similarity** (Δ ~~Δ~~), are grouped according to an aggregative hierarchical method with complete linkage (step 4-2, ~~Supplementary Material S8~~). Δ varies from 0 to 1. Setting Δ to 1 is equivalent to bypassing this ~~second~~ step of clustering and, in this case, MPS-clusters correspond to ~~the~~ EGGs.

Step 5: Selection of MPS-representatives

This step consists in the selection of one representative genome, called **MPS-representative**, within each MPS-cluster. The choice of the MPS-representatives is based on a priority score (Supplementary Material S10), that encompasses, in the following order: user-defined criteria (e.g. type strain), completeness (i.e. genomes with the fewest missing values), and centrality according to the Dice index (Supplementary Material S11).

Output: List of MPS-representatives

MPS-Sampling returns the list of MPS-representative genomes, as well as all intermediate results (e.g. the Lin-clustering matrix, the Lin-combination matrix, the similarity matrix, the link between each input genome and its MPS-representative).

2. Datasets

MPS-Sampling was tested on the bacterial sequences of RiboDB v15.0 (Feb 2023), a dedicated database gathering ribosomal proteins (***r*-prots**) (Jauffrit *et al.*, 2016). This dataset, referred as to the bacterial dataset, encompasses 8,315,939 sequences spread over 48 r-prot families and 178,203 genomes (Supplementary Material S12-16). More precisely, 157,405 (88%) genomes are from RefSeq, among which 16,135 (9%) are labeled as RefSeq-representatives by the NCBI, while other genomes are from Genbank. According to the NCBI, 135,315 genomes (76%) have complete taxonomic information, meaning that each relevant taxonomic level (i.e. phylum, class, order, family, genus, and species) is defined. MPS-Sampling was also tested on the data of the GTDB v214.1 of June 9th 2023, encompassing 120 core protein families present in 394,932 bacterial genomes (Parks *et al.*, 2022). To test MPS-sampling on a larger dataset, we simulated an artificial bacterial dataset (ABD) of 534,609 genomes by applying artificial horizontal gene transfers to r-prot families of the bacterial dataset (see Supplementary Material S13).

3. MPS-Sampling parameters

MPS-Sampling parameters were optimized according to a standardized process (Supplementary Material S14-S24) and set as follow: eValue = 10^{-5}, coverageMode = 0, minCov = 0.8, minSeqID = 0.6, and elven values of $\Delta \in \{1, 0.9, 0.8, 0.7, 0.6, 0.5, 0.4, 0.3, 0.2, 0.1, 0.05\}$.

4. Taxonomic and phylogenetic diversity of the samples

For each sample, a phylogenetic tree was inferred (Supplementary Material S22). The taxonomic diversity is measured by the proportion of phyla, classes, orders, families, genera, and species retained in the sample, the taxonomic redundancy by the average number of genomes kept per taxon, and the phylogenetic diversity by the length of the tree (i.e. the sum of all branch lengths) divided by the number of tips. Phylogenetic tree figures were drown using iTOL (Letunic and Bork, 2021).

Commenté [CBA16]: Dans la figure c'est indiqué similarity sub-matrix

Commenté [CRV17R16]: Les sous-matrices font partie de la matrice. Donc stocker ça dans la matrice revient à stocker ça dans les sous-matrices, dans le cas où on utilise la pré-connection.

Code de champ modifié

a mis en forme : Police :Gras

a mis en forme : Police :Gras

Commenté [CBA18]: Détailler en suppl mat

Commenté [CRV19R18]: Ok

Commenté [CBA20R18]: Faire le bon renvoi

Commenté [CBA21]: Il me semblait que tu étais descendu à 1

Commenté [CRV22R21]: Si tu parles de la taille des échantillons, je suis en effet descendu jusqu'à 1, ce qui correspond à delta = 0. Mais je me suis arrêté à 0.05 à ta demande.

Commenté [CBA23]: Cette section doit dire comment on a estimé la diversité taxonomique et phylogénétique des échantillons. Pour tout ce qui est reconstruction d'arbre, y compris la visualisation avec iToL on renvoie au suppl. mat. Dans la version actuelle du papier, on explique comment on estime la diversité phylogénétique dans le texte. Ca n'est pas bon.

Commenté [CRV24R23]: Cette section dit justement "comment on a estimé la diversité taxonomique et phylogénétique des échantillons.". Je ne vois pas où est le problème...

5. Comparison with benchmark

MPS-Sampling was compared with two other tools: Treemmer and TaxSampler. Treemmer allow to define groups of genomes according to phylogenetic criteria (Menardo *et al.*, 2018), while TaxSampler is a homemade program allowing to sample genomes at a given taxonomic level (Supplementary Material S23).

Commenté [CBA25]: Il faut quand même expliquer comment ça fonctionne. Ill faudra faire un suppl mat pour la version qui sera soumise.

Commenté [CRV26R25]: Ok

Results

Computation time

MPS-sampling tests were done on a single-threaded process on AmalphyLab, a server with 32 cores and 64 threads (AMD EPYC 7542 32 Core Processor @3,40Ghz) and 1 To of DDR4 under Debian Trixie (Supplementary Material S24-28A). MPS-Sampling computation time depends mainly on the size of the dataset (i.e. number of genomes, protein families). Without pre-connection, it required 1h to generate one sample from the bacterial dataset (178,203 genomes, 48 protein families). However, the eleven runs only required 1h41 because steps 1 to 4-1 are common to all samples and executed only once. As expected, the use of pre-connection significantly decreased computation time, from 1h to 17 min (Supplementary Material S25B). In contrast, MPS-Sampling (without pre-connection) is 287 times faster than Treemmer, which required 360h to generate one sample from the bacterial dataset and 1,992h time to generate eleven samples (Supplementary Material S25C). For the ADB dataset (534,609 genomes, 48 protein families), generating one sample required 10h55 (Supplementary Material S25D), while it required 19h17 for the GTB dataset (394,932 genomes, 120 protein families, not shown).

Taxonomic and phylogenetic relevance of MPS-Sampling samples

All results are detailed in Supplementary Material S26-S37. As expected, the lower the value of Δ, the lower the sampling density, with, for example, the selection of 57,332 MPS-representatives (32.17% of the 178,203 genomes) when $\Delta = 1$, 3,474 (1.95%) when $\Delta = 0.4$ and 527 (0.30%) when $\Delta = 0.05$ (Figure 2A). Mapping the sampled genomes onto a reference bacterial phylogeny showed that MPS-representatives are well distributed across the tree, even when the number of retained genomes is very low (0.30%, $\Delta = 0.05$) (Supplementary Material S35-39).

The sampling relevance was assessed by monitoring the phylogenetic and taxonomic diversity of the samples throughout the dereplication process. The phylogenetic diversity increased linearly, from 0.0159 (the complete bacterial dataset) to 0.5638 ($\Delta = 0.05$) (Figure 2B). This indicates that discarded genomes were indeed phylogenetically the most redundant and that MPS-Sampling is successful in capturing the phylogenetic diversity of the bacterial dataset. A closer look shows that the proportion of lineages with incomplete taxonomy increased in the samples (Figure 2C), which is consistent with the fact they represent a significant part of the bacteria diversity (Lewis *et al.*, 2020).

From a taxonomic point of view, reducing the sampling density led to a progressive reduction in the number of genomes within species ($\Delta \leq 1$), of species within genera ($\Delta \leq 0.9$), genera within families ($\Delta \leq 0.8$), families within orders ($\Delta \leq 0.6$), and eventually orders and classes within phyla ($\Delta \leq 0.4$) (Figure 2D and Supplementary Material S37A). For instance, in the bacterial dataset, species were represented by eight genomes in average. When $\Delta = 1$, around two-thirds of the genomes were eliminated, but almost all the species were kept, each being represented by a single genome in average, meaning that discarded genomes corresponded to redundant genomes within species. Similarly, even when sample sizes were very small ($\Delta \leq 0.2$), most classes and phyla were conserved, but represented by few genomes (most often a single genome). Similar trends were observed with the GDTB dataset, except that dereplication was less progressive, meaning that for a given Δ, the fraction of MPS-representatives is greater for the GTDB than for the bacterial dataset (Supplementary Material S38). This can be explained by the fact that the GTDB dataset contained twice as many protein families, some of which having faster evolutionary rates than r-prots, and thus more Lin-clusters and Lin-combinations.

Commenté [CBA27]: Il y a un gap, on passe de S28-37 ici à S42-43 juste après.

Commenté [CBA28]: Je ne dois pas avoir la bonne version du suppl mat car les figures mentionnées ne présentent que trois cercles : 0.7, 0.6, 0.5 et 0.4

Commenté [CRV29R28]: Tu avais dit que laisser cette Figure avec quatre cercles était suffisant.

Commenté [CBA30]: Remettre la reference de park ? et mettrre aussi celle-ci dans l'intro là où on cite Park ?

Commenté [CRV31R30]: Ok.

Commenté [CBA32]: More or less ?

Commenté [CRV33R32]: Less.
C'est donc bon.

Commenté [CBA34]: On rediscutera de ce point après le dépôt de ton manuscrit

Commenté [CRV35R34]: Ok

Commenté [CRV36R34]: A placer en discussion car c'est une analyse du résultat. On peut le mettre quand on parle des familles protéiques à utiliser.

Comparison of MPS-sampling, Treemmer, and TaxSampler

We compare the sampling between MPS-Sampling, Treemmer and TaxSampler (Supplementary Material S39). MPS-Sampling and Treemmer are able to supply samples of variable size; this is not possible with TaxSampler, which is completely rigid in this respect. For the same sample size, MPS-Sampling provides higher phylogenetic diversity than Treemmer or TaxSampler. Indeed, for the first sample ($\Delta = 1$), the phylogenetic diversity between MPS-Sampling and Treemmer is of the same order (0.0489 against 0.0442). But from $\Delta = 0.6$, it is twice as high with MPS-Sampling than with Treemmer. Regarding TaxSampler, sampling at the species level provides a phylogenetic diversity half that of the equivalent sample with MPS-Sampling (0.0640 against 0.1270), and the same holds for the sampling at the genus level (0.1668 compared with 0.3054) (Supplementary Material S39). MPS-Sampling outperforms Treemmer in capturing taxonomic diversity. For example, in the sample with $\Delta = 0.4$, the ratio of represented phyla is maintained at 97% with MPS-Sampling but falls to 49% with Treemmer. MPS-Sampling is better than both Treemmer and TaxSampler in reducing taxonomic redundancy. For example, also with $\Delta = 0.4$, MPS-Sampling eliminates two to three times more intra-family redundancy than Treemmer and TaxSampler (3, 9 and 6 genomes per family respectively) (Supplementary Material S40).

To go further, we performed ~~and~~ an in-depth analysis of the relevance of the samples produced by MPS-Sampling, Treemmer and TaxSampler for three major bacterial families: the *Lactobacillaceae* (Zheng *et al.*, 2020), the *Bacillaceae* (Gupta *et al.*, 2020; Patel and Gupta, 2020), and the *Enterobacteriaceae* (Adeolu *et al.*, 2016) which presented increasing levels of redundancy (Supplementary Material S41~~Supplementary Material S46~~-S4~~85~~). Sampled genomes were mapped on the phylogenies of these families (Figure 3~~Figure 3~~ and Supplementary Material S43~~Supplementary Material S51-47~~). The results showed that MPS-sampling provided more representative samples, unlike Treemmer or TaxSampler, which tended to oversample large taxa with low diversity.

Code de champ modifié

As an example, the genus *Lactiplantibacillus* (light green at the bottom-right, Figure 3) was correctly reduced to one representative in the four samples of MPS-Sampling. Moreover, the square has changed in the fourth crown, meaning that the representative was updated and optimized for the fourth sample. Treemmer did not succeed in dereplicating this genus, choosing 41, 22, 17 and 9 representatives within. Regarding TaxSampler, the species sampling reduced *Lactiplantibacillus* to 18 representatives and the genus sampling to 1 representative.

As another example, at the species level, the genus *Bacillus* (*Bacillaceae*), which is composed of many closely related species, was clearly oversampled with both TaxSampler and Treemmer (Figure 3). Conversely, because *Bacillus* is not monophyletic (Gupta *et al.*, 2020)(Sultanpuram and Mothe, 2016), sampling at the genus level will result in the omission of a large part of the *Bacillus* real diversity.

Lastly, the family *Enterobacteriaceae* provides another example, with 99% of the genomes (16 948 among 17 096) strongly redundant, collapsed in one triangle (Supplementary Material S44). With MPS-Sampling, 5%-7% of the samples were in this triangle, counterbalancing the high redundancy. The proportion was 92%-94% and 62%-82% for Treemmer and TaxSampler, respectively, meaning that the redundancy was not dereplicated at all.

These three examples showed that MPS-Sampling succeeded in adapting the sampling density to the level of redundancy in the data and provided samples that were more consistent with the phylogenetic diversity of the data.

Discussion

MPS-Sampling is a fast, scalable, reliable, sequence similarity-based method for selecting representative sets of genomes. The genome comparison is based on the sequence similarity of a set of protein families (e.g. r-prots, universal single-copy protein families). In this sense, MPS-sampling is a multilocus sequence analysis (MLSA). Through the two-steps of genome clustering and matrix calculation, MPS-Sampling is able to process large genomic datasets in an acceptable computational time. MPS-Sampling also includes pre-connection, an optional step, that significantly reduces computation time.

The development of MPS-Sampling was inspired by the work of Sørensen (1948), which uses the Dice index and the hierarchical clustering with complete-linkage (Sørensen, 1948). Here, the Dice index is used to calculate the similarity between EGGs. The Dice index has interesting properties, is fast to compute and has been used in many comparative studies, including for delineating genus boundary using conserved proteins (Qin *et al.*, 2014). By using hierarchical method with complete-linkage, the intrinsic diversity within each MPS-cluster is controlled: the Dice index between any pair of EGGs is necessarily greater than Δ. MPS-Sampling has therefore a perfect specificity and a very good sensitivity. Indeed, being highly constrained, complete-linkage cannot create false links but can miss some of them, meaning that there is a slight risk of over-sampling, especially for large groups, which was preferred to a possible loss of representativeness. A major issue with hierarchical clustering and complete-linkage is that it does not handle outliers (Aguinis *et al.*, 2013). Here, outliers are genomes that cannot be linked to any other genome and thus correspond to MPS-clusters containing a single genome. For the bacterial dataset, this concerned 27% (Δ = 1) to 69% (Δ = 0.05) of the MPS-clusters. This may seem high, but it must be kept in mind that it is impossible to determine whether these outliers are artefactual (e.g., genome sequencing or assembly error, contamination) or if they represent real but poorly studied fraction of the diversity (e.g., new lineages).

In this study, MPS-Sampling was applied to r-prots and core protein families of the GTDB. These conserved proteins are well suited for sequence comparison on large evolutionary scales (i.e. from species to phyla) (Yutin *et al.*, 2012; Ramulu *et al.*, 2014; Parks *et al.*, 2022). This has been confirmed by the reliability of MPS sampling at both large and small evolutionary scales (*Bacteria*, *Lactobacillaceae*, *Bacillaceae*, *Enterobacteriaceae*). However, at smaller evolutionary scales (e.g., intra-species), using nucleotide sequences or faster evolving protein families would be more appropriate. MPS-Sampling is sensitive to the quality of the data used, as are all algorithms. However, MPS-Sampling is relatively robust to protein family errors and missing data (e.g. gene loss, incomplete genome sequencing) because it encapsulates information from many protein families. For instance, although absences are considered when constructing Lin-clusters and Lin-combinations, allowing genomes to be correctly distinguished from each other, they are not considered in the calculation of the Dice index and do not affect the delineation of the MPS-clusters. As a consequence, MPS-Sampling can be used with non-core protein families. MPS-Sampling is also relatively resistant to horizontal gene transfer (HGT), genome chimerism, and protein family assembly errors because the signal carried by the xenologous and non-homologous sequences will be dominated by the signal carried by the other proteins, allowing the corresponding genomes to be correctly linked. However, high levels of systematic error in protein family assembly or HGT can affect the sampling procedure. For example, if more than 30% of the proteins of genome A from taxon T have been acquired by HGT from a single donor, genome A will be correctively grouped with other genomes of T if $\Delta \leq 0.7$, but not if $\Delta \geq 0.7$. In this case, the genome A will be isolated from the other genomes to form a single MPS-cluster. Because of its chimeric nature, genome A is isolated from the other genomes and placed in its own MPS-cluster.

Commenté [CBA37]: Uses or introduces ?

Commenté [CRV38R37]: Sorensen utilise l'index de Dice introduit par Dice en 1945. En revanche, Sorensen est le premier à utiliser le *clustering* hiérarchique avec lien complet.

Commenté [CBA39]: On peut peut-être se contenter de renvoyer au suppl mat pour ne pas citer trop de de ref dans le main text.

Commenté [CRV40R39]: Ok

Commenté [CBA41]: Pourquoi ne pas donner le chiffre pour D = 1? A priori ça devrait être le chiffre le plus pertinent ?

Commenté [CBA42]: J'ai un petit soucis ici.
On s'attendrait à ce que la proportion de singleton diminue lorsque la densité de l'échantillonnage diminue.

Conclusion

MPS-Sampling is a new method for selecting samples of representative genomes from a huge database, based on Multi-Proteins Similarity (MPS). Our study shows that MPS-Sampling was particularly performant to reduce a large dataset of bacterial genomes, holding most of the evolutive diversity of the original set, both taxonomically and phylogenetically, and at various evolutionary scales. MPS-Sampling is still consistent when taxonomy is misleading and diverging from phylogeny.

Acknowledgments

The authors would like to thank Pierre S. Garcia and Najwa Taib for their help in providing feedback about MPS-Sampling.

Funding information

This work was supported by the National Association of Research and Technology (ANRT – Association Nationale de la Recherche et de la Technologie, France) (grant CIFRE N°2019/1231) and bioMérieux S.A., Marcy l'Étoile, France.

References

Adeolu,M. *et al.* (2016) Genome-based phylogeny and taxonomy of the 'Enterobacteriales': Proposal for enterobacterales ord. nov. divided into the families Enterobacteriaceae, Erwiniaceae fam. nov., Pectobacteriaceae fam. nov., Yersiniaceae fam. nov., Hafniaceae fam. nov., Morgane. *Int. J. Syst. Evol. Microbiol.*, **66**, 5575–5599.

Aguinis,H. *et al.* (2013) Best-Practice Recommendations for Defining, Identifying, and Handling Outliers. *Organ. Res. Methods*, **16**, 270–301.

Anaconda Documentation (2020) Anaconda Software Distribution.

Balaban,M. *et al.* (2019) TreeCluster: Clustering biological sequences using phylogenetic trees. *PLoS One*, **14**, 1–20.

Bateman,A. *et al.* (2017) UniProt: The universal protein knowledgebase. *Nucleic Acids Res.*, **45**, D158–D169.

Chun,J. *et al.* (2018) Proposed minimal standards for the use of genome data for the taxonomy of prokaryotes. *Int. J. Syst. Evol. Microbiol.*, **68**, 461–466.

Chun,J. and Rainey,F.A. (2014) Integrating genomics into the taxonomy and systematics of the Bacteria and Archaea. *Int. J. Syst. Evol. Microbiol.*, **64**, 316–324.

Dice,L.R. (1945) Measures of the Amount of Ecologic Association Between Species Author (s): Lee R . Dice Published by : Ecological Society of America Stable URL : http://www.jstor.org/stable/1932409. *Ecology*, **26**, 297–302.

Garcia,P.S. *et al.* (2021) A Comprehensive Evolutionary Scenario of Cell Division and Associated Processes in the Firmicutes. *Mol. Biol. Evol.*, **38**, 2396–2412.

Goris,J. *et al.* (2007) DNA-DNA hybridization values and their relationship to whole-genome sequence similarities. *Int. J. Syst. Evol. Microbiol.*, **57**, 81–91.

Gupta,R.S. *et al.* (2020) Robust demarcation of 17 distinct bacillus species clades, proposed as novel bacillaceae genera, by phylogenomics and comparative genomic analyses: Description of robertmurraya kyonggiensis sp. nov. and proposal for an emended genus bacillus limiting it o. *Int. J. Syst. Evol. Microbiol.*, **70**, 5753–5798.

Han,A.X. *et al.* (2019) Phylogenetic clustering by linear integer programming (PhyCLiP). *Mol. Biol. Evol.*, **36**, 1580–1595.

Harris,D. and Harris,S. (2012) Digital Design and Computer Architecture Kaufmann,M. (ed).

Jauffrit,F. *et al.* (2016) RiboDB Database: A Comprehensive Resource for Prokaryotic Systematics. *Mol. Biol. Evol.*, **33**, 2170–2172.

Köster,J. and Rahmann,S. (2012) Snakemake-a scalable bioinformatics workflow engine. *Bioinformatics*, **28**, 2520–2522.

Lan,R. and Reeves,P.R. (2002) Escherichia coli in disguise: Molecular origins of Shigella. *Microbes Infect.*, **4**, 1125–1132.

Land,M. *et al.* (2015) Insights from 20 years of bacterial genome sequencing. *Funct. Integr. Genomics*, **15**, 141–161.

Lassmann,T. (2020) Kalign 3: Multiple sequence alignment of large datasets. *Bioinformatics*, **36**, 1928–1929.

Letunic,I. and Bork,P. (2021) Interactive tree of life (iTOL) v5: An online tool for phylogenetic tree display and annotation. *Nucleic Acids Res.*, **49**, W293–W296.

Lewis,W.H. *et al.* (2020) Innovations to culturing the uncultured microbial majority. *Nat. Rev. Microbiol.*, **19**, 225–240.

Maayer,P. De *et al.* (2019) Reorganising the order Bacillales through phylogenomics. *Syst. Appl. Microbiol.*, **42**, 178–189.

Menardo,F. *et al.* (2018) Treemmer: A tool to reduce large phylogenetic datasets with minimal loss of diversity. *BMC Bioinformatics*, **19**, 1–8.

O'Leary,N.A. *et al.* (2016) Reference sequence (RefSeq) database at NCBI: Current status, taxonomic expansion, and functional annotation. *Nucleic Acids Res.*, **44**, D733–D745.

Ondov,B.D. *et al.* (2016) Mash: Fast genome and metagenome distance estimation using MinHash. *Genome Biol.*, **17**, 1–14.

Parks,D.H. *et al.* (2020) A complete domain-to-species taxonomy for Bacteria and Archaea. *Nat. Biotechnol.*, **38**, 1079–1086.

Parks,D.H. *et al.* (2022) GTDB: An ongoing census of bacterial and archaeal diversity through a phylogenetically consistent, rank normalized and complete genome-based taxonomy. *Nucleic Acids Res.*, **50**, D785–D794.

Parks,D.H. *et al.* (2017) Recovery of nearly 8,000 metagenome-assembled genomes substantially expands the tree of life. *Nat. Microbiol.*, **2**, 1533–1542.
Patel,S. and Gupta,R.S. (2020) A phylogenomic and comparative genomic framework for resolving the polyphyly of the genus bacillus: Proposal for six new genera of bacillus species, peribacillus gen. nov., cytobacillus gen. nov., mesobacillus gen. nov., neobacillus gen. nov., metabacillu. *Int. J. Syst. Evol. Microbiol.*, **70**, 406–438.
Philippe,H. *et al.* (2017) Pitfalls in supermatrix phylogenomics. *Eur. J. Taxon.*, **2017**, 1–25.
Pipes,L. and Nielsen,R. (2022) AncestralClust: clustering of divergent nucleotide sequences by ancestral sequence reconstruction using phylogenetic trees. *Bioinformatics*, **38**, 663–670.
Price,M.N. *et al.* (2010) FastTree 2 - Approximately maximum-likelihood trees for large alignments. *PLoS One*, **5**.
Qin,Q.L. *et al.* (2014) A proposed genus boundary for the prokaryotes based on genomic insights. *J. Bacteriol.*, **196**, 2210–2215.
Ramulu,H.G. *et al.* (2014) Ribosomal proteins: Toward a next generation standard for prokaryotic systematics? *Mol. Phylogenet. Evol.*, **75**, 103–117.
Slavík,P. (1996) A tight analysis of the greedy algorithm for set cover. *Proc. Annu. ACM Symp. Theory Comput.*, **Part F1294**, 435–441.
Song,I. *et al.* (1995) A Comparative Analysis of Entity-Relationship Diagrams. *J. Comput. Softw. Eng.*, **3**, 427–459.
Sørensen,T. (1948) A method of establishing groups of equal amplitude in plant sociology based on similarity of species content and its application to analyses of the vegetation on Danish commons.
Steinegger,M. and Söding,J. (2018) Clustering huge protein sequence sets in linear time. *Nat. Commun.*, **9**.
Steinegger,M. and Söding,J. (2017) MMseqs2 enables sensitive protein sequence searching for the analysis of massive data sets. *Nat. Biotechnol.*, **35**, 1026–1028.
Stephens,Z.D. *et al.* (2015) Big data: Astronomical or genomical? *PLoS Biol.*, **13**, 1–11.
Sultanpuram,V.R. and Mothe,T. (2016) Salipaludibacillus aurantiacus gen. Nov., sp. nov. a novel alkali tolerant bacterium, reclassification of Bacillus agaradhaerens as Salipaludibacillus agaradhaerens comb. nov. and Bacillus neizhouensis as Salipaludibacillus neizhouensis comb. nov. *Int. J. Syst. Evol. Microbiol.*, **66**, 2747–2753.
Yates,A.D. *et al.* (2022) Ensembl Genomes 2022: An expanding genome resource for non-vertebrates. *Nucleic Acids Res.*, **50**, D996–D1003.
Yutin,N. *et al.* (2012) Phylogenomics of Prokaryotic Ribosomal Proteins. *PLoS One*, **7**, e36972.
Zheng,J. *et al.* (2020) A taxonomic note on the genus Lactobacillus: Description of 23 novel genera, emended description of the genus Lactobacillus beijerinck 1901, and union of Lactobacillaceae and Leuconostocaceae. *Int. J. Syst. Evol. Microbiol.*, **70**, 2782–2858.

Figure 1 – Overview of the MPS-Sampling workflow

Input. MPS-Sampling uses families of homologous single-copy protein sequences as input. In this example, ten genomes ($\mathbf{g_A}$, $\mathbf{g_B}$, ..., $\mathbf{g_J}$) and four protein families (**uL1**, **uL2**, **uL3**, and **uL4**) are considered. Dashes indicate the absence of a sequence in a genome. Here, **uL2** is missing in $\mathbf{g_A}$, $\mathbf{g_B}$, $\mathbf{g_C}$, $\mathbf{g_I}$ and $\mathbf{g_J}$, and **uL4** is missing in $\mathbf{g_I}$.

Step 1-1: Construction of Lin-clusters. For each protein family, sequence clusters, called **Lin-clusters**, are built using Linclust (Steinegger and Söding, 2018) of the Mmseqs2 suite. Because Linclust is applied to each protein family, the clustering, and thus Lin-clusters, may differ from one protein family to another. Here, the **uL1** sequence of $\mathbf{g_A}$ is clustered with **uL1** sequences of $\mathbf{g_E}$, $\mathbf{g_F}$ and $\mathbf{g_G}$, while **uL4** sequences of $\mathbf{g_A}$ and $\mathbf{g_D}$ are grouped together.

Step 12-12: Labeling of protein sequences. For each protein family, protein sequences are labeled according to the Lin-cluster to which they belong. Sequences from the largest Lin-clusters are labeled first. For instance, **uL1** sequences from the largest Lin-cluster ($\mathbf{g_A}$, $\mathbf{g_E}$, $\mathbf{g_F}$, and $\mathbf{g_G}$) are labeled as **1**, sequences from the second largest Lin-cluster ($\mathbf{g_B}$, $\mathbf{g_C}$, and $\mathbf{g_D}$) as **2**, while the third largest Lin-cluster ($\mathbf{g_H}$, $\mathbf{g_I}$, and $\mathbf{g_J}$) is labeled as 3.

Step 2-12: Construction of the Lin-clustering matrix. Lin-cluster labels are stored in a matrix, called **Lin-clustering matrix**, whose rows correspond to genomes and columns to protein families. Here, genome $\mathbf{g_A}$ (first row) is labeled as **(1, 1, 1, 2)**.

Step 2-23: Re-ordering of the Lin-clustering matrix. To ensure reproducibility of the sampling, protein families (columns) and genomes (rows) are re-ordered. Protein families are ordered according to their number of Lin-clusters and to the hashing value of their name. Genomes are ordered according to the lexicographic order and to the hashing value of their name. Here, all the four protein families have the same number of Lin-clusters (3), thus are ordered according to their hashing value, giving the new order: **uL3, uL1, uL2, uL4**. Then, the genomes are ordered according first to the lexicographic order, then to the hashing value of their name, given the new order: $\mathbf{g_F}$, $\mathbf{g_F}$, $\mathbf{g_G}$, $\mathbf{g_A}$, $\mathbf{g_B}$, $\mathbf{g_C}$, $\mathbf{g_D}$, $\mathbf{g_H}$, $\mathbf{g_I}$, $\mathbf{g_J}$. Here the ordered of the triplet ($\mathbf{g_E}$, $\mathbf{g_F}$, $\mathbf{g_G}$) and the duet ($\mathbf{g_B}$, $\mathbf{g_C}$) is determined given the hashing value of the genome names.

Step 2-34: Construction of the Lin-combination matrix. Redundant lines of the Lin-matrix are fused, leading to a smaller matrix, called **Lin-combination matrix**, whose rows correspond to unique Lin-combinations of Lin-clusters. In this example, genomes $\mathbf{g_B}$ and $\mathbf{g_C}$ harbor the same Lin-combination **(1, 2, -, 1)** and are thus bound together in the same Lin-combination (**Comb3**). At this step, the absence of sequences is not taken into consideration; for example, the absence of **uL2** in $\mathbf{g_B}$ and $\mathbf{g_C}$ is not considered as a difference to separate these two genomes. The genomes grouped together into the same Lin-combination are called an elementary group of genomes (EGG), because they constitute a set of genomes indistinguishable according to the parameters used to the run.

Step 3: Construction of pre-connected components (optional). Lin-combinations are gathered into rough groups called **pre-connected components**. This step, called pre-connection, provides a rough and fast delineation while conserving all pairwise links above a given threshold. In the example, using minNbLinclusters = 2, two pre-connected components are built. A first pre-connected component gathers **Comb1** ($\mathbf{g_F}$, $\mathbf{g_E}$, $\mathbf{g_G}$), **Comb2** ($\mathbf{g_A}$), **Comb3** ($\mathbf{g_B}$, $\mathbf{g_C}$), and **Comb4** ($\mathbf{g_D}$), corresponding to the seven genomes ($\mathbf{g_F}$, $\mathbf{g_E}$, $\mathbf{g_G}$, $\mathbf{g_A}$, $\mathbf{g_B}$, $\mathbf{g_C}$, and $\mathbf{g_D}$), while the second pre-connected component gathers **Comb5** ($\mathbf{g_H}$), **Comb6** ($\mathbf{g_I}$), and **Comb7** ($\mathbf{g_J}$), encompassing the three genomes ($\mathbf{g_H}$, $\mathbf{g_I}$, and $\mathbf{g_J}$) (~~Supplementary Material S4~~).

Step 4-1: Computation of similarity submatrices. Within pre-connected components, the similarity between each pair of Lin-combinations is computed and stored in square submatrices, called **similarity submatrices**. The similarity is expressed by the Dice index which corresponds to the proportion of common Lin-clusters between two Lin-combinations (missing values are omitted) (). Here, **Comb6** and **Comb7** share two Lin-clusters out of five, so their Dice index in the matrix is **2/5**.

Step 4-2: Construction of MPS-clusters. The Lin-combinations (and the corresponding genomes) are clustered into **MPS-clusters** according to a hierarchical method with complete-linkage up to a minimum similarity Δ (~~Supplementary Material S8~~). In the example, five MPS-clusters are built using minimum similarity Δ = 0.5. A first MPS-cluster gathers two Lin-combinations: **Comb1** ($\mathbf{g_F}$, $\mathbf{g_E}$, $\mathbf{g_G}$) and **Comb2** ($\mathbf{g_A}$), corresponding to four genomes ($\mathbf{g_F}$, $\mathbf{g_E}$, $\mathbf{g_G}$, and $\mathbf{g_A}$). A second MPS-cluster gathers two Lin-combinations: **Comb5** ($\mathbf{g_H}$) and **Comb6** ($\mathbf{g_I}$), corresponding to the two genomes ($\mathbf{g_H}$ and $\mathbf{g_I}$). A third MPS-cluster encompasses only one Lin-combination **Comb3** ($\mathbf{g_B}$ and $\mathbf{g_C}$), but it corresponds to two genomes ($\mathbf{g_B}$ and $\mathbf{g_C}$). The two last genomes corresponding to **Comb4** ($\mathbf{g_D}$) and **Comb7** ($\mathbf{g_J}$) are isolated and correspond to singleton MPS-clusters.

Step 5: Selection of MPS-representatives. One **MPS-representative** genome is selected per MPS-cluster. These MPS-representatives are selected according to centrality, quality, and fame criteria. Here, $\mathbf{g_G}$, $\mathbf{g_C}$, $\mathbf{g_D}$, $\mathbf{g_H}$, and $\mathbf{g_J}$ are selected, each representing one MPS-cluster.

Output: MPS-Sampling returns the list of the MPS-representative genomes, as well as the links between each input genome and its MPS-representative genome.

Code de champ modifié

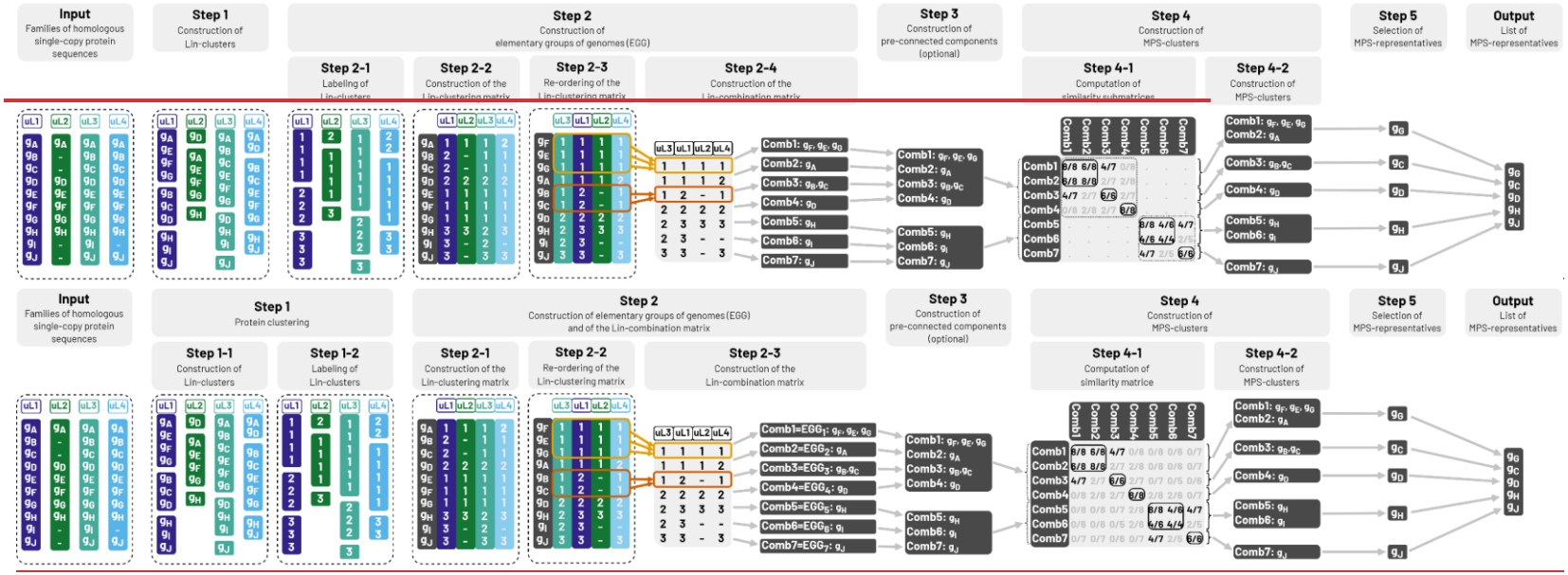

Input
Families of homologous single-copy protein sequences
Step 1
Construction of Lin-clusters
Step 2
Construction of elementary groups of genomes (EGG)
Step 2-1
Labeling of Lin-clusters
Step 2-2
Construction of the Lin-clustering matrix
Step 2-3
Re-ordering of the Lin-clustering matrix
Step 2-4
Construction of the Lin-combination matrix
Step 3
Construction of pre-connected components (optional)
Step 4
Construction of MPS-clusters
Step 4-1
Computation of similarity submatrices
Step 4-2
Construction of MPS-clusters
Step 5
Selection of MPS-representatives
Output
List of MPS-representatives
Input
Families of homologous single-copy protein sequences
Step 1
Protein clustering
Step 1-1
Construction of Lin-clusters
Step 1-2
Labeling of Lin-clusters
Step 2
Construction of elementary groups of genomes (EGG) and of the Lin-combination matrix
Step 2-1
Construction of the Lin-clustering matrix
Step 2-2
Re-ordering of the Lin-clustering matrix
Step 2-3
Construction of the Lin-combination matrix
Step 3
Construction of pre-connected components (optional)
Step 4
Construction of MPS-clusters
Step 4-1
Computation of similarity matrice
Step 4-2
Construction of MPS-clusters
Step 5
Selection of MPS-representatives
Output
List of MPS-representatives

Figure 2 – Sampling of the bacterial dataset (178,203 genomes) by MPS--Sampling.
A: Size of the samples built by MPS--Sampling.
B: Phylogenetic diversity of the samples, computed by the length of all branches of the tree inferred with the sample genomes divided by the number of leaves.
C: Genomic reduction of the 135,315 genomes and the 42,888 genomes with a complete and incomplete taxonomic affiliation, respectively.
D: Taxonomic diversity of the samples. The proportion of phyla, classes, orders, families, genera, and species represented in each sample is indicated.

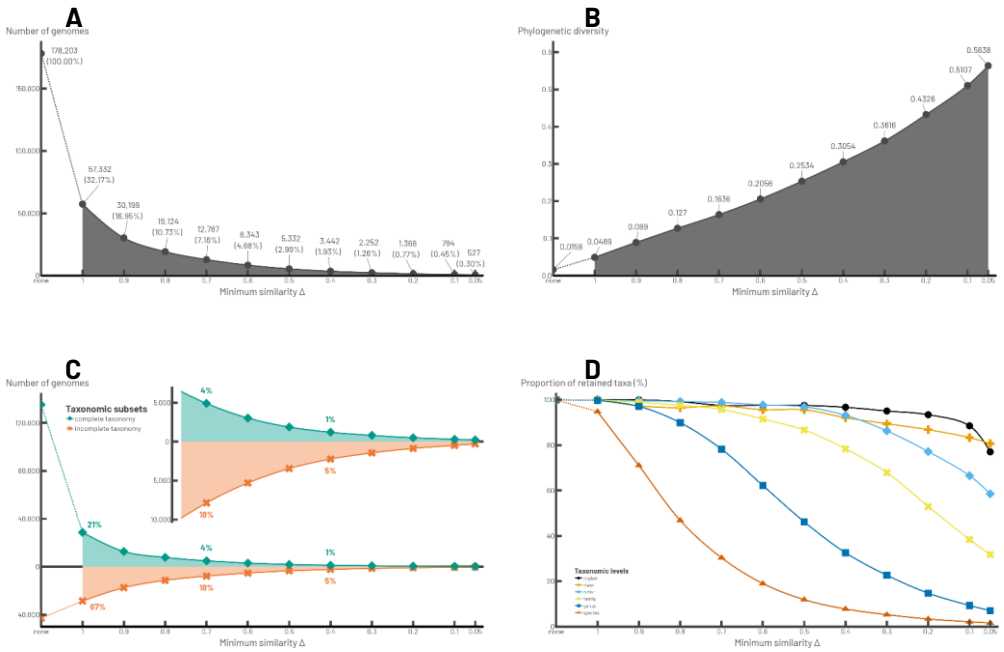


Figure 3 – Phylogenetic distribution on *Lactobacillaceae* and *Bacillaceae*
(legend on the next page)

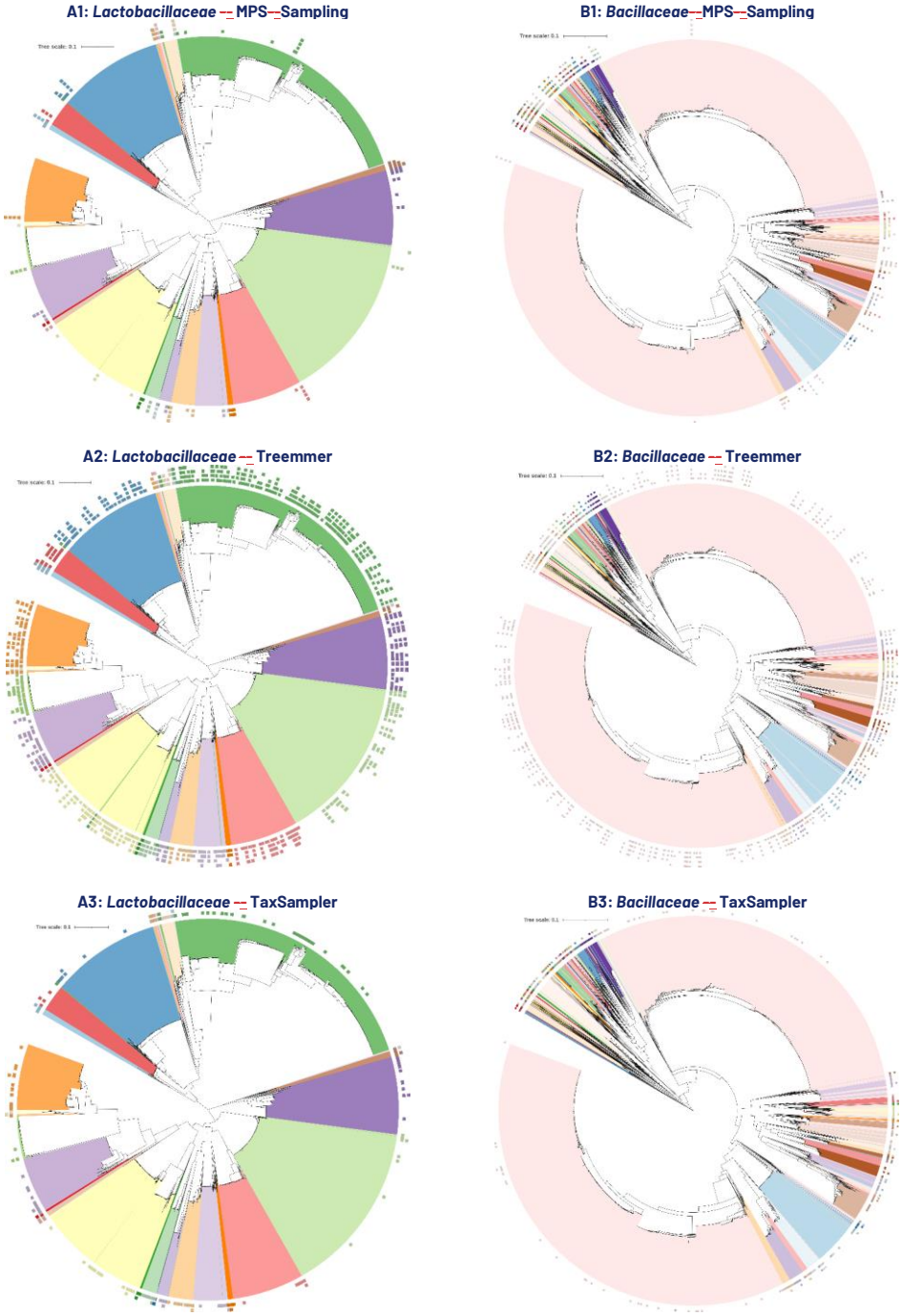


Figure 3 – Phylogenetic distribution on *Lactobacillaceae* and *Bacillaceae*

Sampling by the three methods (MPS-Sampling, Treemmer and TaxSampler) are compared for two taxonomic families (*Lactobacillaceae* and *Bacillaceae*).

Phylogenetic tree inferred from the 6,410 genomes of *Lactobacillaceae*. Leaves are colored according to the 33 genera.
A1: Four samples generated by MPS-Sampling are tagged around the phylogeny, for the respective values: $\Delta \in \{0.7; 0.6; 0.5; 0.4\}$, corresponding to 80, 44, 24 and 12 MPS-representatives, respectively.
A2: Four Treemmer-generated samples are tagged around the phylogeny, corresponding to 429, 276, 188 and 124 Treemmer-representatives respectively. The Treemmer samples studied are the same size as those from MPS-Sampling on Bacteria.
A3: Two samples generated by TaxSampler are tagged around the phylogeny, corresponding to 394 and 33 Tax-representatives respectively. The two TaxSampler samples correspond to both species and genus levels.

Phylogenetic tree inferred from the 7,113 genomes of *Bacillaceae*. Leaves are colored according to the 108 genera.
B1: Four samples generated by MPS-Sampling are tagged around the phylogeny, for the respective values: $\Delta \in \{0.7; 0.6; 0.5; 0.4\}$, corresponding to 247, 142, 74 and 38 MPS-representatives, respectively.
B2: Four Treemmer-generated samples are tagged around the phylogeny, corresponding to 405, 264, 186 and 114 Treemmer-representatives respectively. The Treemmer samples studied are the same size as those from MPS-Sampling on Bacteria.
B3: Two samples generated by TaxSampler are tagged around the phylogeny, corresponding to 693 and 108 Tax--representatives respectively. The two TaxSampler samples correspond to both species and genus levels.

Supplementary material

Multi-Proteins Similarity-based sampling to select representative genomes from large databases

Rémi-Vinh Coudert[1,2,*], Jean-Philippe Charrier[2], Frédéric Jauffrit[2], Jean-Pierre Flandrois[1], Céline Brochier-Armanet[1]

[1] Université de Lyon, Université Lyon 1, CNRS, VetAgro Sup, Laboratoire de Biométrie et Biologie Evolutive, UMR5558, Villeurbanne, France

[2] Microbiology Research & Development, bioMérieux SA, 376 Chemin de l'Orme, 69280 Marcy L'Etoile, France

*To whom correspondence should be addressed.

Table of contents

List of Figures

List of Supplementary Materials

Supplementary Material S1 – Growth of available genomic data over time

Adeolu,M. *et al.* (2016) Genome-based phylogeny and taxonomy of the 'Enterobacteriales': Proposal for enterobacterales ord. nov. divided into the families Enterobacteriaceae, Erwiniaceae fam. nov., Pectobacteriaceae fam. nov., Yersiniaceae fam. nov., Hafniaceae fam. nov., Morganc. *Int. J. Syst. Evol. Microbiol.*, **66**, 5575–5599.

Aguinis,H. *et al.* (2013) Best-Practice Recommendations for Defining, Identifying, and Handling Outliers. *Organ. Res. Methods*, **16**, 270–301.

Balaban,M. *et al.* (2019) TreeCluster: Clustering biological sequences using phylogenetic trees. *PLoS One*, **14**, 1–20.

Bateman,A. *et al.* (2017) UniProt: The universal protein knowledgebase. *Nucleic Acids Res.*, **45**, D158–D169.

Bursteinas,B. *et al.* (2016) Minimizing proteome redundancy in the UniProt Knowledgebase. *Database*, **2016**, 1–9.

Chun,J. *et al.* (2018) Proposed minimal standards for the use of genome data for the taxonomy of prokaryotes. *Int. J. Syst. Evol. Microbiol.*, **68**, 461–466.

Chun,J. and Rainey,F.A. (2014) Integrating genomics into the taxonomy and systematics of the Bacteria and Archaea. *Int. J. Syst. Evol. Microbiol.*, **64**, 316–324.

Dice,L.R. (1945) Measures of the Amount of Ecologic Association Between Species Author (s): Lee R . Dice Published by : Ecological Society of America Stable URL : http://www.jstor.org/stable/1932409. *Ecology*, **26**, 297–302.

Garcia,P.S. *et al.* (2021) A Comprehensive Evolutionary Scenario of Cell Division and Associated Processes in the Firmicutes. *Mol. Biol. Evol.*, **38**, 2396–2412.

Goris,J. *et al.* (2007) DNA–DNA hybridization values and their relationship to whole-genome sequence similarities. *Int. J. Syst. Evol. Microbiol.*, **57**, 81–91.

Gupta,R.S. *et al.* (2020) Robust demarcation of 17 distinct bacillus species clades, proposed as novel bacillaceae genera, by phylogenomics and comparative genomic analyses: Description of robertmurraya kyonggiensis sp. nov. and proposal for an emended genus bacillus limiting it o. *Int. J. Syst. Evol. Microbiol.*, **70**, 5753–5798.

Han,A.X. *et al.* (2019) Phylogenetic clustering by linear integer programming (PhyCLiP). *Mol. Biol. Evol.*, **36**, 1580–1595.

Harris,D. and Harris,S. (2012) Digital Design and Computer Architecture Kaufmann,M. (ed).

Jauffrit,F. *et al.* (2016) RiboDB Database: A Comprehensive Resource for Prokaryotic Systematics. *Mol. Biol. Evol.*, **33**, 2170–2172.

Köster,J. and Rahmann,S. (2012) Snakemake-a scalable bioinformatics workflow engine. *Bioinformatics*, **28**, 2520–2522.

Lan,R. and Reeves,P.R. (2002) Escherichia coli in disguise: Molecular origins of Shigella. *Microbes Infect.*, **4**, 1125–1132.

Land,M. *et al.* (2015) Insights from 20 years of bacterial genome sequencing. *Funct. Integr. Genomics*, **15**, 141–161.

Lassmann,T. (2020) Kalign 3: Multiple sequence alignment of large datasets. *Bioinformatics*, **36**, 1928–1929.

Letunic,I. and Bork,P. (2021) Interactive tree of life (iTOL) v5: An online tool for phylogenetic tree display and annotation. *Nucleic Acids Res.*, **49**, W293–W296.

Lewis,W.H. *et al.* (2020) Innovations to culturing the uncultured microbial majority. *Nat. Rev. Microbiol.*, **19**, 225–240.

Maayer,P. De *et al.* (2019) Reorganising the order Bacillales through phylogenomics. *Syst. Appl. Microbiol.*, **42**, 178–189.

Menardo,F. *et al.* (2018) Treemmer: A tool to reduce large phylogenetic datasets with minimal loss of diversity. *BMC Bioinformatics*, **19**, 1–8.

O'Leary,N.A. *et al.* (2016) Reference sequence (RefSeq) database at NCBI: Current status, taxonomic expansion, and functional annotation. *Nucleic Acids Res.*, **44**, D733–D745.

Ondov,B.D. *et al.* (2016) Mash: Fast genome and metagenome distance estimation using MinHash. *Genome Biol.*, **17**, 1–14.

Parks,D.H. *et al.* (2020) A complete domain-to-species taxonomy for Bacteria and Archaea. *Nat. Biotechnol.*, **38**, 1079–1086.

Parks,D.H. *et al.* (2022) GTDB: An ongoing census of bacterial and archaeal diversity through a

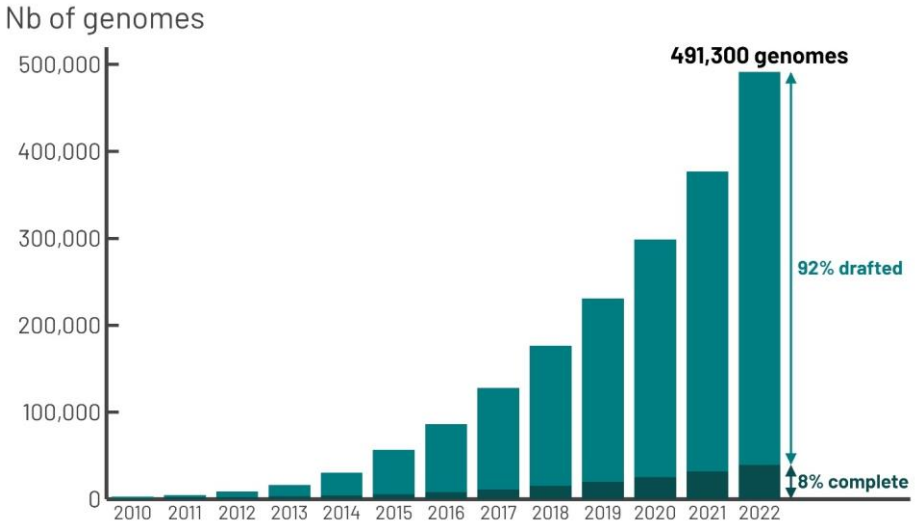
phylogenetically consistent, rank normalized and complete genome-based taxonomy. *Nucleic Acids Res.*, **50**, D785–D794.
Parks, D.H. *et al.* (2017) Recovery of nearly 8,000 metagenome-assembled genomes substantially expands the tree of life. *Nat. Microbiol.*, **2**, 1533–1542.
Patel, S. and Gupta, R.S. (2020) A phylogenomic and comparative genomic framework for resolving the polyphyly of the genus bacillus: Proposal for six new genera of bacillus species, peribacillus gen. nov., cytobacillus gen. nov., mesobacillus gen. nov., neobacillus gen. nov., metabacillu. *Int. J. Syst. Evol. Microbiol.*, **70**, 406–438.
Philippe, H. *et al.* (2017) Pitfalls in supermatrix phylogenomics. *Eur. J. Taxon.*, **2017**, 1–25.
Pipes, L. and Nielsen, R. (2022) AncestralClust: clustering of divergent nucleotide sequences by ancestral sequence reconstruction using phylogenetic trees. *Bioinformatics*, **38**, 663–670.
Price, M.N. *et al.* (2010) FastTree 2 – Approximately maximum-likelihood trees for large alignments. *PLoS One*, **5**.
Qin, Q.L. *et al.* (2014) A proposed genus boundary for the prokaryotes based on genomic insights. *J. Bacteriol.*, **196**, 2210–2215.
Ramulu, H.G. *et al.* (2014) Ribosomal proteins: Toward a next generation standard for prokaryotic systematics? *Mol. Phylogenet. Evol.*, **75**, 103–117.
Slavík, P. (1996) A tight analysis of the greedy algorithm for set cover. *Proc. Annu. ACM Symp. Theory Comput.*, **Part F1294**, 435–441.
Song, I. *et al.* (1995) A Comparative Analysis of Entity-Relationship Diagrams. *J. Comput. Softw. Eng.*, **3**, 427–459.
Sørensen, T. (1948) A method of establishing groups of equal amplitude in plant sociology based on similarity of species content and its application to analyses of the vegetation on Danish commons.
Steinegger, M. and Söding, J. (2018) Clustering huge protein sequence sets in linear time. *Nat. Commun.*, **9**.
Stephens, Z.D. *et al.* (2015) Big data: Astronomical or genomical? *PLoS Biol.*, **13**, 1–11.
Sultanpuram, V.R. and Mothe, T. (2016) Salipaludibacillus aurantiacus gen. Nov., sp. nov. a novel alkali tolerant bacterium, reclassification of Bacillus agaradhaerens as Salipaludibacillus agaradhaerens comb. nov. and Bacillus neizhouensis as Salipaludibacillus neizhouensis comb. nov. *Int. J. Syst. Evol. Microbiol.*, **66**, 2747–2753.
Yates, A.D. *et al.* (2022) Ensembl Genomes 2022: An expanding genome resource for non-vertebrates. *Nucleic Acids Res.*, **50**, D996–D1003.
Yutin, N. *et al.* (2012) Phylogenomics of Prokaryotic Ribosomal Proteins. *PLoS One*, **7**, e36972.
Zheng, J. *et al.* (2020) A taxonomic note on the genus Lactobacillus: Description of 23 novel genera, emended description of the genus Lactobacillus beijerinck 1901, and union of Lactobacillaceae and Leuconostocaceae. *Int. J. Syst. Evol. Microbiol.*, **70**, 2782–2858.

Supplementary Material S1 – Growth of available genomic data over time

From the NCBI website, the following file was downloaded:
ftp://ftp.ncbi.nlm.nih.gov/genomes/GENOME_REPORTS/prokaryotes.txt.

It represents the available genomic data of GenBank. All rows until and including 2022 were considered. A cumulated count by year was computed and showed in the bar plot below. Among them, the drafted genomes (i.e. scaffolds and contigs) were colored in light blue at the top of each bar and the complete genomes in dark blue.

The cumulated amount doubled every 2 or 3 years and reached 491,300 genomes at the end of 2022. At this time, only 8% of the data were complete genomes while the remaining 92% were only drafted genomes.

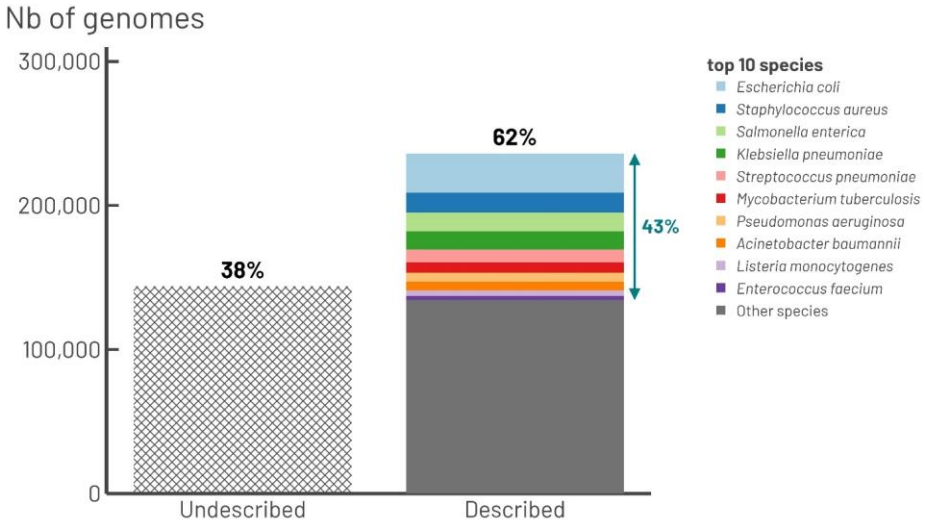


Supplementary Material S2 – Unbalanced representativity in available genomic data

From the NCBI website, the following file was downloaded:
ftp://ftp.ncbi.nlm.nih.gov/genomes/GENOME_REPORTS/prokaryotes.txt.

It represents the available genomic data of GenBank. All rows until and including 2022 were considered. Genomes whose species name ending with sp., bacterium, or archaeon, were classified as "undescribed". Remaining genomes were classified as "described". Among them, the genomes of the 10 most represented species were highlighted, as mentioned in the legend (*Escherichia coli*, *Staphylococcus aureus*…).

In 2022, 38% of the labelled genomes correspond to "undescribed" genomes and 62% are labelled as described genomes, among which 43%
represent only 10 species, the most sequenced ones.



Supplementary Material S3 – Available sets of representative genomes

Several consortia provide ready-to-use sets ~~have made libraries~~ of representative genomes ~~available to users~~. NCBI (O'Leary *et al.*, 2016), Uniprot (Bateman *et al.*, 2017), and Ensembl Genomes (Yates *et al.*, 2022) propose representative genomic data, which contain, as of 2023/01/27, 17,145 RefSeq--representative genomes (including 16,557 for *Bacteria*), 22,121 UniProt reference proteomes (including 8,821 for *Bacteria*), and 33,316 Ensembl--representative genomes (including 31,332 for *Bacteria*) respectively. RefSeq--representative genomes[1] ~~(O'Leary *et al.*, 2016)~~ are " computationally or manually selected as a representative from among the best genomes available for a species or clade". They are ~~chosen on~~ "chosen among eligible assemblies based on [some] criteria", the first criterium being the "manual selection". The UniProt reference proteomes ~~(Bateman *et al.*, 2017)~~ are "selected among all proteomes (manually and algorithmically, according to a number of criteria) to provide broad coverage of the tree of life". Ensembl--representative genomes (Yates *et al.*, 2022) are chosen among UniProt according to automatic dereplication rules ~~(Bursteinas *et al.*, 2016)~~. These three reference~~s~~ sets of ~~genomic~~genomes ~~sets~~ include proteomes of interest for biomedical and biotechnological research, and are thus highly impacted by socio--induced biases. These libraries bring together a qualitative and supposedly representative sampling of the taxonomic diversity of genomic data. The advantage for the users is that they do not have to manage the data sampling step. However, the use of these libraries also has limitations. First, the users have no control over the selection of genomes, so some taxonomic groups of interest for the user may not be represented. Second, the sampling density and the redundancy of the data ~~is~~ are not controlled, which may require a second sampling step. Finally, most of the time, these libraries do not include, most of the time, the genomes of undescribed organisms.

[1] These encompass both RefSeq--representative *sensu stricto* and RefSeq--reference according to the NCBI. RefSeq--reference genomes are "manually selected high--quality genome assembly that NCBI and the community have identified as being important as a standard against which other data are compared", while RefSeq--representative genomes are "computationally or manually selected as a representative from among the best genomes available for a species or clade that does not have a designated reference genome" (O'Leary *et al.*, 2016). To simplify, there are both referred to as RefSeq--representatives.

Supplementary Material S4 – Technical encoding

Distribution

MPS-Sampling is distributed as a Snakemake pipeline available at: https://github.com/rvcoudert/MPS_Sampling

Snakemake is a scalable, Python-based workflow manager (Köster and Rahmann, 2012). The execution of a Snakemake pipeline is managed by a master script, called a SnakeFile and coded in Python, which constructs a master diagram and lists the tasks to be performed. The master script automatically calls secondary scripts to accomplish the necessary tasks. The full MPS-Sampling master diagram is detailed in a flowchart shown in the Supplementary Material S5. All data tables used during the MPS-Sampling workflow are detailed in an entity relationship diagram (ERD) in Supplementary Material S6. For MPS-Sampling, it is chosen to manage computing environments of different scripts using Conda (Anaconda Documentation, 2020). Conda is an environment manager that automatically installs all necessary dependencies and prepares the appropriate environment when running a secondary script with Snakemake. For MPS-Sampling, the main dependency is the Linclust package (Steinegger and Söding, 2018) from the MMseqs2 suite (Steinegger and Söding, 2017). All other intermediate scripts use the R language. For MPS-Sampling, there are therefore two Conda environments used: one to launch Linclust and one to launch the R scripts.

Format of input files

MPS-Sampling has two types of input files: FASTA files for protein sequences and a CSV (Comma-Separated Values) file for indexing genomes.

There is one FASTA file per protein family. In each FASTA file, the sequences must be named after the genome to which they belong: the name of the sequences is the primary key linking sequence and genome. For example, in Figure 1, the $\mathbf{g_B}$ genome has a sequence for the **uL1**, **uL3** and **uL4** families, but not for the **uL2** family. This means that the FASTA files corresponding to the **ul1**, **uL3** and **uL4** families each contain a single-copy sequence belonging to the $\mathbf{g_B}$ genome. On the other hand, the FASTA file corresponding to the **uL2** family does not contain any $\mathbf{g_B}$ genome sequence.

The CSV file constituting the genome index has two columns separated by a comma: **genomeAccession** and **priority_score**.

- **genomeAccession**.
- **priority_score** defines the order of priority for the selection of MPS-representative genomes. The higher a genome score, the more priority it is chosen. This score constitutes the first priority rule when choosing representative MPS-genomes (Supplementary Material S10).

Format of intermediate files

The intermediate files are described in the entity relationship diagram (ERD) in Supplementary Material S6. All files are of CSV type, except the TSV files generated by Linclust and two RDS files to respectively export the similarity matrix and hierarchical clustering. TSV files are simply CSV files but with a tab as separator instead of a comma (Tabulation-Separated Values). The RDS format is chosen to export the similarity matrix and hierarchical clustering because of its practicality. It allows to quickly write and read data with R as well as compress backups.

Format of output files

The list of the selected MPS-representative genomes is accessible in the table **MPS-representatives** containing only a single column: **MPS-representative** the accession number of the MPS-representative genomes. The link between the input genomes and the MPS-representative genomes is accessible in the table **MPS-links**.

a mis en forme : Espace Après : 0 pt
a mis en forme : Retrait : Première ligne : 0 cm
Code de champ modifié

Running MPS-Sampling

An example for running several runs of MPS-Sampling in a single-line command is :

```
snakemake --use-conda --cores 10 -s /home/MPS-Sampling/Snakefile -d /home/Data/RiboDB -config deltas=[0.5,1]
```

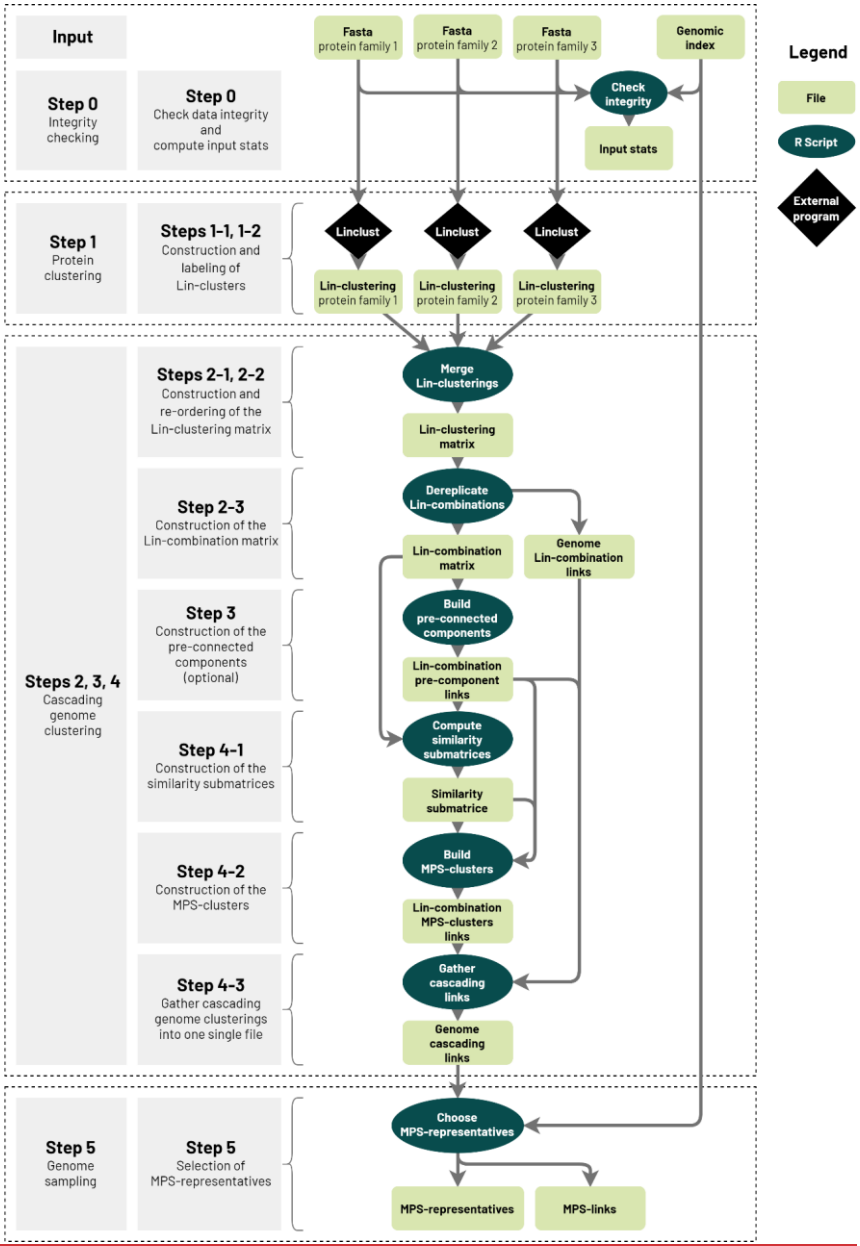
This command line launches MPS-Sampling in parallelization with 10 cores, using the SnakeFile localized in "/home/MPS-Sampling/Snakefile", applied to the data set localized in "/home/Data/RiboDB", in order to generate two samples with $\Delta = 0.5$ and $\Delta = 1$.

a mis en forme : Police :(Par défaut) Barlow

a mis en forme : Police :(Par défaut) Barlow

Supplementary Material S5 – Flowchart of MPS-Sampling

This flowchart shows the data analysis processed by MPS-Sampling, centralized thanks to a Snakemake pipeline. Files are colored in light green. The only external program is Linclust and colored in black. All other tasks are carried out by home-made R scripts, colored in strong blue.

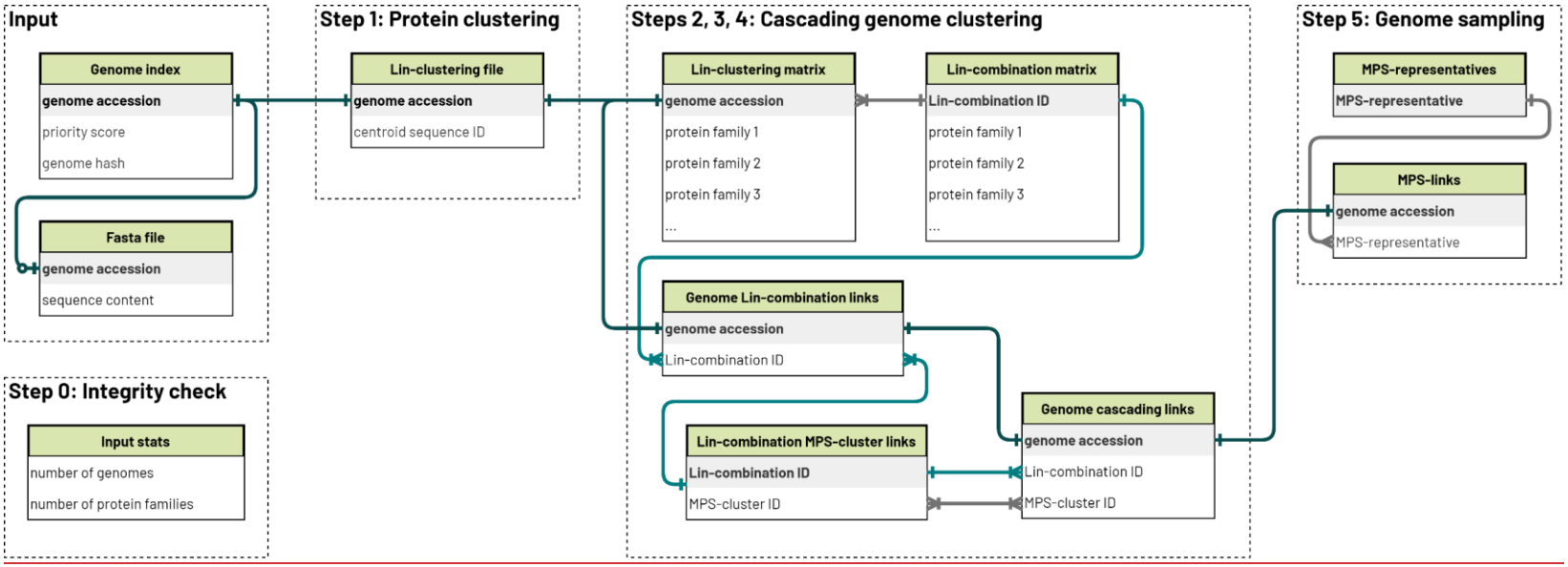


Supplementary Material S6 – Entity relationship diagram (ERD) of MPS-Sampling

This entity relationship diagram (ERD) (Song *et al.*, 1995) below details the relational schema of MPS-Sampling.

Each box presents a distinct data table with its columns. Primary keys are shaded in grey and links between columns are indicated by arrows.

For example, the **Genome index** table has three columns: (i) **genome accession**, which represents the accession number of the genome and is the primary key of the table; (ii) **priority score**, which represents the priority score for choosing MPS-representative genomes (Supplementary Material S10); and (iii) **genome hash**, which contains the hashing value of the accession number of the genome, to separate equivalent choices using a pseudo-random criterion (Supplementary Material S10). The primary key genome accession is shared with the tables Fasta files and Genome Lin-combination links. The type of arrows indicates that there is strictly one entry in the Genome index table for one entry in the Genome Lin-combination links table. This is consistent because each genome belongs strictly to one and only one Lin-combination. On the other hand, for each entry in Genomic index, there is zero or one entry in the Fasta files. This is still consistent because the absence of a sequence in a genome is authorized (zero entry), as is a single-copy sequence (one entry), but duplicated sequences are not accepted (multiple entries forbidden). Conversely, each sequence must have one name and one name only, so each sequence is attached to strictly one genome.



a mis en forme : Normal

a mis en forme : Police :11 pt, Gras, Couleur de police : Automatique, (Asiatique) Chinois (traditionnel, Taïwan)

Code de champ modifié

Code de champ modifié

a mis en forme : Police :11 pt, Non Gras, Couleur de police : Automatique, (Asiatique) Chinois (traditionnel, Taïwan)

a mis en forme : Police :11 pt, Non Gras, Couleur de police : Automatique, (Asiatique) Chinois (traditionnel, Taïwan)

a mis en forme : Police :11 pt, Non Gras, Couleur de police : Automatique, (Asiatique) Chinois (traditionnel, Taïwan)

a mis en forme : Police :11 pt, Non Gras, Couleur de police : Automatique, (Asiatique) Chinois (traditionnel, Taïwan)

a mis en forme : Police :11 pt, Non Gras, Couleur de police : Automatique, (Asiatique) Chinois (traditionnel, Taïwan)

a mis en forme : Police :11 pt, Non Gras, Couleur de police : Automatique, (Asiatique) Chinois (traditionnel, Taïwan)

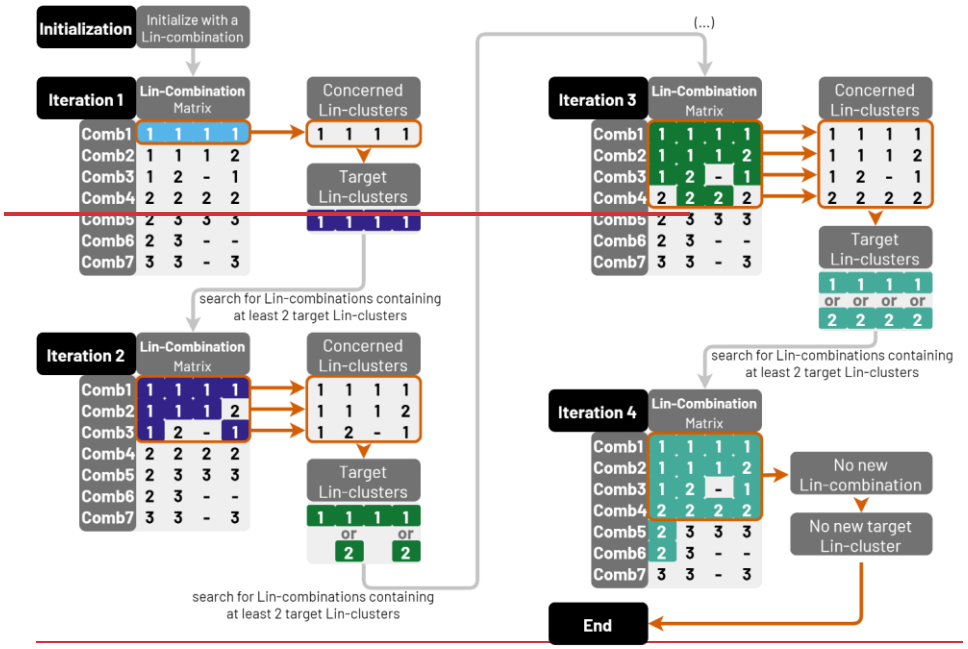
Supplementary Material S74 – From Lin-combinations to pre-connection

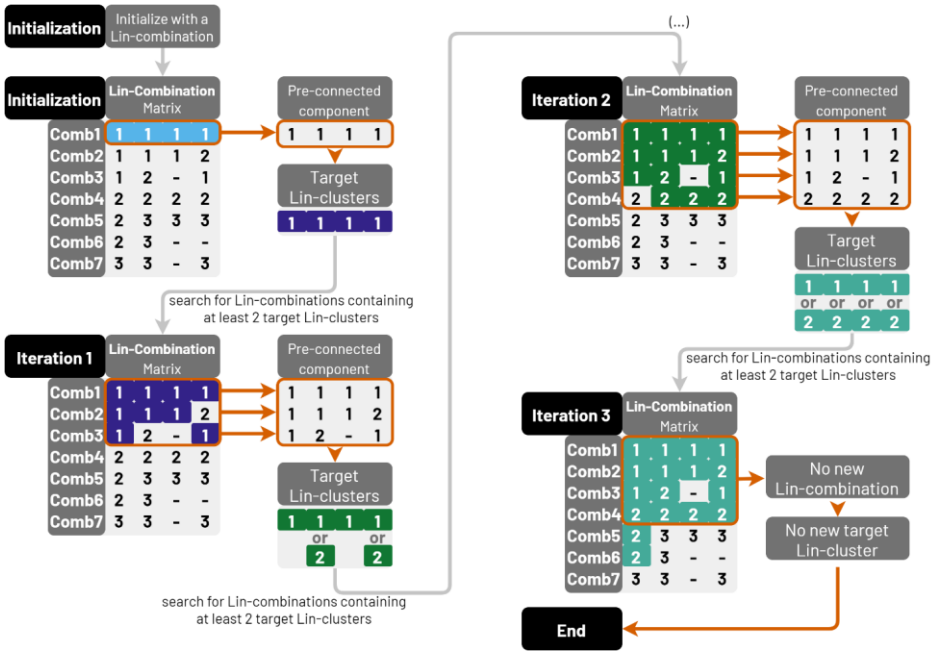
Here an eExample of the construction of a pre-connected component during the pre-connection, with the parameter minNbLinclusters = 2. From a starting Lin-combination, close Lin-combinations are iteratively absorbed according to a single common reference of target Lin-clusters.

- **Initialization:** The pre-connected component is initialized with a given Lin-combination: **Comb1**.
- **Iteration 1:** Only the Lin-clusters of the starting Lin-combination are targeted: **(1111)**.
- **Iteration 12:** Lin-combinations sharing at least minNbLinclusters = 2 minNbLinclusters = 2 targeted Lin-clusters are searched. Herelt results that two 2 Lin-combinations, **Comb2** and **Comb3**, share have at least 2 target Lin-clusters with **Comb1**, **Comb2** and **Comb3**. These two 2 Lin-combinations are added to the current pre-connected component. This is the aggregative aspect of pre-connection. For the next iteration, the list of the target Lin-clusters is updated with the information carried by **Comb2** and **Comb3**.the Lin-clusters of the 3 involved Lin-combinations (**Comb1**, **Comb2**, and **Comb3**) are gathered into a single common reference of target references.
- **Iteration 23:** According to the updated list of target Lin-clusters, Searching Lin-combination with at least minNbLinclusters = 2 target Lin-clusters, a new Lin-combination matches: **Comb4** is identified. This Lin-combination is addedadded, and the current pre-connected component contains now component contains now 4 Lin-combinations: **Comb1**, **Comb2**, **Comb3**, and **Comb4**. The list of the target Lin-custers is again updated.
- **Iteration 34:** No new compatible Lin-combination is found. Thus the pre-connected component is now stablestable, and its delineation ends.

To resume, from the Lin-combination **Comb1**, a pre-connected component has been delineated using the parameter minNbLinclusters = 2 through 34 iterations. It and contains four 4 Lin-combinations: **Comb1**, **Comb2**, **Comb3**, and **Comb4**. The process could be used to build a second pre-connected component. Starting from the Lin-combination **Comb5**, it ends containing three Lin-combinations: **Comb5**, **Comb6**, and **Comb7**.

a mis en forme : Justifié


Initialization
Initialize with a Lin-combination
Iteration 1
Lin-Combination Matrix
Concerned Lin-clusters
Target Lin-clusters
search for Lin-combinations containing at least 2 target Lin-clusters
Iteration 2
search for Lin-combinations containing at least 2 target Lin-clusters
(...)
Iteration 3
search for Lin-combinations containing at least 2 target Lin-clusters
Iteration 4
No new Lin-combination
No new target Lin-cluster
End


Initialization
Initialize with a Lin-combination
Initialization
Lin-Combination Matrix
Pre-connected component
Target Lin-clusters
search for Lin-combinations containing at least 2 target Lin-clusters
Iteration 1
search for Lin-combinations containing at least 2 target Lin-clusters
(...)
Iteration 2
search for Lin-combinations containing at least 2 target Lin-clusters
Iteration 3
No new Lin-combination
No new target Lin-cluster
End

Supplementary Material S8 – Dice index and similarity matrix

Noting A and B the two compared Lin-combinations, the Dice index is given by the formula:

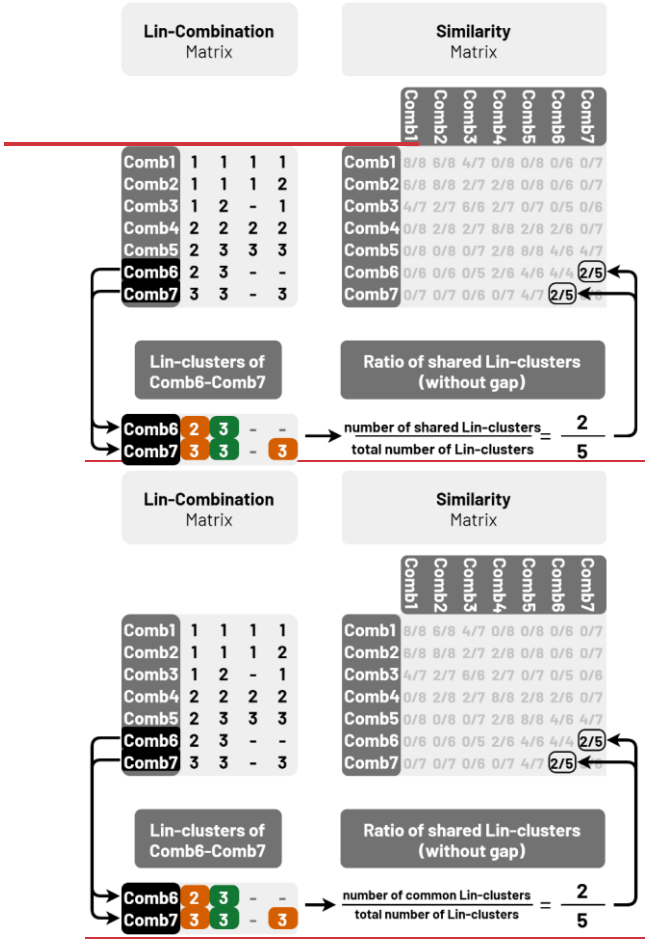
$$\frac{2 \times card(A \cap B)}{card(A) + card(B)}$$

The figure below illustrates the computation of the Dice index between **Comb6** (2, 3, –, -) and **Comb7** (3, 3, –, 3). The number of common Lin-clusters and the total number of Lin-clusters (without missing values) are respectively 2 and 5. The Dice index is thus equals to 2 / 5.

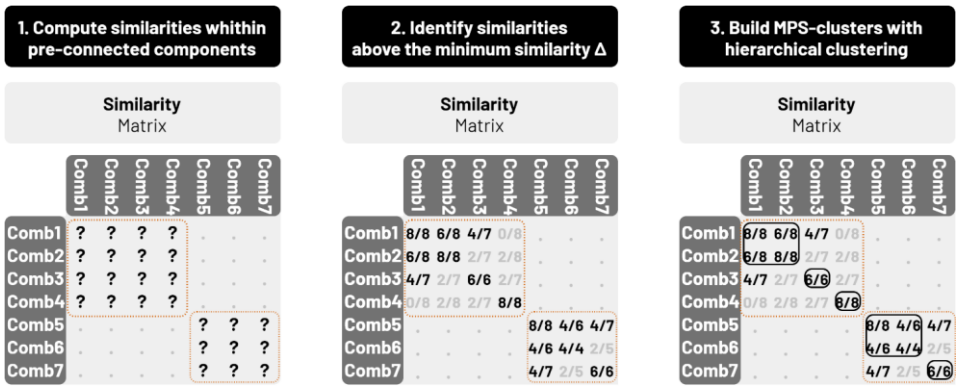
The figure below illustrates ~~T~~the computation of the Dice index between **Comb6** (2, 3, –, -) and **Comb7** (3, 3, –, 3). ~~the pair of the two Lin-combinations (Comb6 - Comb7) will be detailed. To begin with, consider the Lin-clusters of the two Lin-combinations: (2, 3, ,) and (3, 3, , 3). Then count the~~ The number of common Lin-clusters and the total number of Lin-clusters (without missing values) are ~~,~~ respectively 2 and 5. ~~At the end, the~~ The Dice index is thus equals to 2 / 5.

The Dice index handles missing values. It completely ignores the missing values when the protein family is missing in both Lin-combinations (e.g. protein family 3, missing in **Comb6** and **Comb7**) and takes into account when a protein is missing in only one Lin-combination (e.g. protein family 4, missing in **Comb6** but present in **Comb7**).

a mis en forme : Police :Gras
a mis en forme : Police :Gras
a mis en forme : Police :Gras
a mis en forme : Police :Gras
a mis en forme : Police :Gras
a mis en forme : Police :Gras



If the pre-connection is used, only the square submatrices corresponding to the pre-connected components are computed. In the figure below, two submatrices have been computed, corresponding to the two pre-connected components delineated in the Supplementary Material S7. Five MPS-clusters have been built, corresponding to the five MPS-clusters present in the Figure 1.


1. Compute similarities whithin pre-connected components
Similarity Matrix
2. Identify similarities above the minimum similarity Δ
Similarity Matrix
3. Build MPS-clusters with hierarchical clustering
Similarity Matrix
Comb1 Comb2 Comb3 Comb4 Comb5 Comb6 Comb7
Comb1 ? ? ? ? . . .
Comb2 ? ? ? ? . . .
Comb3 ? ? ? ? . . .
Comb4 ? ? ? ? . . .
Comb5 . . . . ? ? ?
Comb6 . . . . ? ? ?
Comb7 . . . . ? ? ?
Comb1 8/8 6/8 4/7 0/8 . . .
Comb2 6/8 8/8 2/7 2/8 . . .
Comb3 4/7 2/7 6/6 2/7 . . .
Comb4 0/8 2/8 2/7 8/8 . . .
Comb5 . . . . 8/8 4/6 4/7
Comb6 . . . . 4/6 4/4 2/5
Comb7 . . . . 4/7 2/5 6/6
Comb1 8/8 6/8 4/7 0/8 . . .
Comb2 6/8 8/8 2/7 2/8 . . .
Comb3 4/7 2/7 6/6 2/7 . . .
Comb4 0/8 2/8 2/7 8/8 . . .
Comb5 . . . . 8/8 4/6 4/7
Comb6 . . . . 4/6 4/4 2/5
Comb7 . . . . 4/7 2/5 6/6

Supplementary Material S9 – Calculation of the similarity matrix

To minimize quadratic complexity, the similarity matrix is calculated per column using the properties of the Dice index (Supplementary Material S8). For two sets A and B, the formula of the Dice index is:

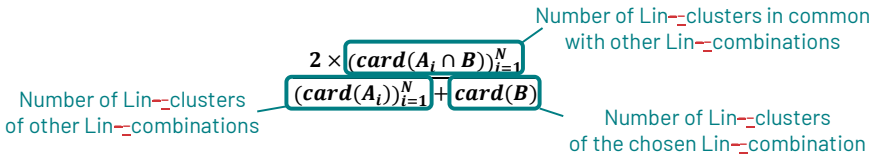
$$\frac{2 \times \boldsymbol{card}(\boldsymbol{A} \cap \boldsymbol{B})}{\boldsymbol{card}(\boldsymbol{A}) + \boldsymbol{card}(\boldsymbol{B})}$$

Let $\boldsymbol{B}$ be a given Lin-combination, calculating the column corresponding to $\boldsymbol{B}$ is equivalent to calculating the Dice index between $\boldsymbol{B}$ and all the $\boldsymbol{N}$ other Lin-combinations $\{\boldsymbol{A_i}\}_{\boldsymbol{i=1}}^{\boldsymbol{N}}$ ($\boldsymbol{B}$ included), i.e. the vector of size $\boldsymbol{N}$:

$$\left(\frac{2 \times \boldsymbol{card}(\boldsymbol{A_i} \cap \boldsymbol{B})}{\boldsymbol{card}(\boldsymbol{A_i}) + \boldsymbol{card}(\boldsymbol{B})}\right)_{\boldsymbol{i=1}}^{\boldsymbol{N}} = \frac{2 \times (\boldsymbol{card}(\boldsymbol{A_i} \cap \boldsymbol{B}))_{\boldsymbol{i=1}}^{\boldsymbol{N}}}{(\boldsymbol{card}(\boldsymbol{A_i}))_{\boldsymbol{i=1}}^{\boldsymbol{N}} + \boldsymbol{card}(\boldsymbol{B})}$$

The idea is to calculate the numerator-vector and the denominator-vector separately. In the numerator-vector, the main factor is $(\boldsymbol{card}(\boldsymbol{A_i} \cap \boldsymbol{B}))_{\boldsymbol{i=1}}^{\boldsymbol{N}}$, which simply is the number of Lin-clusters in common between $\boldsymbol{B}$ and the other Lin-combinations $\{\boldsymbol{A_i}\}_{\boldsymbol{i=1}}^{\boldsymbol{N}}$. In the denominator-vector, the main term is $(\boldsymbol{card}(\boldsymbol{A_i}))_{\boldsymbol{i=1}}^{\boldsymbol{N}}$, which is just the number of Lin-clusters per Lin-combination, plus the number of Lin-clusters from ~~B.~~ $\boldsymbol{B}$.

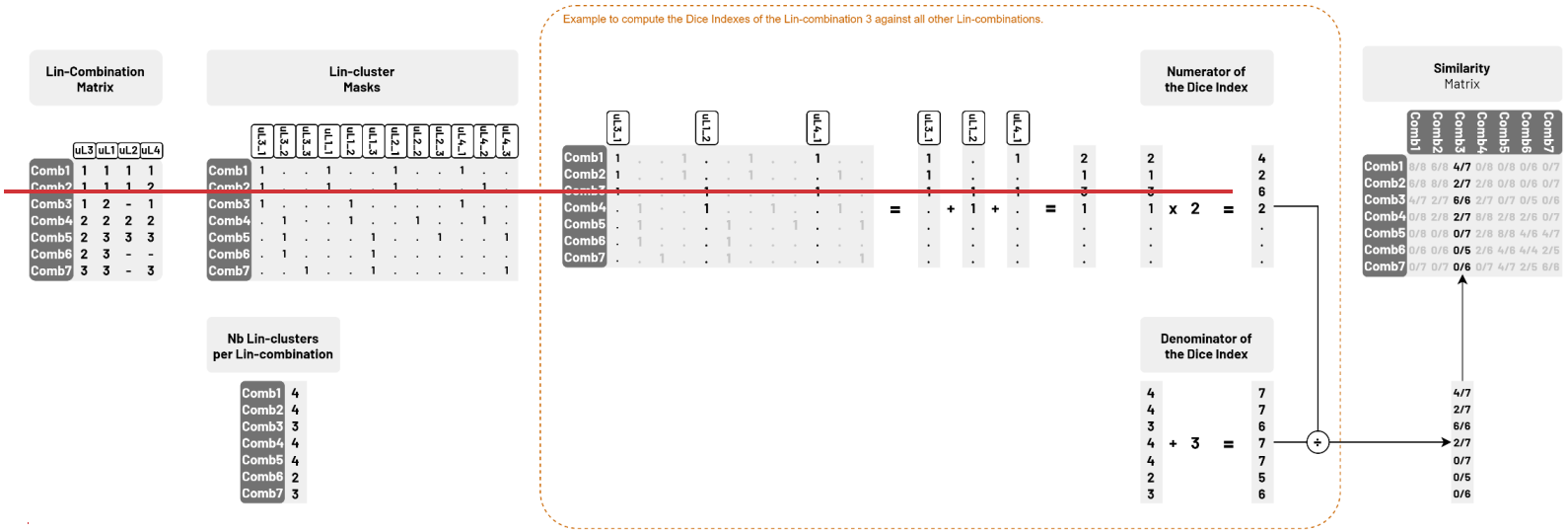
a mis en forme : Police :Non Gras

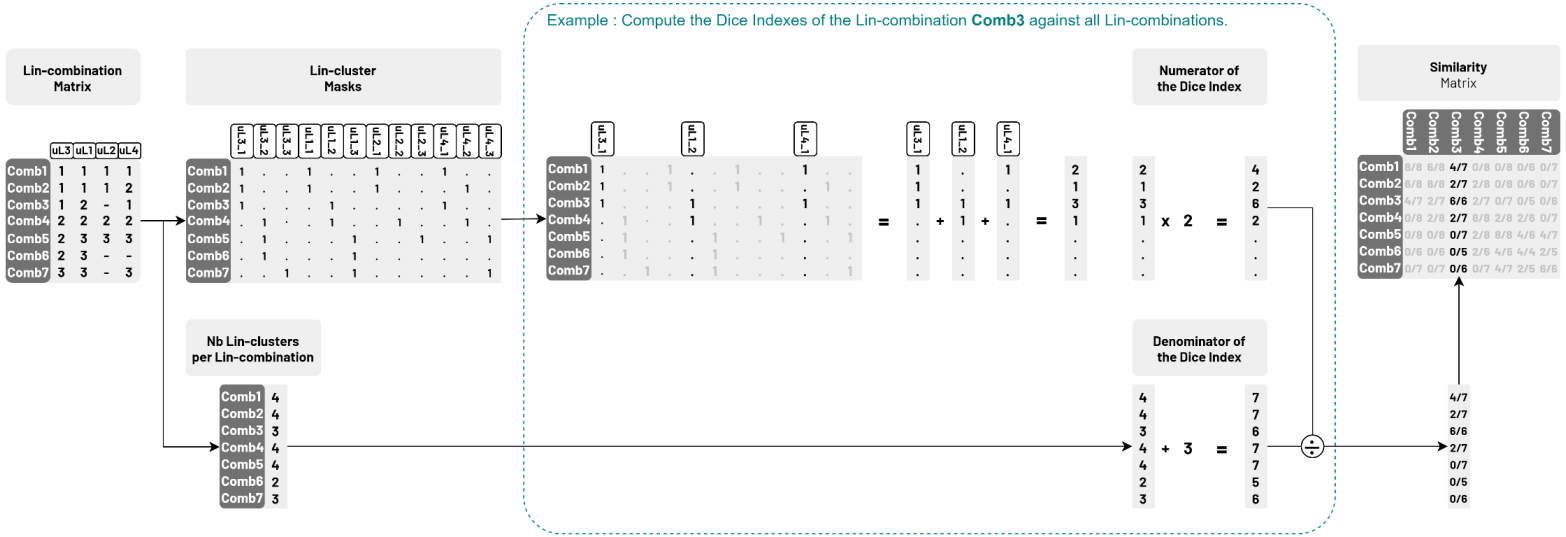


This allows the calculation of the numerator-vector and denominator-vector separately. The figure below presents the computation of the Dice indexes between **Comb3** and the 7 other Lin-combinations.

~~**Supplementary Material S7 – Calculation of the similarity matrix (example)**~~

~~To minimize quadratic complexity, the similarity matrix is calculated per column using the properties of the Dice index (Supplementary Material S5). An example is given in with the calculation of the Dice indices of the Lin-combination **Comb3** with the seven other Lin-combinations from the Supplementary Material S5.~~


Example to compute the Dice Indexes of the Lin-combination 3 against all other Lin-combinations.
Lin-Combination Matrix
Lin-cluster Masks
Numerator of the Dice Index
Similarity Matrix
Nb Lin-clusters per Lin-combination
Denominator of the Dice Index



Firstly, the Lin-combination matrix is transformed through one-hot encoding (Harris and Harris, 2012). This means calculating the mask for each Lin-cluster. The mask of a Lin-cluster is a binary vector over all the protein sequences evaluating membership of the Lin-cluster: 1 if the sequence belongs to the Lin-cluster, 0 otherwise. The resulting matrix is called the mask matrix. This matrix has as many columns as there are different Lin-clusters among all the protein families and as many rows as there are Lin-combinations. The mask matrix has 12 columns because there are 12 different Lin-clusters among the 4 protein families considered; it also has 7 rows because there are 7 different Lin-combinations. At the same time, the number of Lin-clusters per Lin-combination is calculated and stored in a vector.

The numerator-vector is calculated. The masks of the Lin-clusters in the concerned Lin-combination are added together to obtain the number of Lin-clusters in common with all the other Lin-combinations. Here, **Comb3** has 3 Lin-clusters: **ul3_1**, **ul1_2** and **ul4_1**. The masks of these 3 Lin-clusters are added together to obtain the number of Lin-clusters in common between **Comb3** and all the other Lin-combinations:

$$v = \begin{pmatrix}1\\1\\1\\.\\.\\.\\.\end{pmatrix} + \begin{pmatrix}.\\.\\1\\1\\.\\.\\.\end{pmatrix} + \begin{pmatrix}1\\.\\1\\.\\.\\.\\.\end{pmatrix} = \begin{pmatrix}2\\1\\3\\1\\0\\0\\0\end{pmatrix}$$

The digit **2** in the first coordinate means that **Comb3** shares 2 *Lin-clusters* with **Comb1**. This vector $\boldsymbol{v}$ is multiplied by 2 to obtain the numerator-vector of the Dice indices. For **Comb3**, the numerator-vector obtained is:

$$2 \times \boldsymbol{v} = 2 \times \begin{pmatrix} \mathbf{2} \\ \mathbf{1} \\ \mathbf{3} \\ \mathbf{1} \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \end{pmatrix} = \begin{pmatrix} \mathbf{4} \\ \mathbf{2} \\ \mathbf{6} \\ \mathbf{2} \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \end{pmatrix}$$

The denominator-vector is then calculated. The number of Lin-clusters per Lin-combination is added to the number of Lin-clusters in the concerned Lin-combination. **Comb3** has 3 Lin-clusters, so the obtained denominator-vector is:

$$\begin{pmatrix} \mathbf{4} \\ \mathbf{4} \\ \mathbf{3} \\ \mathbf{4} \\ \mathbf{4} \\ \mathbf{2} \\ \mathbf{3} \end{pmatrix} + \mathbf{3} = \begin{pmatrix} \mathbf{7} \\ \mathbf{7} \\ \mathbf{6} \\ \mathbf{7} \\ \mathbf{7} \\ \mathbf{5} \\ \mathbf{6} \end{pmatrix}$$

By dividing the numerator-vector by the denominator-vector, the column of the similarity matrix corresponding to the Lin-combination **Comb3** is obtained, i.e.:

$$\begin{pmatrix} \mathbf{4} \\ \mathbf{2} \\ \mathbf{6} \\ \mathbf{2} \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \end{pmatrix} / \begin{pmatrix} \mathbf{7} \\ \mathbf{7} \\ \mathbf{6} \\ \mathbf{7} \\ \mathbf{7} \\ \mathbf{5} \\ \mathbf{6} \end{pmatrix} = \begin{pmatrix} \mathbf{4/7} \\ \mathbf{2/7} \\ \mathbf{6/6} \\ \mathbf{2/7} \\ \mathbf{0/7} \\ \mathbf{0/5} \\ \mathbf{0/6} \end{pmatrix}$$

Calculating the similarity matrix by column saves time compared to calculating it cell by cell and thus limits the quadratic complexity.

Supplementary Material S8 – From pre-connection to MPS-clustering

1. The pre-connected components indicate which parts to calculate in the sparse similarity matrix. In this example, two pre-connected components have been previously computed (Supplementary Material S4). The first one with **Comb1** to **Comb4** and the second one with **Comb5** to **Comb7**. As a result, only two similarity submatrices are computed (orange dotted lines).

Among computed similarities, only high similarities are used. In this example, the minimum similarity Δ was set to Δ=0.5 ; thus all similarities greater or equal to 0.5 are used. All similarities greater than Δ=0.5 are highlighted in black in the similarity matrix and are marked as links in the graph theory representation. In this example, four links have been found: **Comb1-Comb2**, **Comb2-Comb3**, **Comb5-Comb6** and **Comb5-Comb7**.

2. Connected components are built using all the found links above the minimum similarity Δ=0.5. In this example, 3 connected components are built (blue dashed lines): {**Comb1**, **Comb2**, **Comb3**}, {**Comb4**} and {**Comb5**, **Comb6**, **Comb7**}.

2. MPS-clusters are built using hierarchical method with complete-linkage. It leads to build 5 MPS-clusters (black lines). 2 MPS-clusters gather 2 Lin-combinations, respectively {**Comb1**, **Comb2**} and {**Comb5**, **Comb6**}. 3 Lin-combinations remain isolated into singleton MPS-clusters, respectively {**Comb3**}, {**Comb4**}, and {**Comb7**}.

Note that the pre-connection avoids to compute empty parts of the similarity matrix, saving a substantial amount of time. In this example, only 4x4 + 3x3 = 16 + 9 = 25 similarities have been computed among the total of 7x7 = 49 similarities. As a result, the similarity matrix computation was reduced to 50%.

The pre-connection, the connection and the complete-linkage are three increasingly fine partitions. In this example, it is visible that each MPS-cluster is completely included in a connected component. Then each connected component is itself completely included in a pre-connected component.

This degressive partitioning is a suitable divide-and-conquer: it splits the complete dataset into preliminary subparts, breaking the quadratic complexity, without discarding any important link. To find interesting links, it has a perfect sensibility, but a medium specificity. (Considering the founded links, there are many false positives, but no false negative.)

a mis en forme : Gauche

Supplementary Material S10 – Priority rules for the choice of the MPS-representative genomes

~~(legend on the previous page)~~

Supplementary Material S10~~9~~ – Priority rules for the choice of the MPS--representative genomes

The MPS-representative genomes are selected according to rigorous priority rules, which favor:

(1) The genomes with the highest user-defined priority score. For our analysis, a priority score is given to each genome based on several priority criteria. For example, a genome indicated RefSeq-representative and completed **(RC)** has a score of 32 + 8 = 40.

(2) The genomes with the largest distribution across the protein families used as input, i.e. the genomes with the fewest missing protein families.

(3) The genomes with the best centrality within the MPS-cluster considered, i.e. with the highest possible average Dice index to the other genomes. This is equivalent to the genomes whose Lin-clusters are the most frequent (Supplementary Material S11).

(4) Genomes whose name has the lowest hash value. Hash values are calculated with the function **hash()** of the package **R rlang**.

Step			Description
1	**Tag**	**Priority Score**	Genome with the ~~best~~ highest user-defined priority score
	(R)	32	Genome indicated **RefSeq--representative**[2] by the **NCBI**
	(T)	16	Genome from a type strain, indicated by the **NCBI**
	(C)	8	Genome with an assembly level estimated Complete
	(S)	4	Genome with an assembly level estimated Scaffold
	(U)	2	Genome with an assembly level estimated Unassembled
	(d)	-1	Genome whose quality is questionable
2	**Protein ~~coverage~~distribution**		Genome with the ~~most single-copy sequences~~largest distribution across protein families used as input, i.e. with the fewest missing values
3	**Centrality**		Genome whose Lin--clusters are the most frequent in the concerned MPS--cluster
4	**Pseudo--randomization**		Genome with the smallest hashing value

[2] These encompass both RefSeq--representative *sensu stricto* and RefSeq--reference according to the NCBI. RefSeq--reference genomes are "manually selected high--quality genome assembly that NCBI and the community have identified as being important as a standard against which other data are compared", while RefSeq--representative genomes are "computationally or manually selected as a representative from among the best genomes available for a species or clade that does not have a designated reference genome" (O'Leary *et al.*, 2016). To simplify, there are both referred to as RefSeq--representatives.

Supplementary Material S11 – Centrality criterium

This Supplementary Material will provide the proof that central genomes according to the Dice index can be chosen according to the frequency of their Lin-clusters.

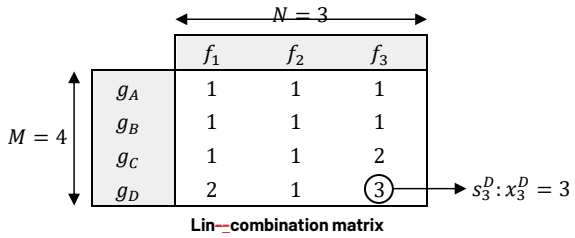
Definition 1: In a set of genomes, the **central genome(s)** is/are the genome(s) whose average Dice index with all the other genomes is the greatest.

In other words, in a set of genomes, centrality is defined by the Dice index.

Theorem 1: In a set of genomes, the central genomes are the genomes whose Lin-clusters are the most frequent.

Introduction

Let's consider M genomes and N protein families. For each genome i and each protein family j, a sequence s_j^i belongs to a Lin--cluster x_j^i. Let's see an example with $M = 4$ genomes (g_A, g_B, g_C, g_D) and $N = 3$ protein families (f_1, f_2, f_3). It gives a matrix (x_j^i) of 4 rows and 3 columns.



Lin--combination matrix

For example, the sequence s_3^D comes from the genome g_D, is attached to the protein family f_3 and was put in the Lin--cluster x_3^D which is labeled 3, i.e. $x_3^D = 3$.

Frequency of Lin--clusters

Now, for each Lin--cluster x_j^I, its frequency $freq(x_j^I)$ can be calculated ~~its frequency $freq(x_j^I)$~~. Noting the Dirac function δ, this frequency can be given by:

$$freq(x_j^I) = \frac{1}{M} \times \sum_{i=1}^{M} \delta(x_j^I, x_j^i)$$

Then an average frequency can be associated to each genome g_I by calculated the average of the frequency of its Lin--clusters. This average frequency can be given by:

$$freq(g_I) = \frac{1}{N} \times \sum_{j=1}^{N} freq(x_j^I) = \frac{1}{NM} \times \sum_{j=1}^{N} \sum_{i=1}^{M} \delta(x_j^I, x_j^i)$$

Let's see some applications from the previous example.

	f_1	f_2	f_3	$freq$
g_A	3/4	4/4	2/4	9/12
g_B	3/4	4/4	2/4	9/12
g_C	3/4	4/4	1/4	8/12
g_D	1/4	4/4	1/4	6/12

Average frequency of Lin-clusters

For example, the Lin-cluster $x_3^D = 3$ of the protein family f_3 appears once out of four ~~genomes (~~genomes (g_D), so its frequency is equal to $freq(x_3^D) = 1/4$. Another example is the Lin-cluster $x_1^A = 1$ of the protein family f_1. It appears in three genomes out of four (g_A, g_B et g_C), so its frequency is $freq(x_1^A) = 3/4$. Last, the genome g_A has three Lin-clusters with the respective frequencies $3/4$, $4/4$ and $2/4$, so the average frequency of the Lin-clusters of this genome g_A is:

$$freq(g_A) = \frac{1}{3} \times (\frac{3}{4} + \frac{4}{4} + \frac{2}{4}) = \frac{9}{12}$$

Dice index and centrality

Now, let's introduce the Dice index between the Lin-clusters of two genomes. In this context, the Dice index counts the proportion of common Lin-clusters between two genomes. The Dice index is given by the formula:

$$Dice(g_A, g_B) = \frac{1}{2\mathrm{N}} \times \sum_{j=1}^{N} 2\delta(x_j^A, x_j^B) = \frac{1}{\mathrm{N}} \times \sum_{j=1}^{N} \delta(x_j^A, x_j^B)$$

Let's see what happens for the previous example.

	g_A	g_B	g_C	g_D	$centrality$
g_A	6/6	6/6	4/6	2/6	18/24
g_B	6/6	6/6	4/6	2/6	18/24
g_C	4/6	4/6	6/6	2/6	16/24
g_D	2/6	2/6	2/6	6/6	12/24

Dice index matrix

For example, the two genomes g_A and g_B share identical Lin-clusters for the three protein ~~families :~~families: $(1, 1, 1)$ so their Dice index is $Dice(g_A, g_B) = 6/6$. On the contrary, the two genomes g_C and g_D share only one Lin-cluster, that of the second protein family f_2, so their Dice index is: $Dice(g_A, g_B) = 2/6$.

Next, the centrality of a genome can be defined by its average Dice index to all the genomes (including itself). It can be given by the formula:

$$centrality(g_A) = \frac{1}{\mathrm{M}} \times \sum_{i=1}^{M} Dice(g_A, g_i) = \frac{1}{\mathrm{MN}} \times \sum_{i=1}^{M} \sum_{j=1}^{N} \delta(x_j^A, x_j^i)$$

The following lemma will link the centrality of a genome and the frequency of its Lin-clusters.

Lemma 1: The centrality of a genome is equals to the average frequency of its Lin-clusters, i.e.:

$$centrality(g_A) = freq(g_A)$$

Proof:

$$centrality(g_A) = \frac{1}{\mathrm{M}} \times \sum_{i=1}^{M} Dice(g_A, g_i) = \frac{1}{\mathrm{MN}} \times \sum_{i=1}^{M} \sum_{j=1}^{N} \delta\left(x_j^A, x_j^i\right)$$

The two sums can be switched because the sum is finite.

$$centrality(g_A) = \frac{1}{\mathrm{NM}} \times \sum_{j=1}^{N} \sum_{i=1}^{M} \delta\left(x_j^A, x_j^i\right) = \frac{1}{N} \times \sum_{j=1}^{N} freq\left(x_j^A\right) = freq(g_A)$$

□

Then the following theorem will be easily proved.

Theorem 1: In a set of genomes, the central genomes are the genomes whose Lin-clusters are the most frequent.

Prof: It is simply because the centrality of a genome is equal to the average frequency of its Lin-clusters. So the genome whose Lin-clusters have the highest average frequency will also be the genome with the higher centrality.

□

Corollary 1: The most central genomes can be chosen according to the frequency of the Lin-clusters. The genomes whose Lin-clusters have the highest average frequency will be the most central, according to the Dice index.

In the previous example, the most central genomes are g_A and g_B because the average frequency of their Lin-clusters and their centrality will be the higher, i.e. $\frac{9}{12} = \frac{18}{24}$.

Supplementary Material S12 – Preparation of the bacterial dataset

~~**Distribution**~~

~~MPS-Sampling is distributed as a Snakemake pipeline. Snakemake is a scalable, Python-based workflow manager (Köster and Rahmann, 2012). The execution of a Snakemake pipeline is managed by a master script, called a SnakeFile and coded in Python, which constructs a master diagram and lists the tasks to be performed. The master script automatically calls secondary scripts to accomplish the necessary tasks. The full MPS-Sampling master diagram is detailed in a flowchart shown in the Supplementary Material S12. All data tables used during the MPS-Sampling workflow are detailed in an entity-relationship diagram (ERD) in Supplementary Material S13. For MPS-Sampling, it is chosen to manage computing environments of different scripts using Conda. Conda is an environment manager that automatically installs all necessary dependencies and prepares the appropriate environment when running a secondary script with Snakemake. For MPS-Sampling, the main dependency is the Linclust package from the MMseqs2 suite. All other intermediate scripts use the R language. For MPS-Sampling, there are therefore two Conda environments used: one to launch Linclust and one to launch the R scripts.~~

Supplementary Material S12 – Preparation of the bacterial dataset

Using a Snakemake pipeline has several advantages.

- **Automation.** When running a pipeline, Snakemake uses the master diagram found in the SnakeFile to list tasks to be performed based on available input files and requested output files. Then, the listed tasks are automatically launched based on the progress of the master diagram. It is also possible to scan a grid of parameters when launching a Snakemake pipeline. For MPS-Sampling, obtaining samples of different sizes through different values of Δ was launched in a single line command.
- **Parallelization.** Depending on the parallelization parameter indicated, Snakemake launches the different tasks in dedicated and independent processes. The management of processors, working memory, and disk access is automated.
- **Traceability.** For each execution of a pipeline, a log file is generated. Snakemake writes down the list of tasks to be carried out. Any task launched is then recorded in the log file, with the input files used, the script executed, the output files generated, the start time and the end time. In particular, a Snakemake log file indicates the execution times of the different intermediate steps, and therefore of the entire pipeline. In addition, it is easy to check how a particular output file is generated, by consulting the SnakeFile and the code of the associated scripts.
- **Progressiveness.** As running tasks are managed by Snakemake, the execution of a pipeline can be interrupted at any time and resumed later from the tasks already completed. When new data is added for analysis, Snakemake does not recalculate results for old data that has already been analyzed. Likewise, if the master diagram is modified, old tasks already completed are not re-executed. In particular, Snakemake checks script modification dates and recalculates all output files when a script is modified. That is to say that the creation date of the output files must be later than the last modification date of the script which generated them. This guarantees the updating and fidelity of the output files with respect to the code present in the master script and secondary scripts.
- **Reliability.** A Snakemake pipeline is thus resistant to errors. Failed tasks are indicated in the log file by Snakemake, and the associated dependent tasks are aborted.
- **Reproducibility.** If the output files, or any intermediate files, are deleted and the Snakemake pipeline restarted via the same command line, it is guaranteed to get exactly the same output files again.
- **Adaptability.** As Snakemake calls the secondary scripts of the tasks to be performed, it is possible to incorporate any language. It is even possible to use different Conda environments to compare output from different package versions. For MPS-Sampling, all intermediate scripts are coded in R, but it is possible to recode them in Python, C, Bash, or even mix scripts from different languages in the same Snakemake pipeline.

Supplementary Material S12 – Preparation of the bacterial dataset

MPS Sampling has two types of input files: FASTA files for protein sequences and a CSV (Comma-Separated Values) file for indexing genomes.

There is one FASTA file per protein family. In each FASTA file, the sequences must be named after the genome to which they belong: the name of the sequences is the primary key linking sequence and genome. For example, in Figure 1, the g_B genome has a sequence for the **uL1**, **uL3** and **uL4** families, but not for the **uL2** family. This means that the FASTA files of the **ul1**, **uL3** and **uL4** families each contain a single-copy sequence belonging to the g_B genome. On the other hand, the FASTA file of the **uL2** family does not contain any g_B genome sequence.

There is a single CSV file constituting the genome index. This file has two columns separated by a comma: **genomeAccession** and **priority_score**.

- **genomeAccession** defines the name of the genomes, as indicated in the primary key of the FASTA files.
- **priority_score** defines the order of priority in the choice of representative MPS genomes. The higher a genome score, the more priority it is chosen. This score constitutes the first priority rule when choosing representative MPS genomes.

Format of intermediate files

The intermediate files are described in the entity-relationship diagram (ERD) in Supplementary Material S13. All files are of CSV type, except the input FASTA files, the TSV files generated by Linclust and two RDS files to respectively export the similarity matrix and hierarchical clustering. FASTA files have already been described above. TSV files are simply CSV files but with a tab as separator instead of a comma (Tabulation-Separated Values). The RDS format is chosen to export the similarity matrix and hierarchical clustering because of its practicality. It allows to quickly write and read data with R as well as compress backups.

Format of output files

The list of the selected MPS representative genomes is accessible in the table **MPS_representatives** containing only a single column: **MPS_representative** the accession number of the MPS representative genomes. The link between the input genomes and the MPS representative genomes is accessible in the table **MPS_links**.

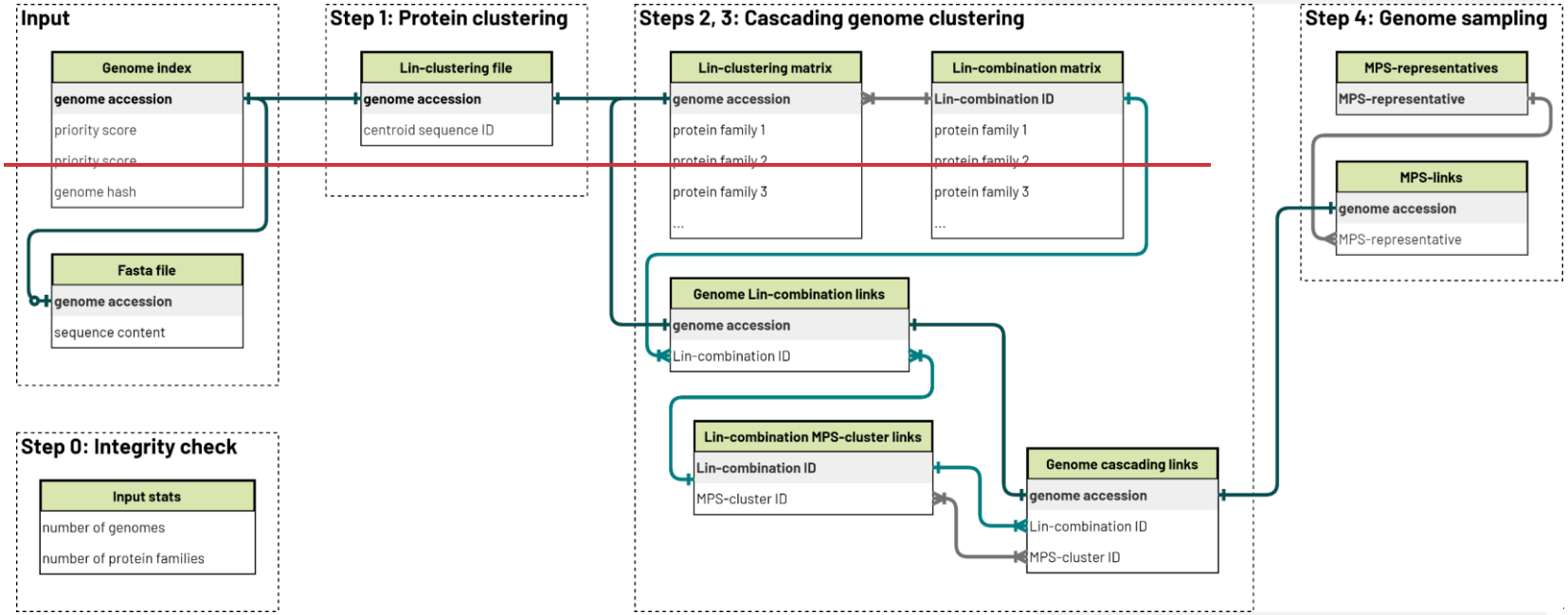
Supplementary Material S12 – Preparation of the bacterial dataset

This flowchart shows the data analysis processed by MPS-Sampling, centralized thanks to a Snakemake pipeline. Each file is colored in light green. The only external program is Linclust and colored in black. All other tasks are carried out by home-made R scripts, colored in strong blue. The R dependencies is mentioned at the right, in white boxes.

Supplementary Material S13 – Entity relationship diagram (ERD) of MPS-Sampling

This entity relationship diagram (ERD) below details the relational schema of MPS-Sampling.

Each table presents a distinct table with its columns. Primary keys are shaded in grey and links between columns are indicated by arrows.
For example, the **Genome index** table has three columns: (i) **genome accession**, which represents the accession number of the genome and is the primary key of the table; (ii) **priority score**, which represents the priority score for choosing MPS-representative genomes; and (iii) genome hash, which contains the hashing value of the accession number of the genome, to separate equivalent choices using a pseudo-random criterion. The primary key **genome accession** is shared with the tables **Fasta files** and **Genome Lin-combination links**. The type of arrows indicates that there is strictly one entry in the **Genome index** table for one entry in the **Genome Lin-combination links** table. This is consistent because each genome belongs strictly to one and only one Lin-combination. On the other hand, for each entry in Genomic index, there is zero or one entry in the Fasta files. This is still consistent because the absence of a sequence in a genome is authorized (zero entry), as is a single-copy sequence (one entry), but duplicated sequences are not accepted (multiple entries forbidden). Conversely, each sequence must have one name and one name only, so each sequence is attached to strictly one genome.



Supplementary Material S12~~14~~ – ~~Trimming~~ Preparation of the bacterial dataset

The initial dataset corresponds to ~~encompassed~~ 55 protein families from 200,565 bacterial genomes from the database RiboDB (Jauffrit *et al.*, 2016). The multi-copy sequences, or duplicated sequences, are discard from the start. ~~along 55 protein families, with~~The datasets includes 9,693,984 single-copy sequences ~~representing~~, with ~~87.88% of the cells~~12.12% of missing values (see ~~Supplementary Material S15~~). ~~First, only protein families present in more than 50% of the genomes were kept, leading to the elimination of 55 - 53 = 2 protein families. Then, bacterial genomes containing more than 80% of the 53 remaining protein families were kept, leading to the elimination of 200,565 - 178,693 = 21,872 genomes. Next, protein families were selected again, keeping only those present in more than 80% of the genomes. And again with the genomes present in more than 80% of the 50 remaining protein families. Now, protein sequences with abnormally short size (i.e. < 75% of the median length within protein families) were excluded from the analysis. This leads to the exclusion of 8,546,835 - 8,334,499 = 212,336 sequences. After two last trimming on the protein families and the genomes, the final trimming of the bacterial dataset encompassed 178,203 genomes along 48 protein families, with 8,315,939 single-copy sequences representing 97,22% of the cells.~~

Code de champ modifié

Each line in the table indicates a filtration step.

- The original set consists of 9,693,984 single-copy protein sequences, spread across 55 protein families and 200,565 genomes, i.e. 88% of possible single copies.
- Only protein families with more than 50% of their sequences reported are retained. 2 families have been discarded, reducing the number of protein families from 55 to 53.
- Only genomes with more than 80% of their sequences complete are retained. 21,872 genomes are discarded, going from 200,565 to 178,693 genomes.
- Only protein families with more than 80% of their sequences complete are retained. 3 families are discarded, reducing the number of protein families from 53 to 50.
- Only genomes with more than 80% of their sequences complete are retained. No genome is discarded at this stage.
- Only sequences whose length is greater than or equal to the median length in the protein family are retained. 86,480 sequences are discarded, reducing the number of genomes from 8,633,315 to 8,546,835.
- Only protein families with more than 80% of sequences are retained. 2 families are discarded, reducing the number of protein families from 50 to 48.
- Only genomes with more than 80% of their sequences known are kept. 18,560 genomes were discarded, reducing the number of genomes from 8,334,499 to 8,315,939.
- The final set consists of 8,315,939 protein sequences, spread across 48 protein families and 178,203 genomes, i.e. 97% of possible unique copies.

The final set represents 178,203 x 48 = 8,553,744 cells. 179,103 of these cells (2%) are empty, i.e. there are 8,553,744 - 179,103 = 8,374,641 filled cells (98%) including 8,436,399 sequences. There are more sequences than cells because there are duplications, i.e. several sequences can correspond to the same cell. Of the 8,374,641 filled cells, 51,026 are duplication cases (0.61%). Of the 8,436,399 sequences, 112,784 are involved in duplication cases (1.34%).

Step	Number of ~~rows~~genomes	Number of ~~columns~~r-prot families	Number of single-copy ~~filled cells~~sequences	~~Ratio of filled cells~~Proportion of missing data
The original set consists of 9,693,984 single copy protein sequences, spread across 55 protein families and 200,565 genomes, i.e. 88% of possible single copies.~~Initial data~~	200,565	55	9,693,984	87.88%
• Only protein families with more than 50% of their sequences reported are retained. 2 families have been discarded, reducing the number of protein families from 55 to 53.~~Column ≥ 50%~~	200,565	53	9,685,194	91.11%
• Only genomes with more than 80% of their sequences complete are retained. 21,872 genomes are discarded, going from 200,565 to 178,693 genomes.~~Row ≥ 80%~~	178,693	53	9,029,460	95.34%
• Only protein families with more than 80% of their sequences complete are retained. 3 families are discarded, reducing the number of protein families from 53 to 50.~~Column ≥ 80%~~	178,693	50	8,633,315	96.63%
• Only genomes with more than 80% of their sequences complete are retained. No genome is discarded at this stage.~~Row ≥ 80%~~	178,693	50	8,633,315	96.63%
• Only sequences whose length is greater than or equal to the median length in the protein family are retained. 86,480 sequences are discarded, reducing the number of genomes from 8,633,315 to 8,546,835.~~Column: Sequence Length ≥ 75% of Median~~	178,693	50	8,546,835	95.66%
• Only protein families with more than 80% of sequences are retained. 2 families are discarded, reducing the number of protein families from 50 to 48.~~Column ≥ 80%~~	178,693	48	8,334,499	97.17%
• The final set consists of 8,315,939 protein sequences, spread across 48 protein families and 178,203 genomes, i.e. 97% of possible unique copies.~~Row ≥ 80%~~	178,203	48	8,315,939	97.22%

There are 179,103 missing data (2%). Additionally to the 8,315,939 single-copy sequences, the 178,203 genomes and the 48 protein families involved 112,784 duplicated sequences (1.34%).

a mis en forme le tableau

a mis en forme : Police :Non Gras

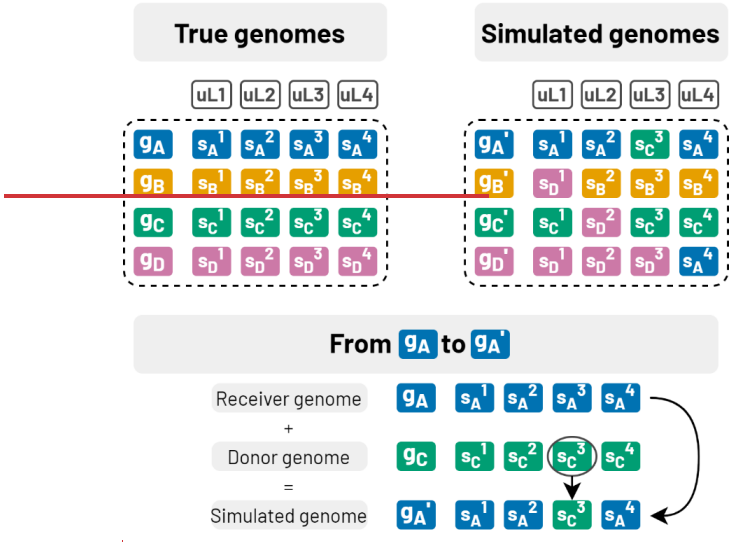
a mis en forme : Police :Non Gras

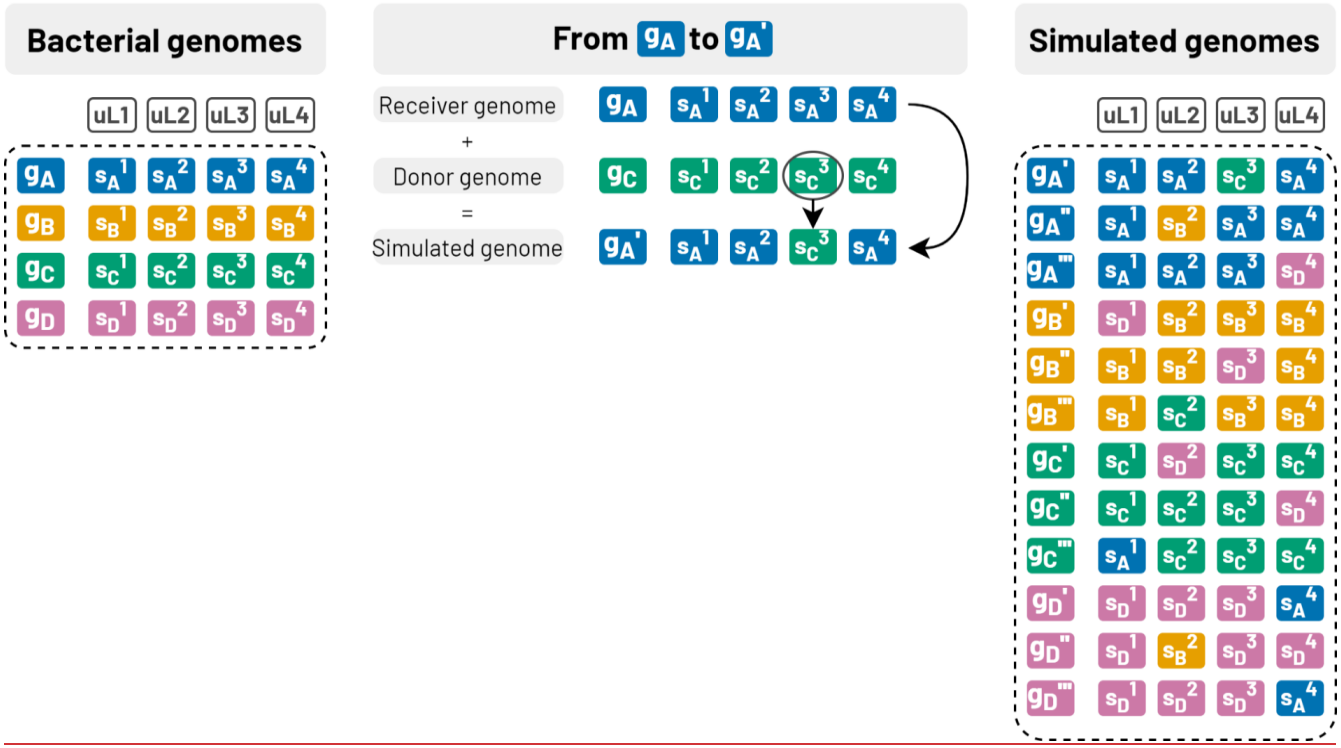
a mis en forme : Justifié, Retrait : Première ligne : 0,5 cm, Espace Après : 6 pt, Interligne : Multiple 1,15 li

~~The bacterial dataset encompassed 178,203 x 48 = 8,553,744 cases (see Supplementary Material S14). 179,103 of these cases (2.09%) were empty, i.e. there were 8,553,744 – 179,103 = 8,374,641 filled cases (97.91%) encompassing 8,436,399 sequences. 51,026 of the 8,374,641 filled cases (0.61%) are duplication cases. 112,784 of the 8,436,399 sequences (1.34%) were involved in duplication cases. Only single-copy sequences were considered, meaning that 112,784 sequences (1.34% of the sequences) were excluded of the analysis. At the end, this represented 8,436,399 – 112,784 = 8,323,615 protein sequences.~~

Supplementary Material S1317 – Generation of the artificial bacterial dataset

The natural bacterial datasetgenomes haves been used to simulate artificial genomes. In the example below, 4 true bacterial genomes ($\mathbf{g_A}$, $\mathbf{g_B}$, $\mathbf{g_C}$, $\mathbf{g_D}$) have been used to generate 4 artificial genomes ($\mathbf{g_A'}$, $\mathbf{g_B'}$, $\mathbf{g_C'}$, $\mathbf{g_D'}$). Each true bacterial genome is used as a receiver and receives a gene from another genome, used as donor. In the example below, the genome $\mathbf{g_A}$ is used as receiver genome and receives the gene $\mathbf{s_C^3}$ through an horizontal gene transfer from the genome $\mathbf{g_C}$ used as receiverdonor. This process is replicates three times per meaning that each genome from the bacterial dataset will be used to generate three artificial genomes ($\mathbf{g_A'}$, $\mathbf{g_A''}$, $\mathbf{g_A'''}$).genome. As a result, the 178,203 x 3 = 534,609 artificial genomes have been generated from the 178,203 natural genomes.


True genomes
Simulated genomes
uL1
uL2
uL3
uL4
From gA to gA'
Receiver genome
+
Donor genome
=
Simulated genome


Bacterial genomes
uL1
uL2
uL3
uL4
From g_A to $g_{A'}$
Receiver genome
+
Donor genome
=
Simulated genome
Simulated genomes
uL1
uL2
uL3
uL4

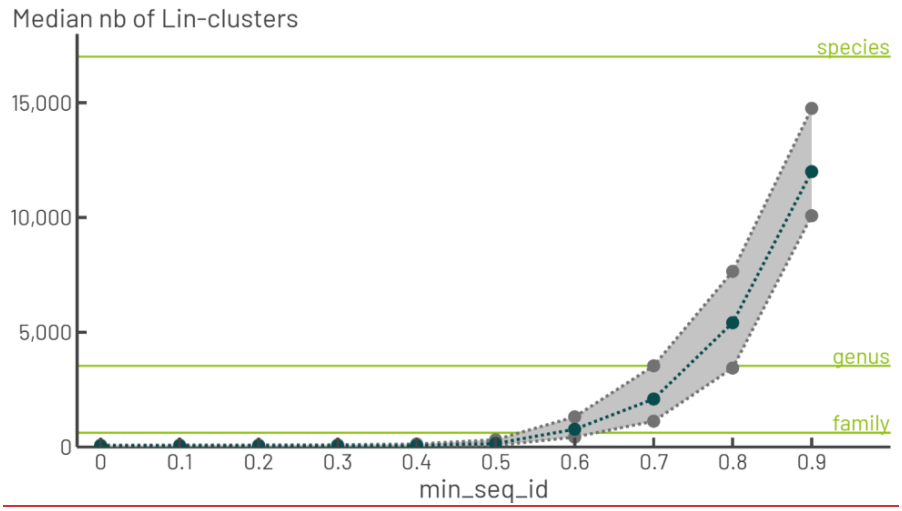
Supplementary Material S14~~18~~ -- ~~Optimization~~ Choice of the parameters of MPS--Sampling (without pre-connection)

For the construction of the Lin-clusters (~~s~~Step 1), Linclust parameters were set as follow: eValue = 10^{-5}, coverageMode = 0, minCov = 0.8, and minSeqID = 0.6. This corresponded to the code for Linclust: "--e 0.00001 --cov--mode--0 --c 0.8 –min--seq--id 0.6". The coverage mode 0 corresponded to a bidirectional coverage between query and target. The step involving Linclust lasted 1 minute without parallelization (Supplementary Material S24~~Supplementary Material S26~~) and could be automatically parallelized with the Snakemake pipeline if needed. ~~Thus, it was possible to test various sets of parameters. Here, t~~The Linclust parameters were ~~optimized~~ tested. ~~to generate a number of Lin-clusters between 2% and 10% of the bacterial dataset, which corresponds to 3,000 and 17,000 genomes, respectively. Regarding our tests, o~~Only the minimum sequence identity seems to impact significantly the number of Lin--clusters (Supplementary Material S15 – Median number of Lin-clusters according to minimum sequence identity

Univariate study of the median number of Lin-clusters according to the minimal sequence identity (**minSeqID**) with the fixed parameters: **covMode = 0**, **eValue = 10e-5** and **minCov = 0.8**.

Three lines represent the median number (blue) as well as the first and third quartiles (grey) of Lin-clusters. As an indication, three green lines represent the number of species, genera and families present in the dataset.

The minimum sequence identity had clearly an important impact on the cluster finesse and constitutes the major parameter to fine-tune for Linclust. By fixing it to **minSeqID = 0.6**, the median number of Lin-clusters (773) approached the number of taxonomic families (661).



a mis en forme : Normal

a mis en forme : Police :Non Gras

Supplementary Material S16-S22). This value of minSeqID = 0.6 brought a median of 773 Lin-clusters along the 48 proteins (), which represent roughly the number of taxonomic families of 661. Based on these parameters, within each Lin-cluster, protein sequences were expected to share at least half of their amino acid positions with the centroid sequences (minSeqID x minCov = 0.60 x 0.80 = 0.48). To resume, Linclust parameters were set as follow: eValue = 10^{-5}, coverageMode = 0, minCov = 0.8, and minSeqID = 0.6.

For the construction of the Lin-combinations and the elementary groups of genomes (EGG) (Step 2), there is no parameter to adjust the construction of the EGG.

For
pre-connection (Step 3), this step was skipped for the main run of MPS-Sampling so no parameter
pre-connection (Step 3), this step was skipped for the main run of MPS-Sampling so no parameter
pre-connection (Step 3), this step was skipped for the main run of MPS-Sampling so no parameter
pre-connection (Step 3), this step was skipped for the main run of MPS-Sampling so no parameter
pre-connection (Step 3), this step was skipped for the main run of MPS-Sampling so no parameter
pre-connection (Step 3), this step was skipped for the main run of MPS-Sampling so no parameter
pre-connection (Step 3), this step was skipped for the main run of MPS-Sampling so no parameter
pre-connection (Step 3), this step was skipped for the main run of MPS-Sampling so no parameter
pre-connection (Step 3), this step was skipped for the main run of MPS-Sampling so no parameter
pre-connection (Step 3), this step was skipped for the main run of MPS-Sampling so no parameter
pre-connection (Step 3), this step was skipped for the main run of MPS-Sampling so no parameter

~~The two figures are coherent because it confirms that, the greater the minNbLinclusters was, the more the overall genomic dataset was divided. Regarding the size of the largest pre-connected component, a first platoon can be seen until 24 (Supplementary Material S23), with the largest pre-connected component comprising most of the dataset, it means that the pre-connection was not able to separate it into several parts. Then the curve is decreasing, with two other observable platoons. Parallelly, the number of pre-connected components was increasing as minNbLinclusters increased, mostly at the end (Supplementary Material S24). For the bacterial dataset, the pre-connection was adjust with the optimal value minNbLinclusters = 25. This leaded to 488 pre-connected components. This means there will be at least 488 MPS-representatives. The size of the pre-connected components whose size varied comprised from 1 to 79,145 genomes. 25 shared Lin-clusters over 48 (≈ 0.5) are required to be pre-connected, which corresponds to a minimal similarity of Δ = 0.5. It means that all similarities from 1 and 0.50 were captured by pre-connection and that the minimum similarity Δ of the complete-linkage was constrained between Δ = 1 and Δ = 0.5. However, Δ could decreases slightly below Δ, down to Δ = 0.4, to accentuate the dereplication of areas containing many relatively close genomes.~~pre-connection (Step 3), this step was skipped for the main run of MPS-Sampling so no parameter needed to be chosen.

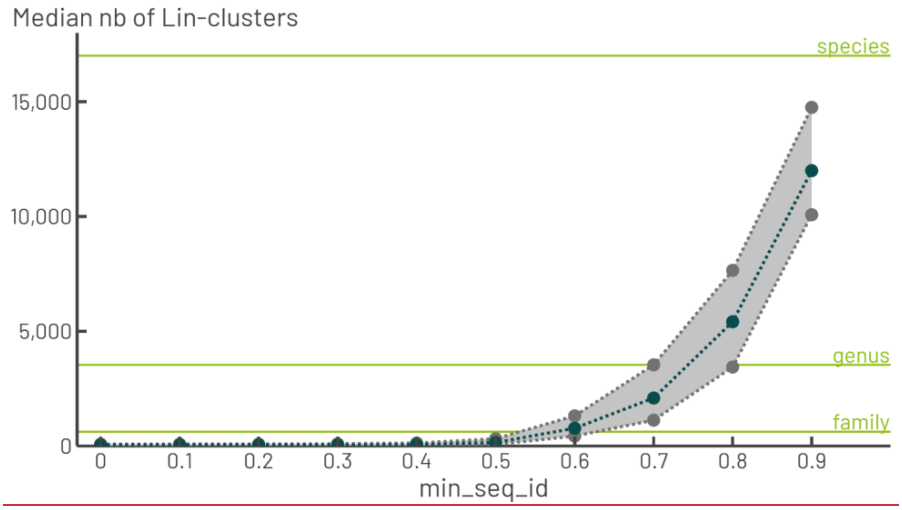
For the construction of the MPS-clusters (Step 4), the minimum similarity Δ was set to eleven different values $\Delta \in \{1.00, 0.90, 0.80, 0.70, 0.60, 0.50, 0.40, 0.30, 0.20, 0.10, 0.05\}$, leading to eleven different runs and thus to seven samplings of different sizes.

Supplementary Material S15 – Median number of Lin-clusters according to minimum sequence identity

Univariate study of the median number of Lin-clusters according to the minimal sequence identity (**minSeqID**) with the fixed parameters: **covMode = 0**, **eValue = 10e-5** and **minCov = 0.8**.

Three lines represent the median number (blue) as well as the first and third quartiles (grey) of Lin-clusters. As an indication, three green lines represent the number of species, genera and families present in the dataset.

The minimum sequence identity had clearly an important impact on the cluster finesse and constitutes the major parameter to fine-tune for Linclust. By fixing it to **minSeqID = 0.6**, the median number of Lin-clusters (773) approached the number of taxonomic families (661).



a mis en forme : Police :Non Gras

Supplementary Material S16 – Median number of Lin-clusters according to coverage mode

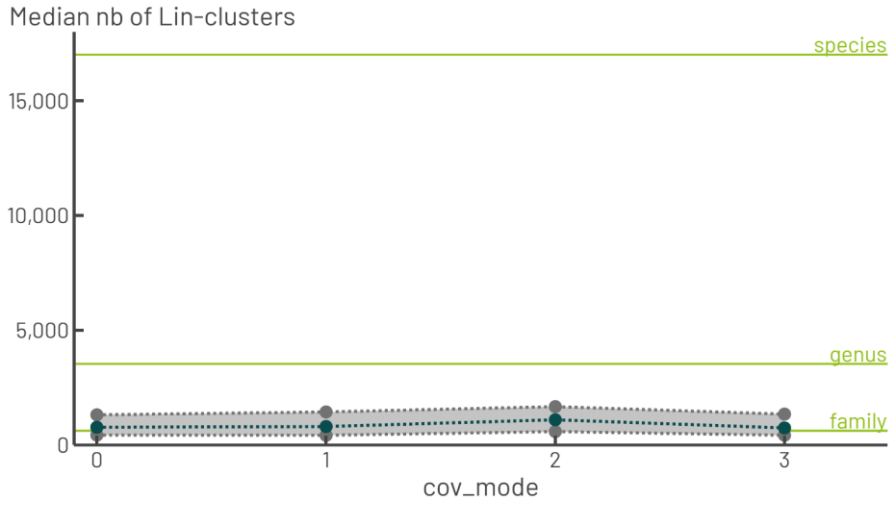
Univariate study of the median number of Lin-clusters according to the coverage mode (**covMode**) with the fixed parameters: **eValue = 10e-5**, **MinCov = 0.8** and **minSeqID = 0.6**.

with the fixed parameters: **eValue = 10e--5**, **MinCov = 0.8** and **minSeqID = 0.6**.

Three lines represent the median number (blue) as well as the first and third quartiles (grey) of Lin-clusters. As an indication, three green lines represent the number of species, genera and families present in the dataset.

Three lines represent the median number (blue) as well as the first and third quartiles (grey) of Lin--clusters. As an indication, three green lines represent the number of species, genera and families present in the dataset.

The line is almost flat, so the coverage mode did not have a large impact on the sequence cluster finesse.



Supplementary Material S17 – Median number of Lin-clusters according to eValue

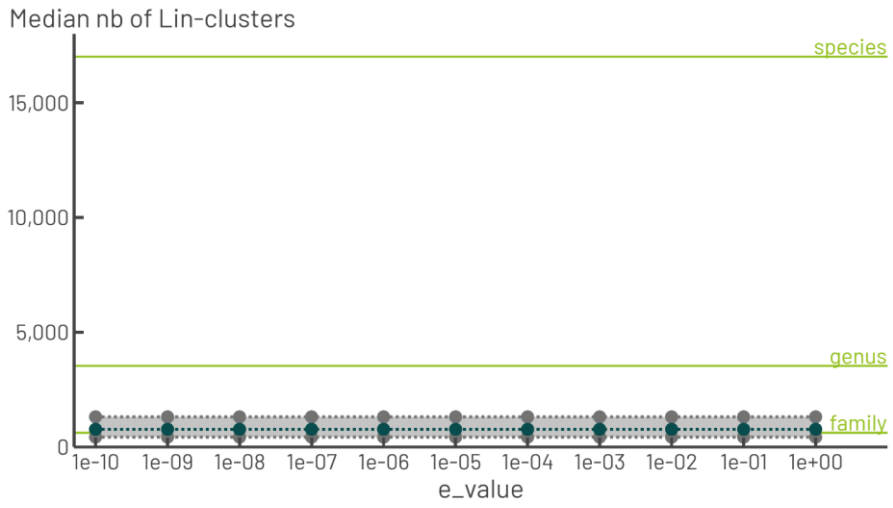
Univariate study of the median number of Lin-clusters according to the **eValue** with the fixed parameters: **covMode = 0**, **minCov = 0.8** and **minSeqID = 0.6**.

with the fixed parameters: **covMode = 0**, **minCov = 0.8** and **minSeqID = 0.6**.

Three lines represent the median number (blue) as well as the first and third quartiles (grey) of Lin-clusters. As an indication, three green lines represent the number of species, genera and families present in the dataset.

Three lines represent the median number (blue) as well as the first and third quartiles (grey) of Lin--clusters. As an indication, three green lines represent the number of species, genera and families present in the dataset.

The line is completely flat, so the eValue had almost no impact on the sequence cluster finesse.



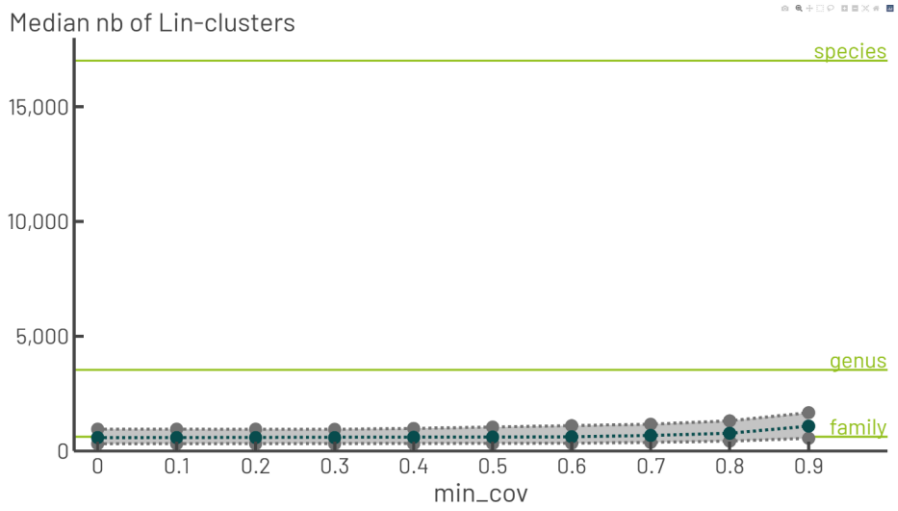
Supplementary Material S18 – Median number of Lin-clusters according to minimum coverage

Univariate study of the median number of Lin-clusters according to the minimal coverage (**minCov**) with the fixed parameters: **covMode = 0**, **eValue = 10e-5** and **minSeqID = 0.6**.

with the fixed parameters: **covMode = 0**, **eValue = 10e-5** and **minSeqID = 0.6**.

Three lines represent the median number (blue) as well as the first and third quartiles (grey) of Lin-clusters. As an indication, three green lines represent the number of species, genera and families present in the dataset.

The minimum coverage had almost no impact from 0.3 to 0.7. From 0.8 to 0.9, it had a little impact.

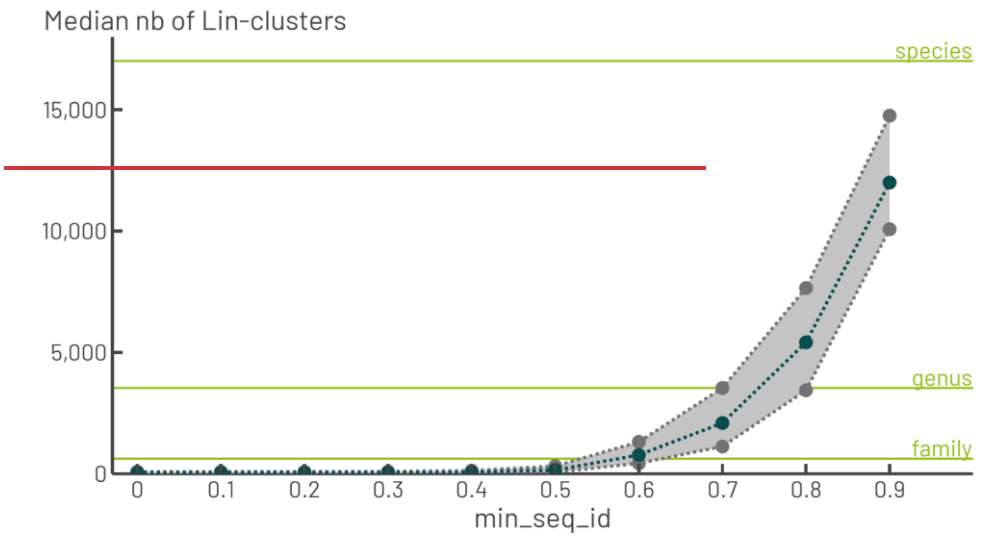


~~**Supplementary Material S22 – Median number of Lin-clusters according to minimum sequence identity**~~

~~Univariate study of the median number of Lin-clusters according to the minimal sequence identity (**minSeqID**) with the fixed parameters: **covMode = 0**, **eValue = 10e-5** and **minCov = 0.8**.~~

~~Three lines represent the median number (blue) as well as the first and third quartiles (grey) of Lin-clusters.~~

~~The minimum sequence identity had clearly an important impact on the cluster finesse and constitutes the major parameter to fine-tune for Linclust. By fixing it to **minSeqID = 0.6**, the median number of Lin-clusters (773) approached the number of taxonomic families (661).~~


Median nb of Lin-clusters
species
15,000
10,000
5,000
genus
family
0
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
min_seq_id

Supplementary Material S19 – Choice of the parameters of MPS-Sampling (for pre-connection)

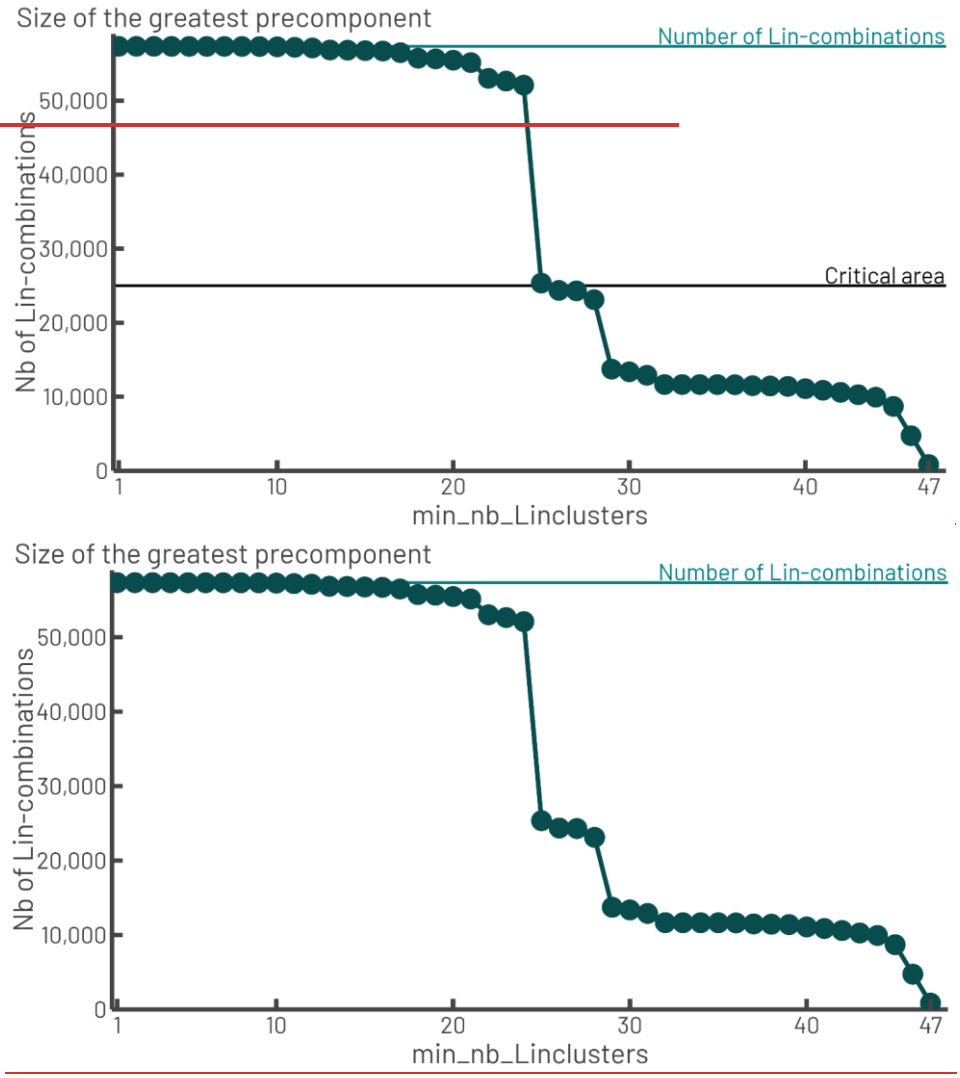
When using the pre-connection (Step 3), no other parameter was changed, as described in the Supplementary Material S14. For step 3, the pre-Connection was very fast (2 minutes) (Supplementary Material S24).

The parameter was set to **MinNbLinclusters = 25**. It leaded to trade-off between the number of pre-connected components (488) and the size of the largest pre-connected component, which encompassed 25,351 Lin-combinations

Supplementary Material S20 – Size of the largest pre-connected component depending on MinNbLinclusters

The decreasing of the curve is coherent because the greater the MinNbLinclusters, the more the overall genomic dataset was divided and the smaller the largest pre-connected component.

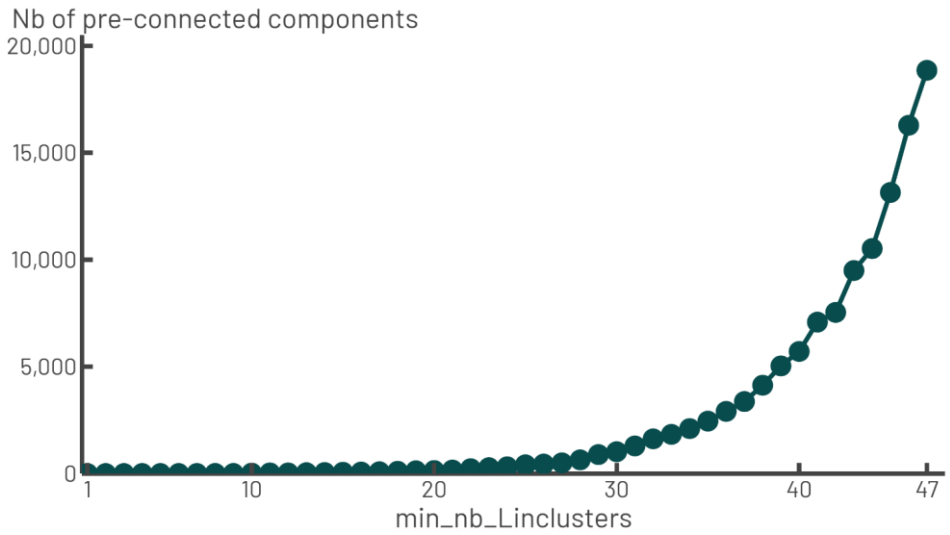
~~The parameter was set to **MinNbLinclusters = 25**, the smallest value according to the critical area. The largest pre-connected component encompassed 25,351 Lin-combinations, a little above the critical area of 25,000.~~



Supplementary Material S21 – Number of pre-connected components depending on MinNbLinclusters

The decreasing of the curve is coherent because the greater the MinNbLinclusters, the more the overall genomic dataset was divided, and the more pre-connected components there were.

~~The parameter was set to **MinNbLinclusters = 25**, leading to 488 pre-connected components.~~


Nb of pre-connected components
20,000
15,000
10,000
5,000
0
1
10
20
30
40
47
min_nb_Linclusters

Supplementary Material S22 - Phylogenetic reconstruction

For phylogenetic inference, for each of the 48 studied r-prot families, protein sequences were aligned with kalign v3.2.2 (March 22) (Lassmann, 2020) with default parameters. The resulting alignments were trimmed by removing positions containing more than 30% of gaps and sequences with more than 30% of gaps. Trimmed alignments were combined to build a large supermatrix. A second trimming round has been performed on the supermatrix to remove the columns of positions containing more than 20% of gaps and sequences containing more than 20% of gaps. These filters are used to limit the amount of missing data in the supermatrices that may bias phylogenetic reconstructions (Philippe *et al.*, 2017). Phylogenies were reconstructed using FastTree v2.1.11 (Feb 20) (Price *et al.*, 2010) with default parameters. The phylogeny inference of the 178,203 genomes of the bacterial dataset lasted 51.50 hours with a parallelization using 3 cores on a server with 32 cores and 64 threads (AMD EPYC 7542 32 Core Processor @3,40Ghz) and 1 To of DDR4 under Debian Trixie.

Supplementary Material S23 – TaxSampler, a homemade-software for sampling genomes based on taxonomy

TaxSampler is an unpublished homemade-software that samples genomes according to their taxonomic affiliation. The taxonomic affiliation that has been used comes from the NCBI. It can be applied to any taxonomic level: phylum, class, order, family, genus, or species. TaxSampler uses as input a list of genomes with taxonomy and provides as output a list of representative genomes for the chosen taxonomic level, called Tax-representatives.

- TaxSampler uses as input a list of genomes with taxonomy.
- TaxSampler discards genomes whose taxonomic affiliation of the chosen taxonomic level is unknown.
- TaxSampler groups genomes according to the taxonomic affiliation of the chosen taxonomic level.
- TaxSampler choses a representative genome, called Tax-representative, within each group. This Tax-representative is chosen according to the first and fourth priority rules of MPS-Sampling (Supplementary Material S10). First, the genomes with the highest priority score are chosen. Second, the genome with the smallest hashing value is chosen.

Supplementary Material S24 – Computational time MPS-Sampling concerning the 178,027 genomes of the bacterial dataset

MPS-sampling was run without pre-connection on a single-threaded process. The involved machine was AmalphyLab, a shared server with 32 cores and 64 threads (AMD EPYC 7542 32 Core Processor @3,40Ghz) and 1 To of DDR4 under Debian Trixie. The computational time is measured with elapsed time and has been rounded to the minute.

The computational time of one MPS-sample was 4 + 1 + 51 + 1 + 2 = 59 min. Because the steps 1 to 4-1 are common to several MPS-samples, the computational time for eleven MPS-samples was 4 + 1 + 51 + 1 + (1 + 3) * 11 = 101 min = 1h41.

The most time-consuming step was the computing of similarity submatrices from pre-connected components (step 4-1) due to quadratic complexity. This limitation was partially relaxed thanks to pre-connection, saving a substantial amount of time (see Discussion).

Algorithm	Step	Task	Computational time
--	--	Retrieval and download of ribosomal sequence families from RiboDB website	3 min
--	--	Filter genomes and ribosomal protein families	5 min
--	--	Format FASTA files	4 min
MPS--Sampling	Step 1	Construction of Lin--clusters	4 min
MPS--Sampling	Step 2--1	Labelling of Lin--clusters	1 min
MPS--Sampling	Step 2--2	Construction of the Lin--clustering matrix	
MPS--Sampling	Step 2--3	Re--ordering of the Lin--clustering matrix	
MPS--Sampling	Step 2--4	Construction of the Lin--combination matrix	
MPS--Sampling	Step 3	None (skipped step)	
MPS--Sampling	Step 4--1	Computation of the similarity matrix	51 min
MPS--Sampling	Step 4--2	Construction of MPS--clusters	1 min
MPS--Sampling	Step 5	Selection of MPS--representatives	2 min

Supplementary Material S2527 – Computational time

Here are four studies about MPS--Sampling and computational time. In these studies, MPS-Sampling was always launched to generate one sample with $\Delta = 0.4$.

A: Some subsets of the~~The~~ bacterial dataset (-178,203 genomes) have been analyzed on two different machines:

- **PkDBServ:** A dedicated server with 20 cores and 40 threads (2 Intel Xeon E5 2660v2 CPUs @2.20Ghz) and 128 GB of DDR3 RAM running Debian 10 (Buster~~)~~).
- **AmalphyLab:** A shared server with 32 cores and 64 threads (AMD EPYC 7542 32 Core Processor @3.40Ghz) and 1 TB of DDR4 RAM running Debian Trixie.

The difference is still modest. With the 178,203 genomes, the run was 30% slower with PkDBServ than with AmalphyLab (respectively 4,520 secs and 3,472 secs).

B: Some subsets of the bacterial dataset (178,203 genomes) have been analyzed with and without pre-connection. With pre-connection, the analysis was much faster. With the 178,203 genomes, the run was 70% faster with pre-connection than without (respectively 1,030 secs and 3,472 secs).

C: MPS--Sampling and Treemmer were compared. Treemmer was much slower than MPS--Sampling. With the 178,203 genomes, Treemmer lasted 360h, that was 360 times slower than MPS--Sampling.

~~**C:**~~

D: The running elapsed time was measured up to 534,609 ABD genomes. It was compared with the running time of the 178,203 true genomes.

In the top left--hand corner, a zoom between 0--150k genomes shows that the running time was similar between the true genomes and the ABD genomes.

The relation between the number of ABD genomes and the running time was estimated. According to a regression model, the relation was $T = 1.30 * 10^7 * N$ where T is the computational elapsed time in seconds and N the number of analyzed genomes (p--value <2e--16).

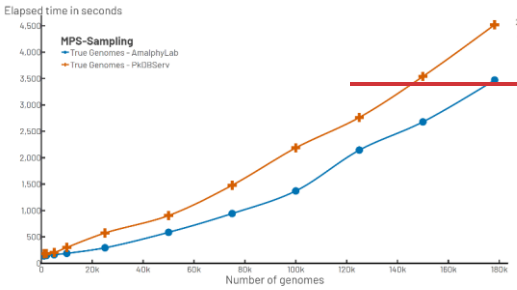
~~**E:** The 178,203 genomes have been analyzed with and without pre-connection.~~

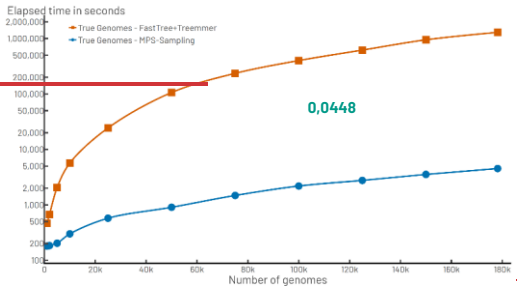
~~With pre-connection, the analysis was much faster. With the 178,203 genomes, the run was 70% faster with pre-connection than without (respectively 1,030 secs and 3,472 secs).~~

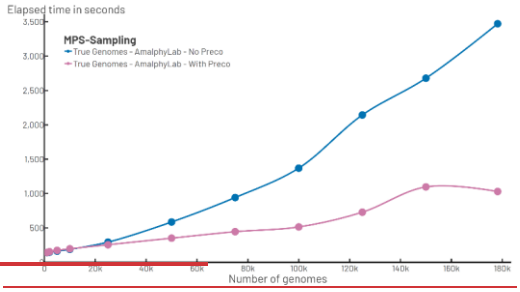
a mis en forme : Retrait : Gauche : 0,5 cm, Sans numérotation ni puces

a mis en forme : Retrait : Gauche : 0,5 cm, Sans numérotation ni puces

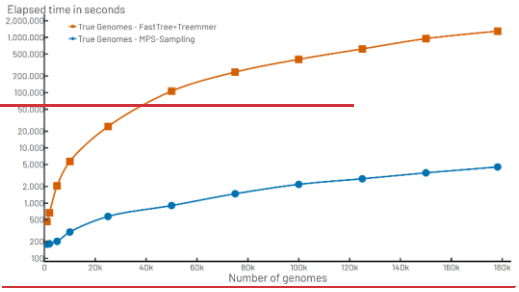
A **B**


Elapsed time in seconds
MPS-Sampling
True Genomes - AmalphyLab
True Genomes - PkDBServ
Number of genomes

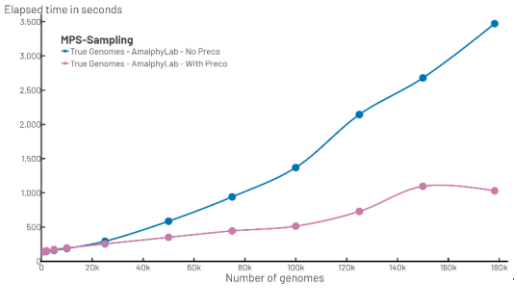

Elapsed time in seconds
True Genomes - FastTree+Treemmer
True Genomes - MPS-Sampling
0,0448
Number of genomes

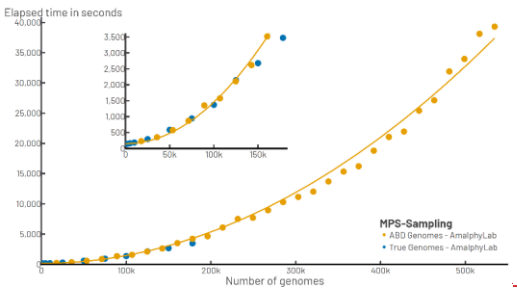

Elapsed time in seconds
MPS-Sampling
True Genomes - AmalphyLab - No Preco
True Genomes - AmalphyLab - With Preco
Number of genomes

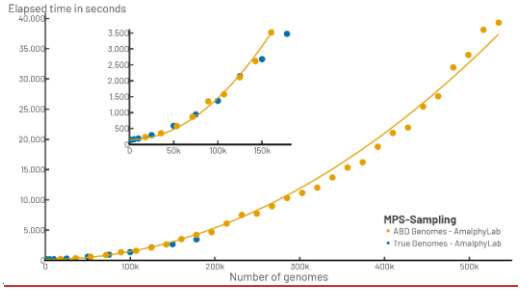
C


Elapsed time in seconds
True Genomes - FastTree+Treemmer
True Genomes - MPS-Sampling
Number of genomes

D


Elapsed time in seconds
MPS-Sampling
True Genomes - AmalphyLab - No Preco
True Genomes - AmalphyLab - With Preco
Number of genomes


Elapsed time in seconds
MPS-Sampling
ABD Genomes - AmalphyLab
True Genomes - AmalphyLab
Number of genomes


Elapsed time in seconds
MPS-Sampling
ABD Genomes - AmalphyLab
True Genomes - AmalphyLab
Number of genomes

Supplementary Material S2628 – Intermediate results of MPS-Sampling concerning the bacterial dataset

Starting from the 178,203 bacterial genomes, 59 to 2,789 Lin-clusters per r-prot family were built (median = 773) (step 1, Supplementary Material S27Supplementary Material S29). This reflects sequence variation (e.g., selective pressure, sequencingsequencing, or assembly errors) among r-prot families: the higher the identity among sequences within a protein family, the smaller the number of Lin-clusters. The number of Lin-clusters was not correlated with the length of the protein sequences (Supplementary Material S28Supplementary Material S30). These Lin-clusters constituted 57,332 Lin-combinations and thus as many EGG, from which 48,296 (84,8%) were singleton (i.e. contain a single genome) (step 2, Supplementary Material S29Supplementary Material S31). In contrast, the three largest encompassed 4.21% of initial dataset (i.e. 2,891, 2,575, and 2,037 genomes, respectively).

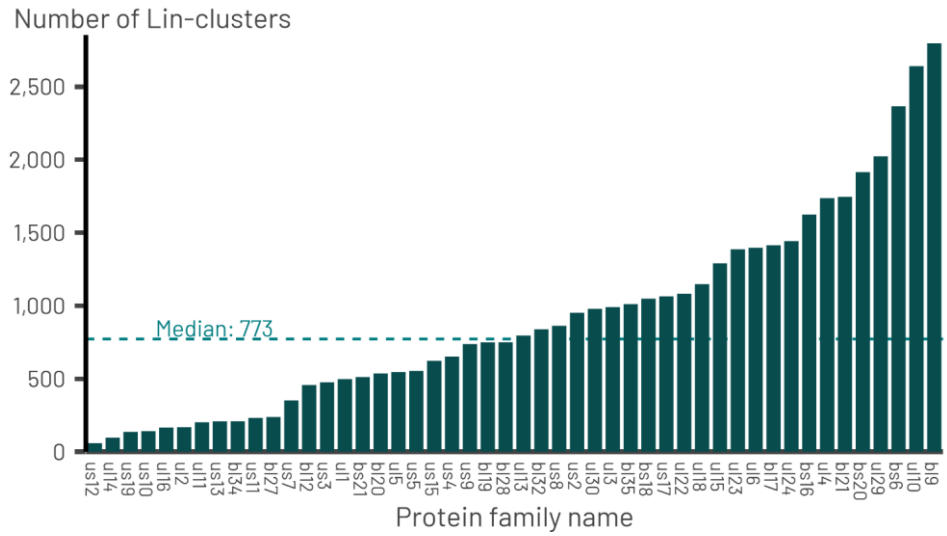
From these Lin-combinations, 488 pre-connected components were built (step 3, Supplementary Material S30Supplementary Material S32) whose size varied greatly: 163 pre-connected components (46%) contained a single genome, while the two largest encompassed together 150,858 genomes and 57,332 Lin-combinations, respectively 84.66% and 76.48% of the bacterial dataset.

Then, MPS-clusters were computed (step 4). At this stage, the higher the value of Δ, the lower the number of MPS-clusters: from 57,332 (Δ = 1) to 3,474 (Δ = 0.4). This corresponded to a sample from 32.17% to 1.95% of the complete bacterial dataset (Figure 2Figure 2A). As an example, when Δ = 0.7, 12,775 MPS-clusters were built (Supplementary Material S31Supplementary Material S33), among which 5,329 contained a single genome, 2,404 two genomes, while the two largest gather 13,820 and 3,385 genomes, respectively.

The process ended with the selection of the MPS-representatives according to priority rules (step 4). The frequency of each criteria use is shown in the Supplementary Material S32Supplementary Material S34. The most decisive criteria were fame, protein distribution, centrality and pseudo-randomness, respectively.

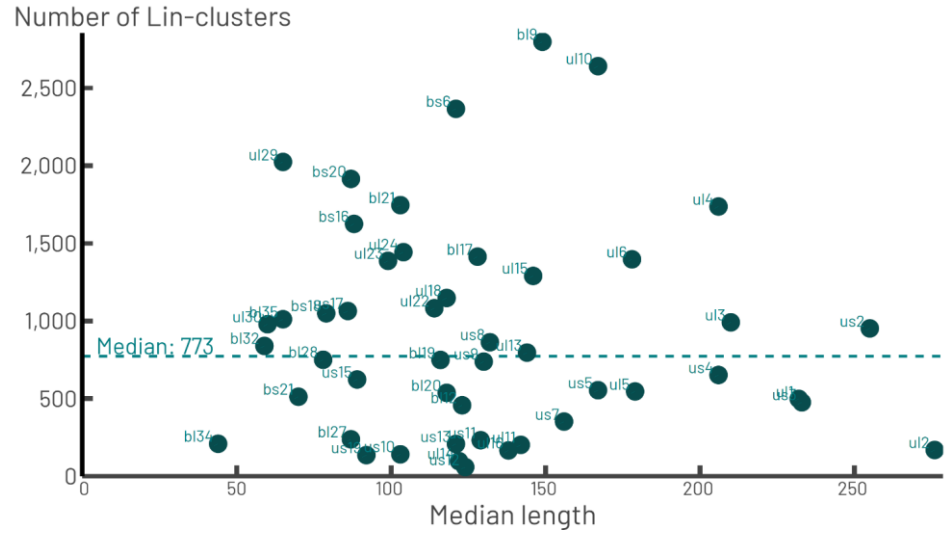
Supplementary Material S2729 – Number of Lin--clusters per protein family

During the analysis of the bacterial dataset, the number of Lin--clusters varied from 59 to 2,798 with a median of 773.



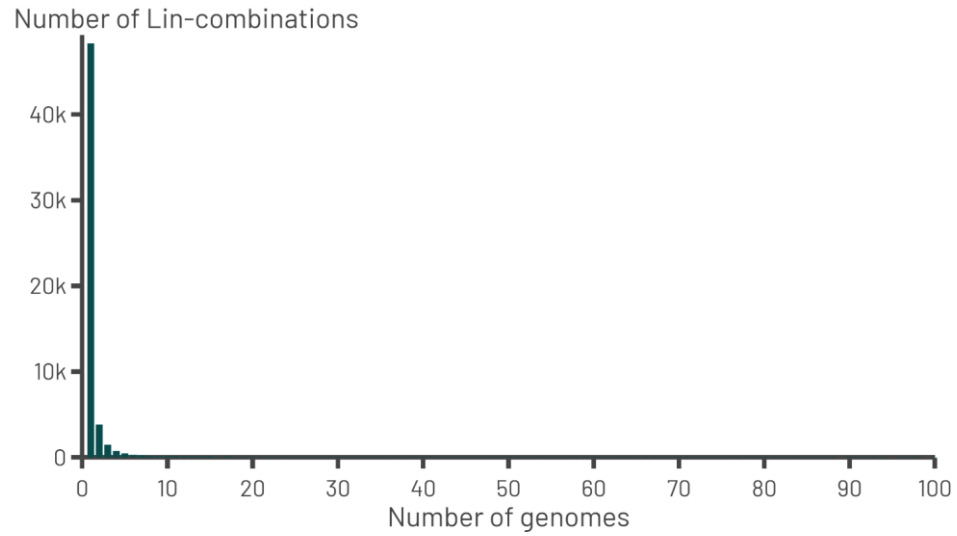
Supplementary Material S2830 – Median length of protein sequences VS Number of Lin--clusters

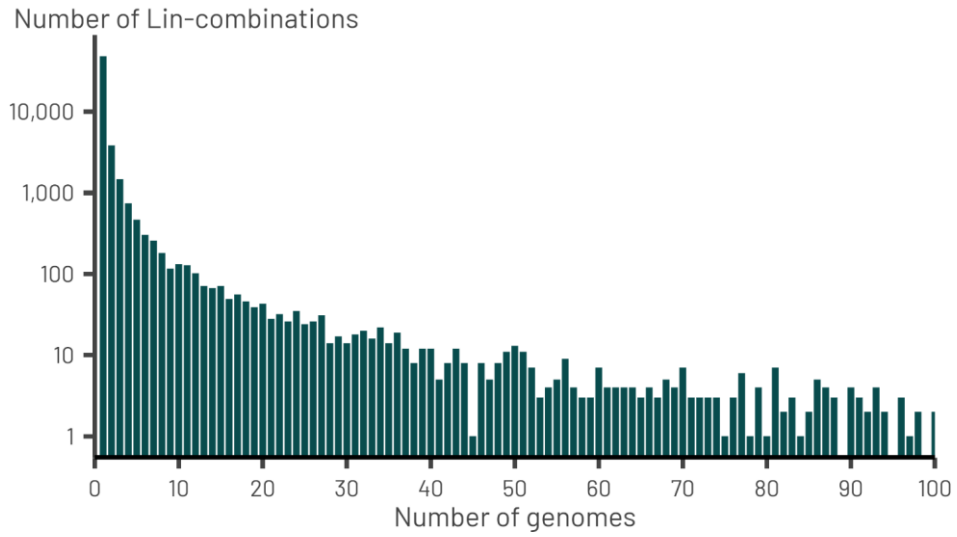
During the analysis of the bacterial dataset, along the 48 considered r--prot families, the number of Lin--clusters was not correlated with the median length of protein sequences.



Supplementary Material S2931 – Number of genomes within the 57,332 Lin-combinations

During the analysis of the bacterial dataset, 57,332 Lin-combination were built. 48,296 Lin-combinations admitted only one genome. On the contrary, the three largest Lin-combinations contained respectively 2,891, 2,575 and 2,037 genomes. 198 Lin-combinations have more than 100 genomes and are "out of the plot" on the right.
The second histogram has a logarithmic scale on the y-axis.

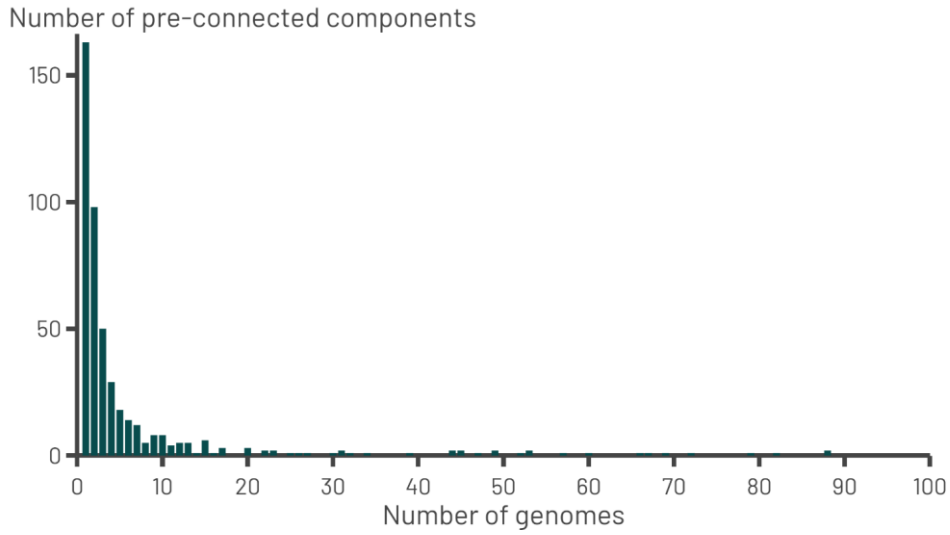


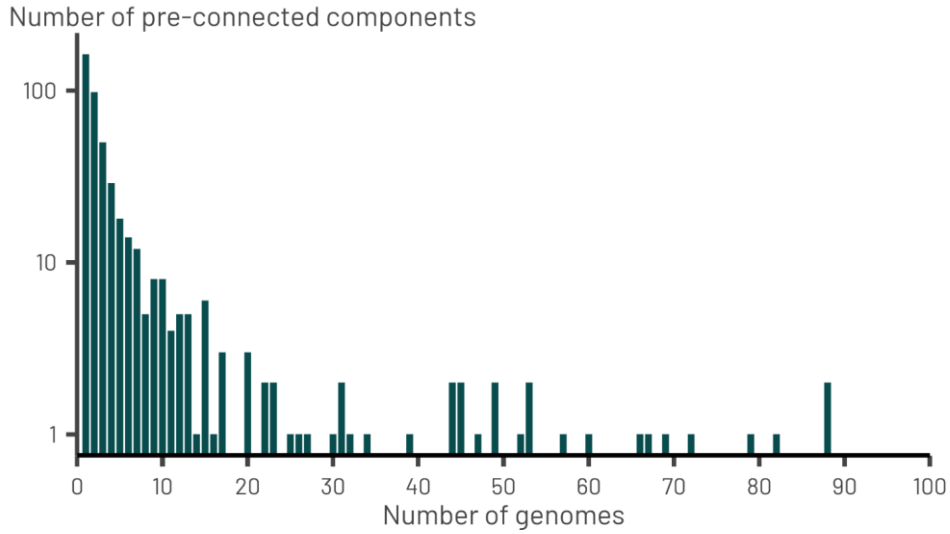


Supplementary Material S3032 – Number of genomes within the 488 pre-connected components

During the analysis of the bacterial dataset, 488 pre-connected components were built. 163 pre-connected components admitted only one genome. On the contrary, the two largest pre-connected components contained 79,145 and 71,713 genomes respectively, encompassing 150,858 out of the 178,203 genomes of the bacterial dataset (84.66%). These two largest pre-connected components contained 25,351 and 18,499 Lin-combinations respectively, encompassing 43,850 out of the 57,332 Lin-combinations of the bacterial dataset (76.48%). 22 pre-connected components has more than 100 genomes and are "out of the plot" on the right.

The second histogram has a logarithmic scale on the y-axis.

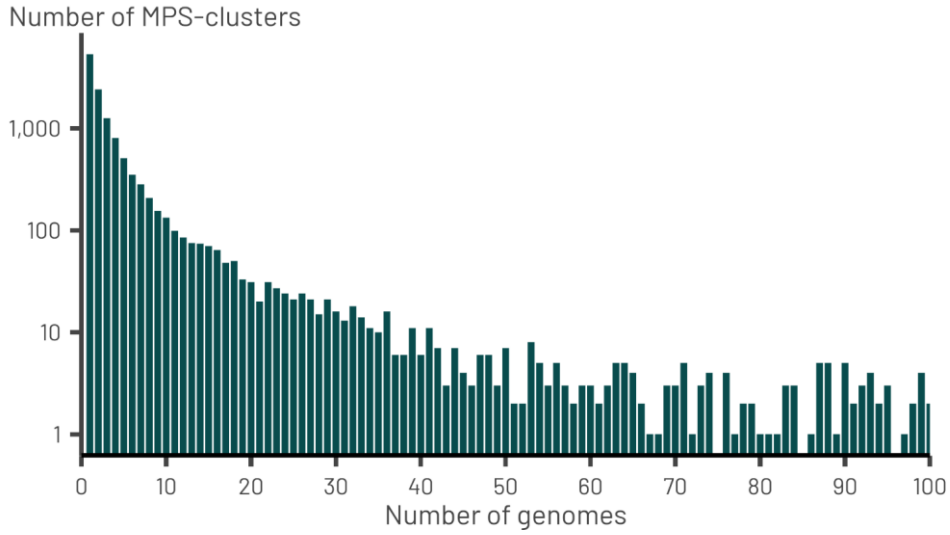


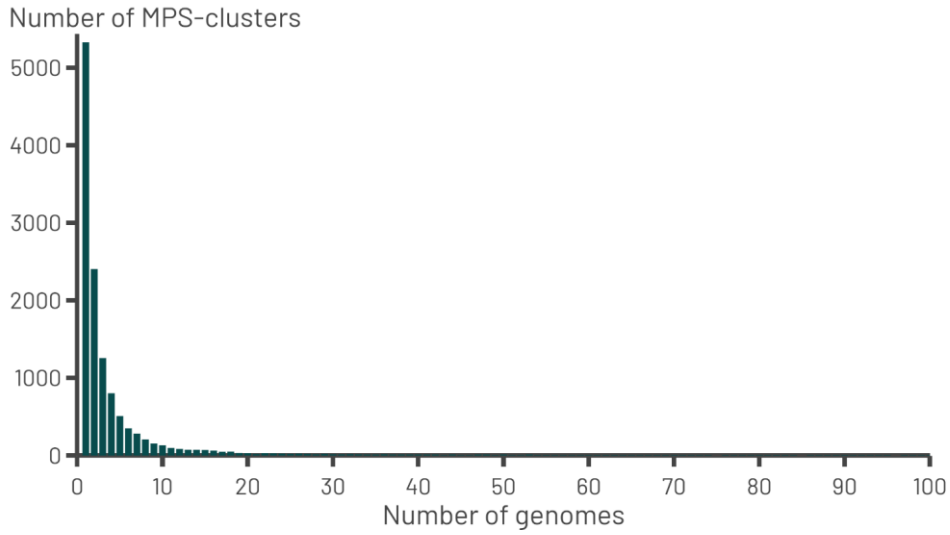


Supplementary Material S3133 – Number of genomes within the 12,775 MPS-clusters (Δ=0.7)

During the analysis of the bacterial dataset, 12,775 MPS-clusters were built using Δ=0.7. 5,329 MPS-clusters admitted only one genome and 2,404 have only two genomes. On the contrary, the two largest MPS-clusters contained respectively 13,820 and 3,385 genomes and respectively 563 and 367 Lin-combinations. 194 MPS-clusters have more than 100 genomes and are "out of the plot" on the right.

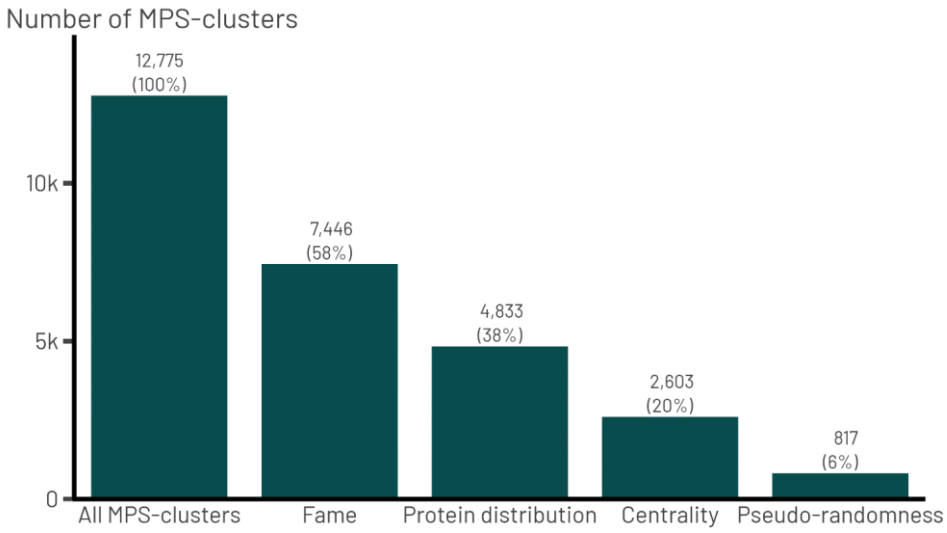
The second histogram has a logarithmic scale on the y-axis.





Supplementary Material S3234 – Number of MPS-clusters where each selection rule was applied

With Δ=0.7, 12,775 MPS-clusters were generated. Among them, 12,775 – 7,446 = 5,329 MPS-clusters contain only one genome, so no selection rule needed to be applied. It meant that 7,446 MPS-clusters had more than one genome. The first rule (fame) was applied to all these 7,446 MPS-clusters. After that, 4,833 MPS-clusters still had more than one genome, so the second rule (protein distribution) was applied to these 4,833 MPS-clusters. Then, the third rule (centrality) was applied to 2,603 MPS-clusters. Finally, the last rule (pseudo-randomness) was applied to 817 MPS-clusters.



Supplementary Material S35 – Reduction of the whole *Bacteria* dataset using MPS-Sampling

Commenté [CRV43]: À supprimer ?

	Complete dataset	MPS-Sampling runs						
		Δ=1	Δ=0.9	Δ=0.8	Δ=0.7	Δ=0.6	Δ=0.5	Δ=0.4
Genomic size								
Number of genomes	178,203	57,332	30,196	19,117	12,775	8,352	5,347	3,474
	100%	32.17%	16.94%	10.73%	7.17%	4.69%	3.00%	1.95%
Number of genomes with complete taxonomic affiliation	135,315	28,641	12,634	7,698	4,888	3,033	1,872	1,205
	100%	21.17%	9.34%	5.68%	3.61%	2.24%	1.38%	0.89%
Number of genomes with partial taxonomic affiliation	42,888	28,691	17,562	11,419	7,887	5,319	3,475	2,269
	100%	66.90%	40.95%	26.63%	18.39%	12.40%	8.10%	5.29%
Number of genomes with no taxonomic affiliation	44	41	35	30	23	21	14	9
	100%	93.18%	79.55%	68.18%	52.27%	47.73%	31.82%	20.45%
Number of genomes with no phylum affiliation	86	82	70	56	42	36	26	17
	100%	95.35%	81.40%	65.12%	48.84%	41.86%	30.23%	19.77%
Number of genomes with no class affiliation	5,619	4,897	3,620	2,651	2,001	1,488	1,084	817
	100%	87.15%	64.42%	47.18%	35.61%	26.48%	19.29%	14.54%
Number of genomes with no order affiliation	9,507	8,726	6,542	4,835	3,697	2,720	1,822	1,393
	100%	91.79%	68.81%	50.86%	38.89%	28.61%	20.22%	14.65%
Number of genomes with no family affiliation	13,616	12,554	9,437	6,932	5,255	3,817	2,668	1,844
	100%	92.20%	69.31%	50.91%	38.59%	28.03%	19.59%	13.54%
Number of genomes with no genus affiliation	18,663	17,344	12,995	9,387	6,905	4,815	3,175	2,079
	100%	92.93%	69.63%	50.30%	37%	25.80%	17.01%	11.14%
Number of genomes with no species affiliation	41,752	28,077	17,161	11,120	7,663	5,157	3,347	2,172
	100%	67.25%	41.10%	26.63%	18.35%	12.35%	8.02%	5.20%
Number of genomes with at least one Candidatus entry	2,991	2,621	1,723	1,256	946	698	526	431
	100%	87.63%	57.61%	41.99%	31.63%	23.34%	17.59%	14.41%
Number of genomes that are RefSeq-representative	16,135	14,730	11,146	7,518	4,909	3,100	1,947	1,276
	100%	91.29%	69.08%	46.59%	30.42%	19.21%	12.07%	7.91%

	Complete dataset	MPS-Sampling runs						
		Δ=1	Δ=0.9	Δ=0.8	Δ=0.7	Δ=0.6	Δ=0.5	Δ=0.4
Genomic size								

Number of genomes linked to a representative with no order affiliation	9,507	9,492	9,455	9,277	8,98	8,659	8,145	7,601
	100%	99.84%	99.45%	97.58%	94.46%	91.08%	85.67%	79.95%
Number of genomes linked to a representative with no family affiliation	13,616	13,590	13,421	13,110	12,617	12,033	11,360	10,406
	100%	99.81%	98.57%	96.28%	92.66%	88.37%	83.43%	76.42%
Number of genomes linked to a representative with no genus affiliation	18,663	18,497	17,866	16,870	15,682	14,307	12,88	11,265
	100%	99.11%	95.73%	90.93%	84.03%	76.66%	69.01%	60.36%
Number of genomes linked to a representative with no species affiliation	41,752	34,284	26,379	21,165	18,063	15,670	13,942	12,027
	100%	82.11%	63.18%	50.69%	43.26%	37.53%	33.39%	28.81%
Number of genomes linked to a representative with at least one Candidatus entry	2,991	2,988	2,995	3,027	3,050	3,006	3,126	3,224
	100%	99.90%	100.13%	101.20%	101.97%	100.50%	104.51%	107.79%
Number of genomes linked to a representative that is RefSeq representative	134,851	99,677	140,556	155,319	159,235	162,622	164,555	166,451
	100%	73.92%	104.23%	115.18%	118.08%	120.59%	122.03%	123.43%
Number of genomes that are singleton MPS cluster	178,203	48,296	19,803	9,882	5,329	2,768	1,415	742
	100%	27.1%	11.11%	5.55%	2.99%	1.55%	0.79%	0.42%
	Complete dataset	**MPS Sampling runs**						
		Δ=1	**Δ=0.9**	**Δ=0.8**	**Δ=0.7**	**Δ=0.6**	**Δ=0.5**	**Δ=0.4**
Taxonomic diversity								
Number of phyla	122	122	122	121	119	119	119	117
	100%	100%	100%	99.18%	97.54%	97.54%	97.54%	95.90%
Number of classes	114	114	112	111	111	110	108	107
	100%	100%	98.25%	97.37%	97.37%	96.49%	94.74%	93.86%
Number of orders	263	263	262	261	259	255	251	242
	100%	100%	99.62%	99.24%	98.48%	96.96%	95.44%	92.02%
Number of families	661	661	654	648	628	607	567	519
	100%	100%	98.94%	98.03%	95.01%	91.83%	85.78%	78.52%
Number of genera	3,896	3,885	3,782	3,491	3,023	2,404	1,770	1,251
	100%	99.72%	97.07%	89.60%	77.59%	61.70%	45.43%	32.11%
Number of species	16,814	15,881	11,858	7,846	5,063	3,173	1,985	1,292
	100%	94.45%	70.52%	46.66%	30.11%	18.87%	11.81%	7.68%
Average number of genomes per phylum	1,459.98	469.26	246.93	157.53	107.00	69.88	44.71	29.55
	100%	32.14%	16.91%	10.79%	7.33%	4.79%	3.06%	2.02%
Average number of genomes per class	1,501.99	452.07	233.31	146.33	96.02	61.95	39.31	24.77
	100%	30.10%	15.53%	9.74%	6.39%	4.12%	2.62%	1.65%
	641.43	184.81	90.28	54.72	35.05	22.09	13.65	8.60

	Complete dataset	Δ=1	Δ=0.9	Δ=0.8	Δ=0.7	Δ=0.6	Δ=0.5	Δ=0.4
Genomic size								
Number of genomes	6,401	1,094	310	147	80	45	24	12
	100%	17.07%	4.84%	2.28%	1.25%	0.70%	0.37%	0.19%
Number of genomes with complete taxonomic affiliation	6,268	1,039	298	142	77	44	24	12
	100%	16.58%	4.75%	2.27%	1.23%	0.70%	0.38%	0.19%
Number of genomes with partial taxonomic affiliation	142	55	12	5	3	1	0	0
	100%	38.73%	8.45%	3.52%	2.11%	0.70%	0%	0%
Number of genomes with no genus affiliation	2	1	1	1	1	0	0	0
	100%	50%	50%	50%	50%	0%	0%	0%
Number of genomes with no species affiliation	142	55	12	5	3	1	0	0
	100%	38.73%	8.45%	3.52%	2.11%	0.70%	0%	0%
Number of genomes with at least one Candidatus entry	1	1	1	1	1	0	0	0
	100%	100%	100%	100%	100%	0%	0%	0%
Number of genomes that are RefSeq representative	366	332	236	132	73	44	24	12
	100%	90.71%	64.48%	36.07%	19.95%	12.02%	6.56%	3.28%

	Complete dataset	MPS Sampling runs						
		Δ=1	Δ=0.9	Δ=0.8	Δ=0.7	Δ=0.6	Δ=0.5	Δ=0.4
Taxonomic diversity								
Number of genera	33	33	33	33	32	27	20	12
	100%	100%	100%	100%	96.97%	81.82%	60.61%	36.36%
	379	362	254	140	77	44	24	12

Phylogenetic diversity								
Length of the ML tree / Number of genomes	0.0026	0.0145	0.0469	0.0811	0.1174	0.1458	0.1896	0.2621

Supplementary Material S37 – The *Bacillaceae* family within the *Bacteria* reduction using MPS-Sampling

	Complete dataset	MPS-Sampling runs						
		Δ=1	Δ=0.9	Δ=0.8	Δ=0.7	Δ=0.6	Δ=0.5	Δ=0.4
Genomic size								
Number of genomes	7,113	1,622	688	417	249	142	78	39
	100%	22.80%	9.67%	5.86%	3.50%	2%	1.10%	0.55%
Number of genomes with complete taxonomic affiliation	6,160	1,258	551	370	236	138	76	39
	100%	20.42%	8.94%	6.01%	3.83%	2.24%	1.23%	0.63%
Number of genomes with partial taxonomic affiliation	953	364	137	47	13	4	2	0
	100%	38.20%	14.38%	4.93%	1.36%	0.42%	0.21%	0%
Number of genomes with no genus affiliation	17	13	8	5	4	1	1	0
	100%	76.47%	47.06%	29.41%	23.53%	5.88%	5.88%	0%
Number of genomes with no species affiliation	953	364	137	47	13	4	2	0
	100%	38.20%	14.38%	4.93%	1.36%	0.42%	0.21%	0%
Number of genomes with at least one Candidatus entry	0	0	0	0	0	0	0	0
	-	-	-	-	-	-	-	-
Number of genomes that are RefSeq-representative	625	585	494	362	234	136	74	37
	100%	93.60%	79.04%	57.92%	37.44%	21.76%	11.84%	5.92%

Number of species	644	619	509	366	236	138	76	39
	100%	96.12%	79.04%	56.83%	36.65%	21.43%	11.80%	6.06%
Average number of genomes per genus	65.70	14.80	6.36	4.00	2.63	1.86	1.00	1.22
	100%	22.67%	9.67%	6.09%	4.01%	2.82%	2.13%	1.85%
Average number of genomes per species	9.57	2.03	1.08	1.01	1.00	1.00	1.00	1.00
	100%	21.25%	11.32%	10.57%	10.45%	10.45%	10.45%	10.45%

Phylogenetic diversity								
Length of the ML tree / Number of genomes	0.0046	0.0199	0.0447	0.0834	0.0840	0.1083	0.1341	0.1643

Supplementary Material S38 – The *Enterobacteriaceae* family within the *Bacteria* reduction using MPS-Sampling

	Complete dataset	**MPS-Sampling runs**						
		Δ=1	**Δ=0.9**	**Δ=0.8**	**Δ=0.7**	**Δ=0.6**	**Δ=0.5**	**Δ=0.4**
Genomic size								
Number of genomes	17,096	682	113	40	37	28	21	18
	100%	3.99%	0.66%	0.23%	0.22%	0.16%	0.12%	0.11%
Number of genomes with complete taxonomic affiliation	15,692	550	82	36	26	22	16	13
	100%	3.50%	0.52%	0.23%	0.17%	0.14%	0.10%	0.08%
Number of genomes with partial taxonomic affiliation	1,404	132	31	13	11	6	5	5

	Complete dataset	MPS-Sampling runs						
		Δ=1	Δ=0.9	Δ=0.8	Δ=0.7	Δ=0.6	Δ=0.5	Δ=0.4
Taxonomic diversity								
Number of genera	58	52	38	27	22	20	16	13
	100%	89.66%	65.52%	46.55%	37.93%	34.48%	27.59%	22.41%
Number of species	178	139	67	33	24	22	16	13
	100%	78.09%	37.64%	18.54%	13.48%	12.36%	8.99%	7.30%
Average number of genomes per genus	294.21	12.56	2.58	1.56	1.41	1.25	1.19	1.23
	100%	4.27%	0.88%	0.53%	0.48%	0.42%	0.40%	0.42%
Average number of genomes per species	88.16	3.96	1.22	1.09	1.08	1.00	1.00	1.00
	100%	4.49%	1.39%	1.24%	1.23%	1.13%	1.13%	1.13%

Bacterial backbone								
Number of leaves	35,103	31,631	24,248	17,547	12,775	8,352	5,347	3,474
Length of the ML tree / Number of leaves	0.0716	0.0793	0.1009	0.1290	0.1590	0.2003	0.2473	0.2983
Lactobacillaceae								
Number of leaves	6,410	1,094	310	147	80	45	24	12
Length of the ML tree / Number of leaves	0.0026	0.0145	0.0469	0.0811	0.1174	0.1458	0.1896	0.2621
Bacillaceae								
Number of leaves	7,113	1,622	688	417	240	142	78	30
Length of the ML tree / Number of leaves	0.0046	0.0199	0.0447	0.0634	0.0840	0.1083	0.1341	0.1643
Enterobacteriaceae								
Number of leaves	17,096	682	113	40	37	28	21	18
Length of the ML tree / Number of leaves	0.0008	0.0186	0.1002	0.2016	0.2525	0.2912	0.3311	0.3580

Supplementary Material S3340 – Taxonomic statistics about each investigated subset

MPS-–Sampling was launched on the bacterial dataset, encompassing 178,203 bacterial genomes. In addition to the complete dataset, some subsets were investigated. The constitution of the bacterial backbone, a subset encompassing 35,103 genomes, is described in the Supplementary Material S35~~Supplementary Material S42~~. Three taxonomic families have also been investigated, respectively *Lactobacillaceae* *Bacillaceae* and *Enterobacteriaceae*.

For example, the family *Lactobacillaceae* encompassed 6,410 genomes, including 4,992 (98%) with a complete nomenclature. It involves 361 species and 32 genera, with 11.28 species per genus in average. Concerning the taxonomic density, there were 13.83 and 159.72 genomes per species and genus in average, respectively.

Subset		Genome Number			Taxonomic Diversity		Taxonomic Density		
Level	**Name**	**Total**	**Having a complete nomenclature**		**Number of species** *(without sp. and subsp.)*	**Number of genera**	**Species per genus**	**Genomes per species** *(from complete nomenclature)*	**Genomes per genus** *(from complete nomenclature)*
domain	bacterial dataset	178,203	135,315	76%	16,814	3,896	4.32	8.12	40.95
domain	bacterial backbone	35,103	16,836	48%	16,228	3,787	4.29	1.07	5.13
family	*Lactobacillaceae*	6,410	6,268	98%	379	33	11.48	16.56	194.18
family	*Bacillaceae*	7,113	6,160	87%	644	108	5.96	9.57	65.70
family	*Enterobacteriaceae*	17,096	15,692	92%	178	58	3.07	88.16	294.21

Supplementary Material S3441 – Phylogenic statistics about each phylogenetic inference

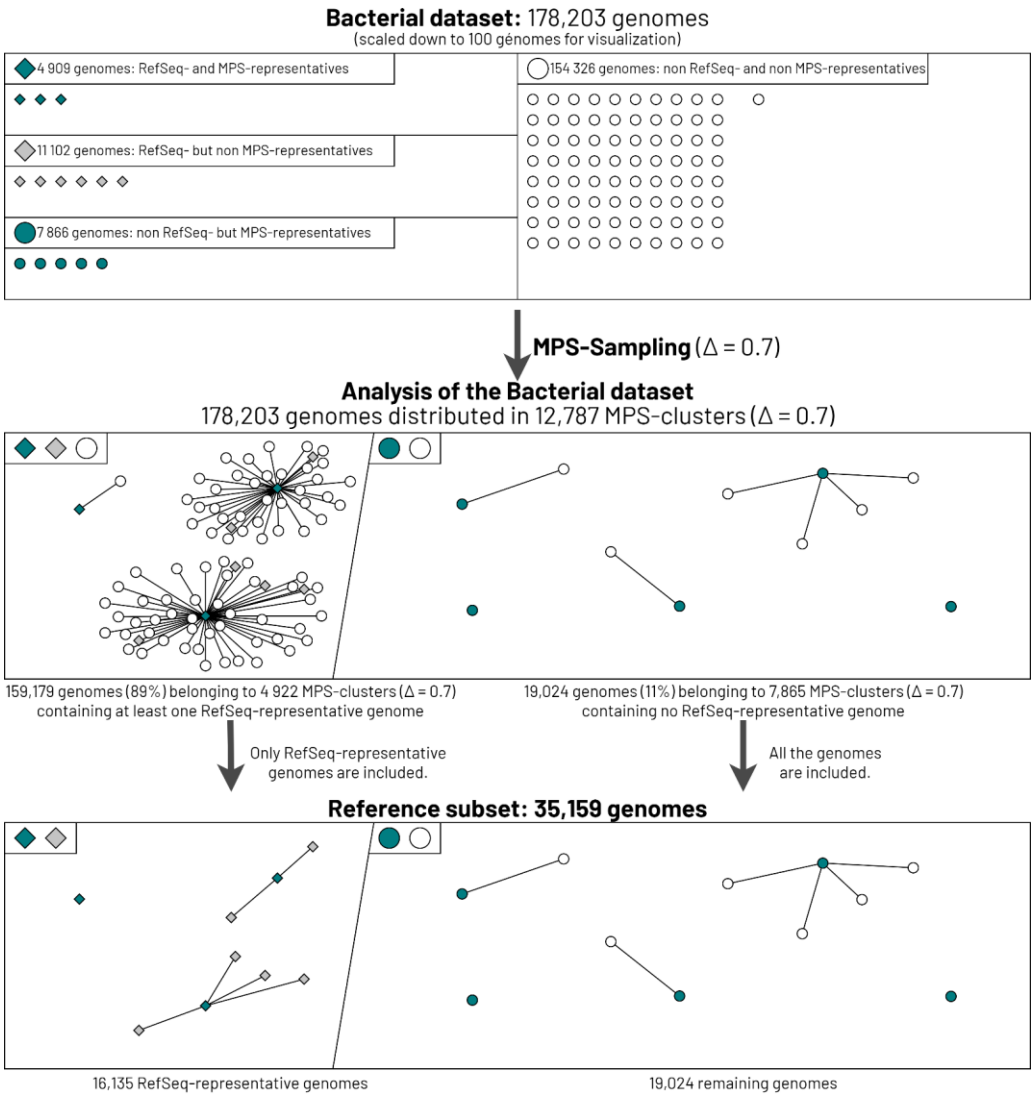
For each investigated group (level + group name), a set of genomes was used for phylogenetic inference (genomes number). After recruitment, alignment and trimming, a supermatrix was built with a given number of rows (sequences number) and columns (positions number); moreover, it had a given number of missing values (gaps number) and a given proportion of missing values among all cells (gaps ratio). From the computed phylogenetic tree, the sum of all branch lengths was divided by the number of tips; this quotient gave an indication about the phylogenetic diversity of the genomic set (total length / number of tips).

For example, the family *Lactobacillaceae* encompassed 6,410 genomes, i.e. 6,410 sequences (one sequence per genome) and 6,105 positions. So, the supermatrix had 6,410 rows and 6,105 columns. There were 1,020,435 gaps in the supermatrix, i.e. 2.61% of it. After the phylogenetic inference, the sum of all branch lengths divided by the number of tips was equal to 0.0026. This provides an indicator about the phylogenetic inference.

Subset		Supermatrix					Tree
Level	**Group Name**	**Genomes Number**	**Sequences Number**	**Positions Number**	**Gaps Number**	**Gaps Ratio**	**Total length / Number of tips**
domain	bacterial dataset	178,203	178,203	5,874	30,880,709	2.95%	0.0159
domain	bacterial backbone	35,103	35,103	5,819	10,386,208	5.08%	0.0716
family	*Lactobacillaceae*	6,410	6,410	6,105	1,020,435	2.61%	0.0026
family	*Bacillaceae*	7,113	7,113	6,138	434,529	1.00%	0.0046
family	*Enterobacteriaceae*	17,096	17,096	6,188	514,233	0.49%	0.0008

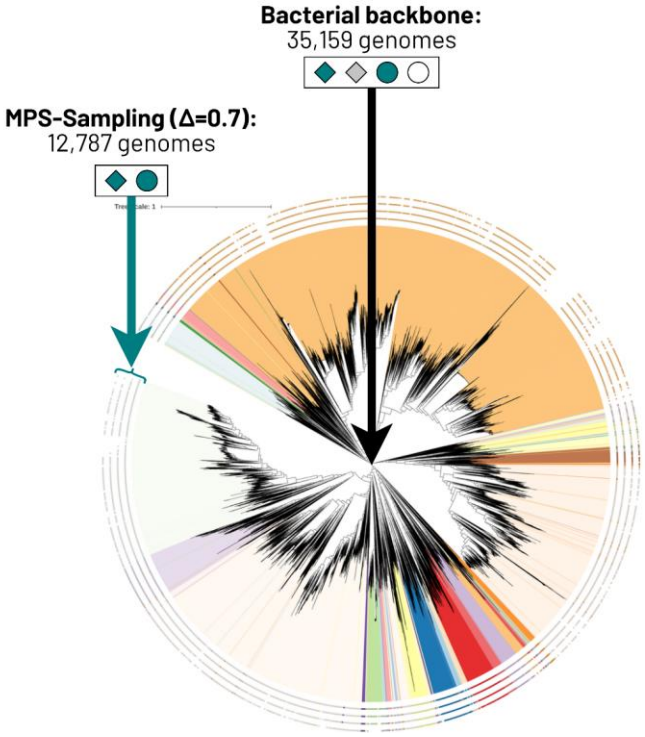
Supplementary Material S3542 – Construction of the bacterial backbone

For visualization, the bacterial dataset of 178,203 genomes was reduced to a bacterial backbone of 35,103 genomes. More precisely, all the 16,135 RefSeq–representative genomes were kept, as they were considered as a standard against which other data should be compared. When $\Delta = 0.7$, these genomes were MPS–representatives and/or belonged to 4,909 MPS clusters, that together represent 159,235 genomes (90%). It was assumed that these 16,135 RefSeq–representative genomes were a reliable reference to represent these 159,235 genomes. The remaining 18,968 genomes (10%) were distributed across the 7,866 MPS clusters that did not contain any RefSeq representative genomes. Because these genomes could not be linked to any reference external to our analysis, all of them were kept for the phylogenetic analysis. Thus, in total, 16,135 + 18,968 = 35,103 genomes were used to infer a reference phylogeny from the bacterial dataset.



Supplementary Material S35 – Construction of the bacterial backbone~~Supplementary Material S42 – Construction of the bacterial backbone~~ (suite)

For visualization, the bacterial dataset of 178,203 genomes was reduced to a bacterial backbone of 35,103 genomes. More precisely, all the 16,135 RefSeq–representative genomes were kept, as they were considered as a standard against which other data should be compared. When $\Delta = 0.7$, these genomes were MPS–representatives and/or belonged to 4,909 MPS clusters, that together represent 159,235 genomes (90%). It was assumed that these 16,135 RefSeq–representative genomes were a reliable reference to represent these 159,235 genomes. The remaining 18,968 genomes (10%) were distributed across the 7,866 MPS clusters that did not contain any RefSeq representative genomes. Because these genomes could not be linked to any reference external to our analysis, all of them were kept for the phylogenetic analysis. Thus, in total, 16,135 + 18,968 = 35,103 genomes were used to infer a reference phylogeny from the bacterial dataset.

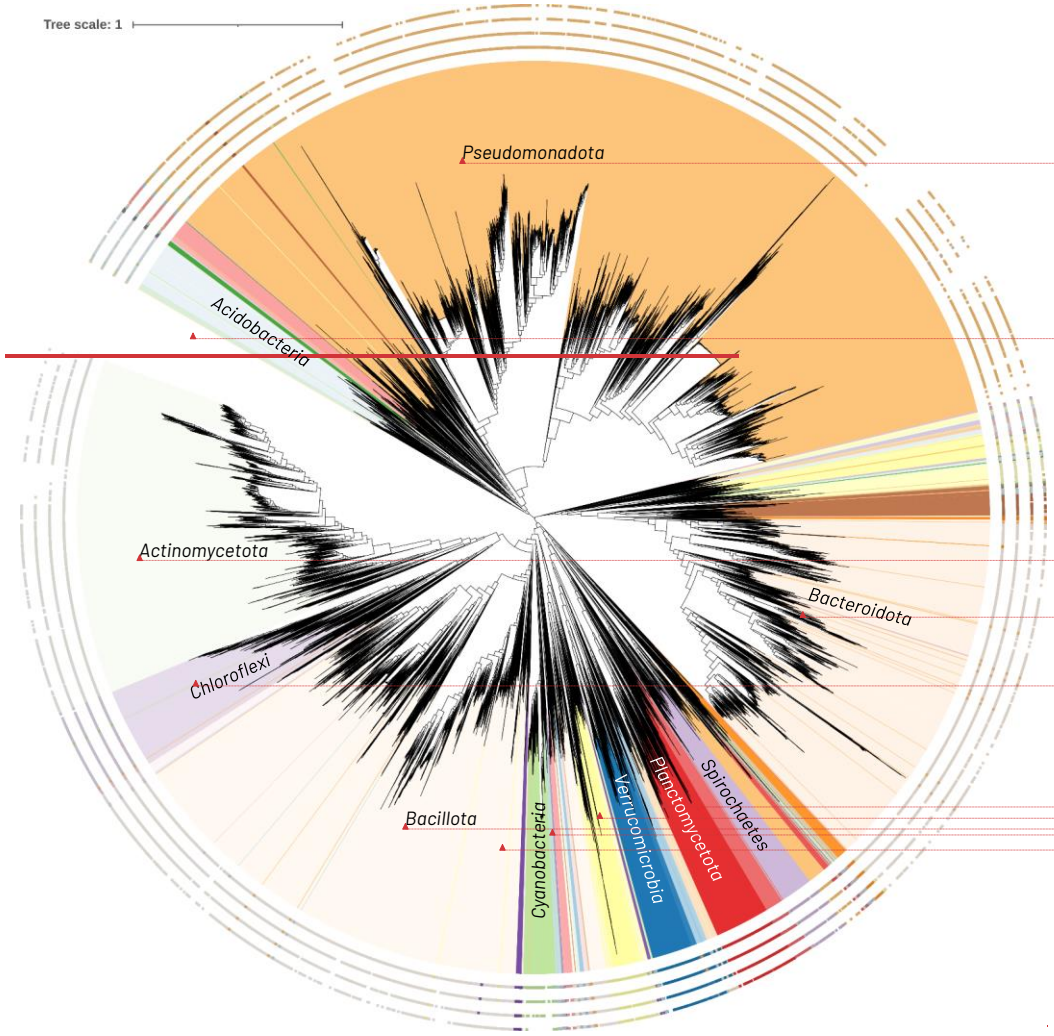


Supplementary Material S36 – Phylogenetic distribution of the MPS-representatives

The MPS-representative genomes from the four samples corresponding to $\Delta \in \{0.7; 0.6; 0.5; 0.4\}$ were tagged around the reference phylogeny of the bacterial backbone of the 35,159 genomes (Supplementary Material S35). From the innermost to the outermost crown correspond four values: $\Delta \in \{0.7; 0.6; 0.5; 0.4; 0.3; 0.2; 0.1; 0.05\}$, indicating 12,787, 8,343, 5,332, 3,442, 2,252, 1,368, 794 and 527 MPS- representative genomes, respectively. The scale represents the average number of amino acid substitutions per position in the protein sequences used to infer the tree.

The 10 most represented phyla are shown:

- *Pseudomonadota* (formerly *Proteobacteria*, 12,816 leaves in orange at the top);
- *Bacillota* (formerly *Firmicutes*, 5,767 leaves in light beige at the bottom);
- *Bacteroidota* (formerly *Bacteroidetes*, 4,705 leaves in light orange on the right);
- *Actinomycetota* (formerly *Actinobacteria*, 4,259 leaves in light green on the left);
- *Chloroflexi* (1,001 leaves in light purple on the left);
- *Planctomycetota* (formerly *Planctomycetes*, 691 leaves in red below);
- *Acidobacteria* (615 leaves in light blue top left);
- *Verrucomicrobia* (560 leaves in dark blue below);
- *Spirochaetes* (415 purple leaves bottom right);
- *Cyanobacteria* (413 leaves in green at bottom).



a mis en forme : Police :10 pt

a mis en forme : Police :10 pt

a mis en forme : Police :10 pt

a mis en forme : Police :10 pt

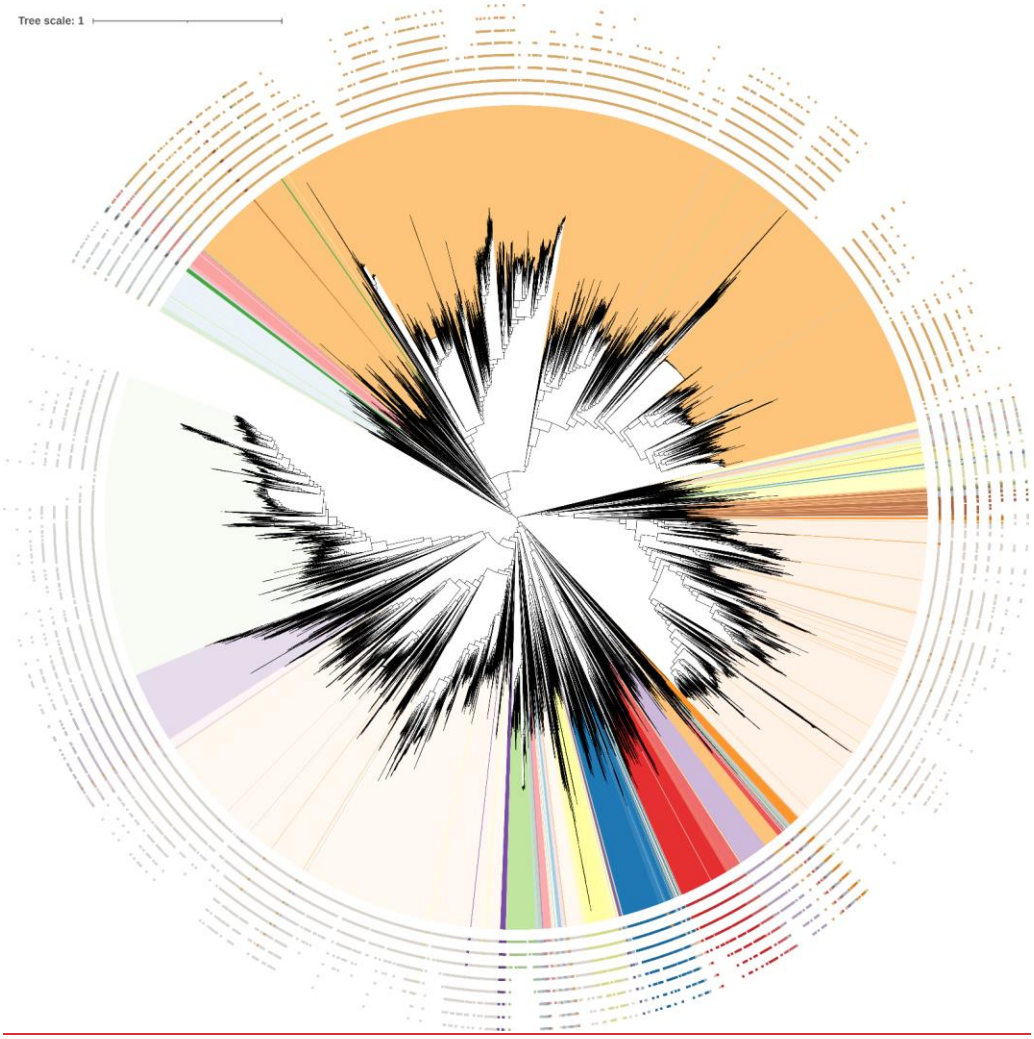
a mis en forme : Police :10 pt

a mis en forme : Police :10 pt

a mis en forme : Police :10 pt

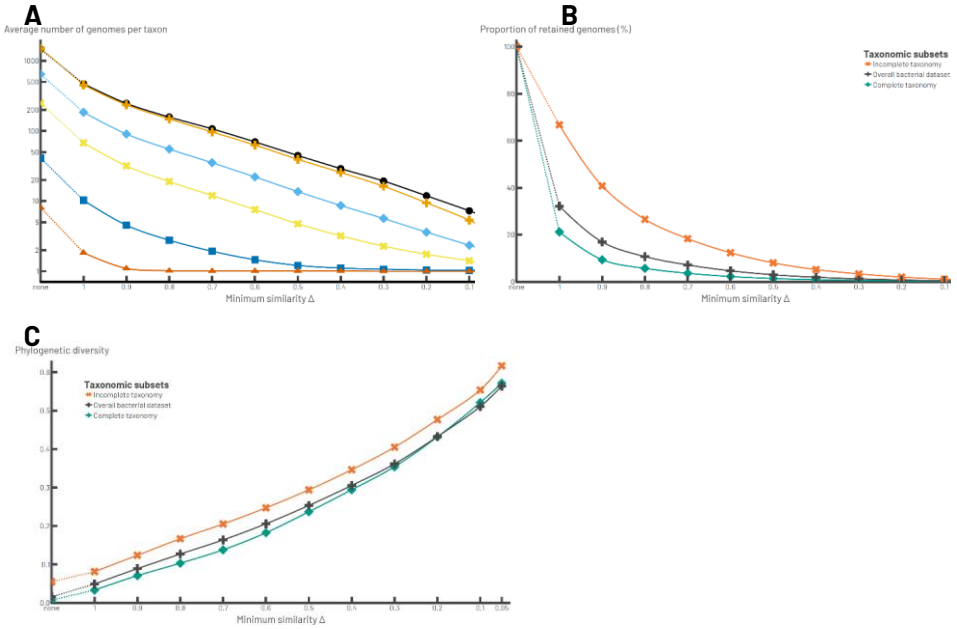
a mis en forme : Police :10 pt

a mis en forme : Police :10 pt


Tree scale: 1

Supplementary Material S3744 – Reduction and taxonomic affiliation

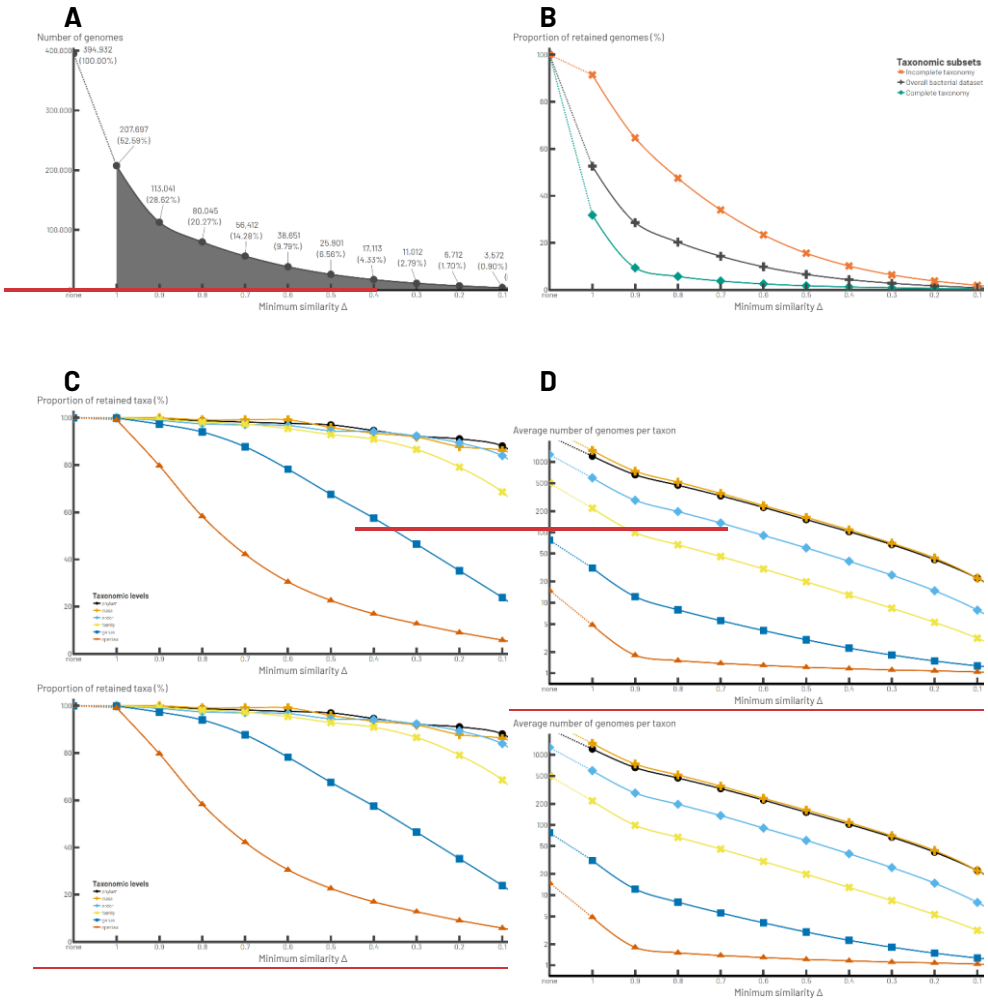
A: Taxonomic redundancy of the samples. Average number of genomes representing each taxonomic level in the samples.
B: Genomic reduction of the 135,315 genomes and the 42,888 genomes with a complete and incomplete taxonomic affiliation, respectively, with a normalized scale.
C: Phylogenetic diversity of three subsets: the overall bacterial dataset, the subset of genomes with complete taxonomic affiliation and the subset of genomes with incomplete taxonomic affiliation. The phylogenetic diversity was computed by the length of all branches divided by the number of leaves.



Supplementary Material S3845 – MPS-Sampling on the GTDB data

MPS-Sampling was also tested on the data of the GTDB, encompassing 120 core protein families present in 394,932 bacterial genomes (Parks *et al.*, 2022). Here are some results.

A: Size of the samples built by MPS-Sampling.
B: Genomic reduction of the 135,315 genomes and the 42,888 genomes with a complete and incomplete taxonomic affiliation, respectively, with a normalized scale.
C: Taxonomic diversity of the samples. The proportion of phyla, classes, orders, families, genera, and species represented in each sample is indicated.
D: Taxonomic redundancy of the samples. Average number of genomes representing each taxonomic level in the samples.



Supplementary Material S46 – Inspection of *Bacteria* samplings at a local scale

Sampling of *Lactobacillaceae* family (*Firmicutes* phylum)

The *Lactobacillaceae* family included 6,401 genomes from 33 genera and 379 species (Supplementary Material S36). The taxonomic distribution was balanced, with on average 17 genomes per species. Because the *Lactobacillaceae* was well studied and characterized, a high level of redundancy was observed at both taxonomic and genetic levels: the average number of genomes per genus and species was more than four times higher in this taxonomic family than in the complete bacterial dataset (194 and 41 genomes per genus, and 17 and 8 genomes per species, respectively). This redundancy was also obvious at the phylogenetic level, as the diversity of *Lactobacillaceae* was 3 times lower than for the bacterial dataset (0.0047 and 0.0159, respectively) (Supplementary Material S47B).

Consistently, the dereplication of *Lactobacillaceae* genomes was more intense than for *Bacteria* (Supplementary Material S47A). For instance, when $\Delta = 1$, even if 100% of the genera and 96% of the species were conserved, only 17% of the genomes were kept. As Δ decreased, the number of genomes per genus and per species gradually decreased to 1 (for $\Delta \leq 0.9$ and $\Delta \leq 0.5$ respectively), meaning that all the intra-species and intra-genus redundancy was eliminated.

Compared to *Bacteria*, the reduction of *Lactobacillaceae* was 2 to 8 times higher (Supplementary Material S47A), while the genetic diversity of the *Lactobacillaceae* gradually increased to reach a comparable level when $\Delta = 0.4$ (Supplementary Material S47B). The phylogenetic mapping of MPS-representatives was also reliable, with a denser sampling in regions of the tree with greater phylogenetic diversity (Supplementary Material S48A). This illustrated the ability of MPS-Sampling to adjust sampling intensity according to taxonomic and genetic redundancy in the data, but also to homogenize the taxonomic and genetic diversity of a data set at different evolutionary scales.

Sampling of *Bacillaceae* family (*Firmicutes* phylum)

The case of the *Bacillaceae* was more complex, because this family may not be monophyletic (Maayer *et al.*, 2019). In this context, it was quite possible that part of the reconstructed MPS-clusters mixed genomes from *Bacillaceae* and other families, leading to sampling biases. In the bacterial dataset, the *Bacillaceae* family included 7,113 genomes from 108 genera and 644 species (Supplementary Material S37). However, their taxonomic distribution was unbalanced (Supplementary Material S48B), with the genus *Bacillus* alone comprising three-quarters of the *Bacillaceae* genomes (5,260 genomes out of 7,113). Despite this, the samplings were reliable regarding the phylogenetic distribution of the MPS-representatives (Supplementary Material S48B). In particular, the overrepresentation of *Bacillus* was successfully reduced as they account for 18% to 10% of the samples, 44 out of 249 when $\Delta = 0.7$ and 4 out of 39 when $\Delta = 0.4$, respectively. In fact, most MPS-representatives belonged to genera other than *Bacillus*, demonstrating the ability of MPS-Sampling to capture the genetic diversity of *Bacillaceae*. It also balanced the representativeness of *Bacillaceae*, whose taxonomic and genetic diversity in the samples became comparable to that of *Lactobacillaceae* and *Bacteria* (Supplementary Material S47B).

Through the example of *Bacillaceae*, MPS-Sampling showed its ability to produce relevant samples even in cases where the initial level of redundancy was very high and unbalanced, but also when taxonomy and phylogeny disagreed.

Sampling of the *Enterobacteriaceae* family (*Proteobacteria* phylum)

The *Enterobacteriaceae* represented another interesting case because their taxonomy was more unbalanced than that of the *Bacillaceae*. The *Enterobacteriaceae* family included 17,096 genomes from 178 species and 58 genera, which theoretically corresponded to an average of 88 and 294 genomes per species and per genus, respectively (Supplementary Material S38). However, this was far from the case as six genera (*Klebsiella*, *Enterobacter*, *Salmonella*, *Shigella*, *Escherichia*, and *Citrobacter*) accounted for 92% of the genomes (15,728 out of 17,096). Consistently, even with the densest sampling ($\Delta = 1$), a few

genomes (3.99%) were kept, compared to 32.17%, 17.07%, and 22.80% for *Bacteria*, *Lactobacillaceae*, and *Bacillaceae*, respectively (Supplementary Material S47A). The situation was even more extreme when $\Delta = 0.4$: as 0.11% of the genomes were retained, compared to 1.95%, 0.19%, and 0.55% for *Bacteria*, *Lactobacillaceae*, and *Bacillaceae*, respectively (Supplementary Material S47A). From $\Delta \leq 0.7$, the six most represented genera were reduced to only 1 MPS-representative. Regarding the taxonomic density, one genome per species and genus was kept for *Enterobacteriaceae*, from $\Delta \leq 0.9$ and $\Delta \leq 0.7$ respectively, indicating that as for *Bacteria*, *Lactobacillaceae*, and *Bacillaceae*, most of the redundancy within genera and species was eliminated. However, dereplication was much higher, as 37.93% of the genera conserved when $\Delta = 0.7$, compared to 96.97%, and 86.11% for *Lactobacillaceae*, and *Bacillaceae*, respectively. This reflected a much lower inter-genera and inter-species diversity in *Enterobacteriaceae* than in the two others families. This might be due to biases in the delineation of taxa, which might reflect historical legacy and/or practical convenience, as in the case of the genera *Shigella* and *Escherichia* (Lan and Reeves, 2002). Regardless of the origin of these biases, taxonomy-based sampling would have led to an over-representation of *Enterobacteriaceae*, but also of higher taxa (i.e. *Enterobacterales* and *Gammaproteobacteria*) in the samples.

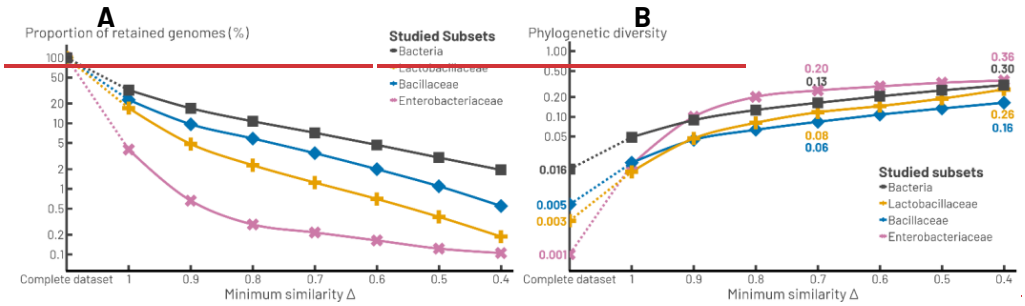
Considering phylogenetic diversity, *Enterobacteriaceae* were initially less diversified than *Bacteria*, *Lactobacillaceae*, and *Bacillaceae* (0.0008, compared to 0.0159, 0.0026, and 0.0046, respectively (Supplementary Material S48C). As with other taxonomic groups, the phylogenetic diversity of *Enterobacteriaceae* increased as sampling density decreased. More precisely, although *Enterobacteriaceae* has the lowest initial phylogenetic diversity, the diversity of its MPS-samples is higher than for the other groups from $\Delta = 0.8$ to $\Delta = 0.4$ (Supplementary Material S48C). This indicated that the true phylogenetic diversity of *Enterobacteriaceae* was hidden by its extreme genetic redundancy. And indeed, the *Enterobacteriaceae* included highly divergent strains and species, with very diverse lifestyles and habitats (e.g., insect endosymbionts, plant and animal pathogens, extremophiles...), some of which might have high rates of evolution. While being phylogenetically agnostic, but thanks to its ability to adapt the sampling density, MPS-Sampling succeeded in revealing the true phylogenetic diversity of *Enterobacteriaceae*, despite a very high initial redundancy.

Supplementary Material S47 – Reduction of three taxonomic families

A: Genomic reduction of the three taxonomic families *Lactobacillaceae*, *Bacillaceae* and *Enterobacteriaceae*, in comparison with *Bacteria*.

B: Phylogenetic diversity of the three taxonomic families *Lactobacillaceae*, *Bacillaceae* and *Enterobacteriaceae*, in comparison with *Bacteria*, computed by the length of all branches divided by the number of leaves.

All precise statistics are available in Supplementary Material S35–S25.

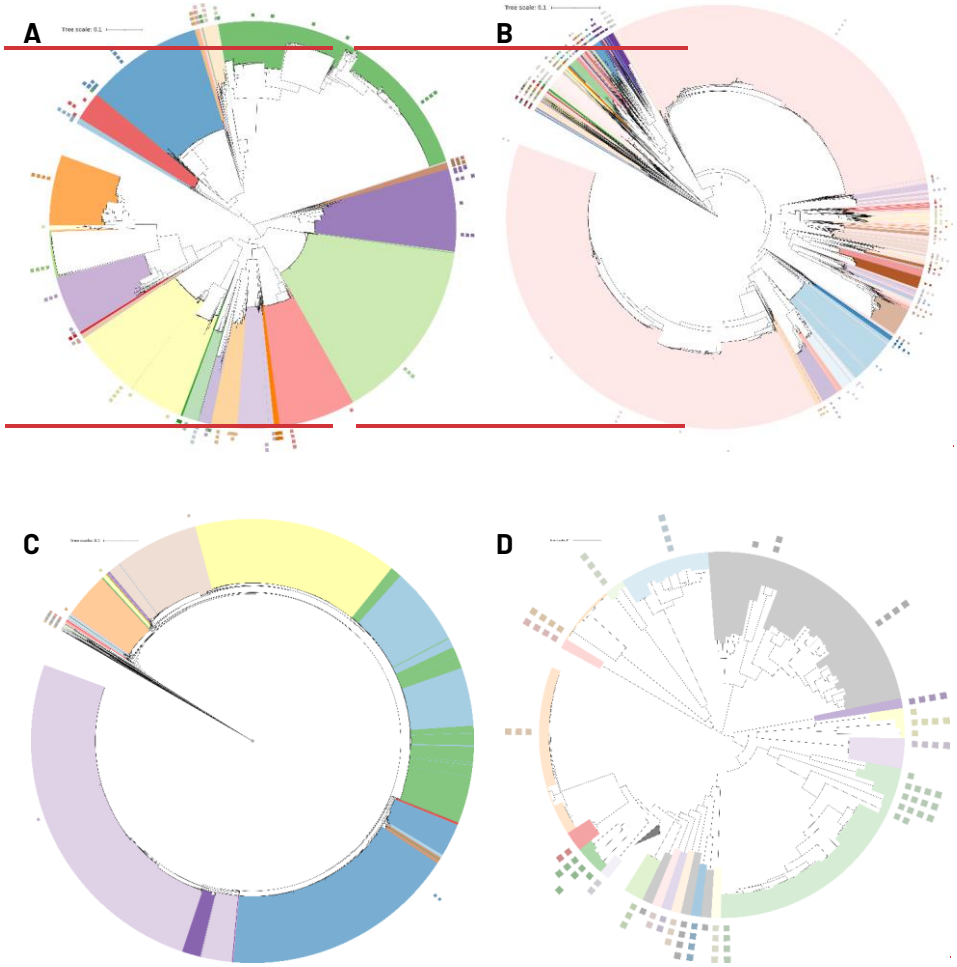


A: ML tree of the 6,410 *Lactobacillaceae* genomes present in the bacterial dataset. The tree has been inferred with r_prot sequences. From the innermost to the outermost circle: MPS-representatives corresponding to Δ=0.7 (80 MPS-representatives), Δ=0.6 (45 MPS-representatives), Δ=0.5 (24 MPS-representatives), and Δ=0.4 (12 MPS-representatives). Colors correspond to the 33 genera of *Lactobacillaceae*.

B: ML tree of the 7,113 *Bacillaceae* genomes present in the bacterial dataset. The tree has been inferred with r-prot sequences. From the innermost to the outermost circle: MPS-representatives corresponding to Δ=0.7 (249 MPS-representatives), Δ=0.6 (142 MPS-representatives), Δ=0.5 (78 MPS-representatives), and Δ=0.4 (39 MPS-representatives). Colors correspond to the 108 genera of *Bacillaceae*.

C: ML tree of the 17,096 *Enterobacteriaceae* genomes present in the bacterial dataset. The tree has been inferred with r-prot sequences. From the innermost to the outermost circle: MPS-representatives corresponding to Δ=0.7 (37 MPS-representatives), Δ=0.6 (28 MPS-representatives), Δ=0.5 (21 MPS-representatives), and Δ=0.4 (18 MPS-representatives). Colors correspond to the 58 genera of *Enterobacteriaceae*.

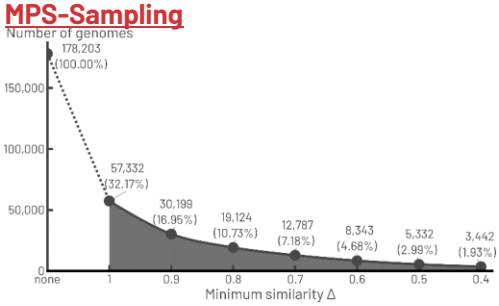
D: ML tree of the 17,096 *Enterobacteriaceae* genomes with 16,987 leaves collapsed. It highlights the 18 Candidatus genera contained within the *Enterobacteriaceae* genomes, representing the large majority of the MPS-samples: 27 out of 37 and 14 out of 18 MPS-representatives are Candidatus genomes, for Δ=0.7 and Δ=0.4 respectively.



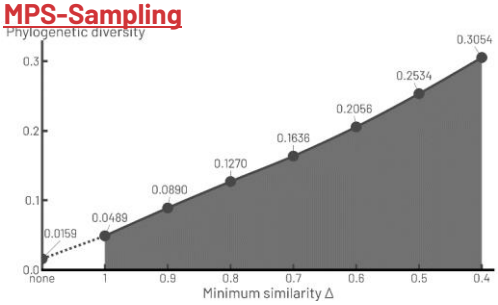
Supplementary Material S39 – Comparison of sampling rate and phylogenetic diversity
Supplementary Material S3949 – Comparison of sampling rate and phylogenetic diversity
Samplings of the bacterial dataset with MPS-Sampling, Treemmer and TAX-Sampling are compared according to :

- the sampling rate (left column);
- the phylogenetic diversity (right column).

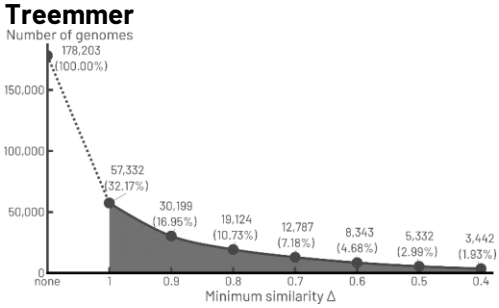
MPS-Sampling



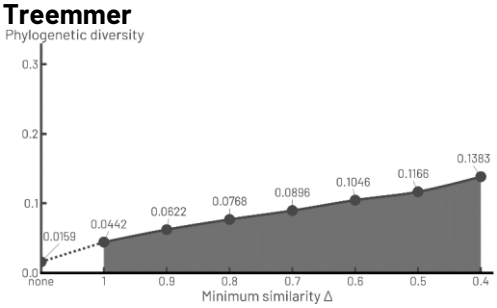
MPS-Sampling



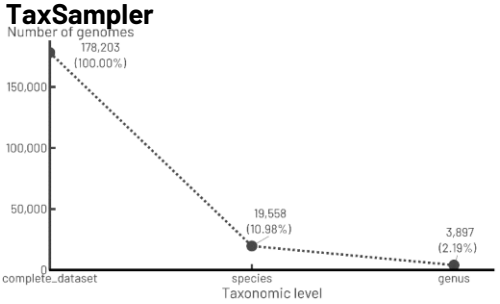
Treemmer



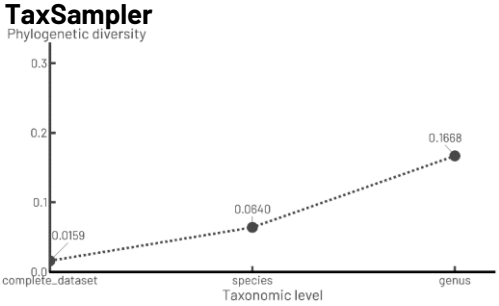
Treemmer



TaxSampler



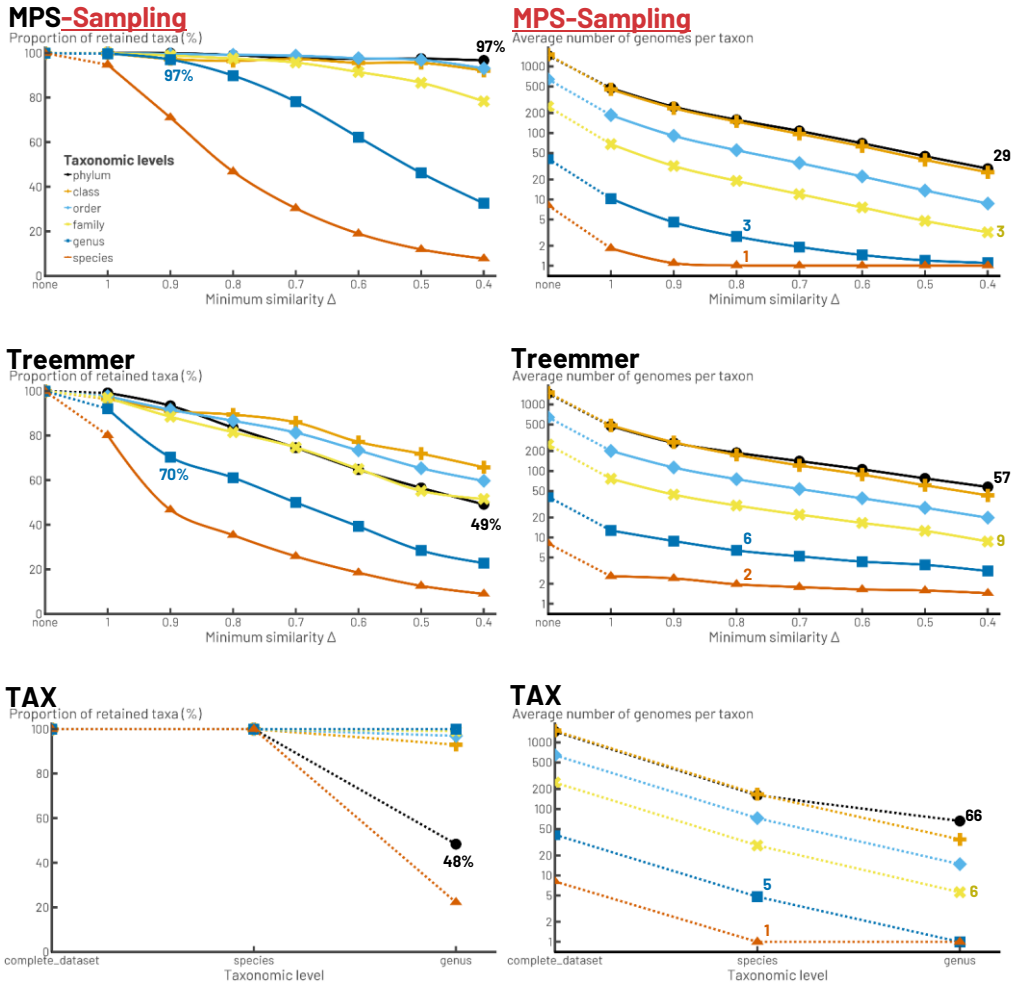
TaxSampler



Supplementary Material S40 – Comparison of taxonomic diversity and taxonomic redundancy

Samplings of the bacterial dataset with MPS-Sampling, Treemmer and TAX-Sampling are compared according to:

- the taxonomic diversity, with the proportion of retained taxa per level (left column);
- the taxonomic redundancy, with the average number of genomes per taxon for each level (right column).



Supplementary Material S41 – Inspection of *Bacteria* samplings at a local scale

Sampling of *Lactobacillaceae* family (*Firmicutes* phylum)

The *Lactobacillaceae* family included 6,401 genomes from 33 genera and 379 species (). The taxonomic distribution was balanced, with on average 17 genomes per species. Because the *Lactobacillaceae* was well studied and characterized, a high level of redundancy was observed at both taxonomic and genetic levels: the average number of genomes per genus and species was more than four times higher in this taxonomic family than in the complete bacterial dataset (194 and 41 genomes per genus, and 17 and 8 genomes per species, respectively). This redundancy was also obvious at the phylogenetic level, as the diversity of *Lactobacillaceae* was 3 times lower than for the bacterial dataset (0.0047 and 0.0159, respectively)(B).

Consistently, the dereplication of *Lactobacillaceae* genomes was more intense than for *Bacteria* (A). For instance, when $\Delta = 1$, even if 100% of the genera and 96% of the species were conserved, only 17% of the genomes were kept. As Δ decreased, the number of genomes per genus and per species gradually decreased to 1 (for $\Delta \leq 0.9$ and $\Delta \leq 0.5$ respectively), meaning that all the intra-species and intra-genus redundancy was eliminated.

Compared to *Bacteria*, the reduction of *Lactobacillaceae* was 2 to 8 times higher (A), while the genetic diversity of the *Lactobacillaceae* gradually increased to reach a comparable level when $\Delta = 0.4$ (B). The phylogenetic mapping of MPS-representatives was also reliable, with a denser sampling in regions of the tree with greater phylogenetic diversity (A).This illustrated the ability of MPS-Sampling to adjust sampling intensity according to taxonomic and genetic redundancy in the data, but also to homogenize the taxonomic and genetic diversity of a data set at different evolutionary scales.

Sampling of *Bacillaceae* family (*Firmicutes* phylum)

The case of the *Bacillaceae* was more complex, because this family may not be monophyletic (Maayer *et al.*, 2019). In this context, it was quite possible that part of the reconstructed MPS-clusters mixed genomes from *Bacillaceae* and other families, leading to sampling biases. In the bacterial dataset, the *Bacillaceae* family included 7,113 genomes from 108 genera and 644 species (). However, their taxonomic distribution was unbalanced (B), with the genus *Bacillus* alone comprising three-quarters of the *Bacillaceae* genomes (5,260 genomes out of 7,113). Despite this, the samplings were reliable regarding the phylogenetic distribution of the MPS-representatives (B). In particular, the overrepresentation of *Bacillus* was successfully reduced as they account for 18% to 10% of the samples, 44 out of 249 when $\Delta = 0.7$ and 4 out of 39 when $\Delta = 0.4$, respectively. In fact, most MPS-representatives belonged to genera other than *Bacillus*, demonstrating the ability of MPS-Sampling to capture the genetic diversity of *Bacillaceae*. It also balanced the representativeness of *Bacillaceae*, whose taxonomic and genetic diversity in the samples became comparable to that of *Lactobacillaceae* and *Bacteria* (B).

Through the example of *Bacillaceae*, MPS-Sampling showed its ability to produce relevant samples even in cases where the initial level of redundancy was very high and unbalanced, but also when taxonomy and phylogeny disagreed.

Sampling of the *Enterobacteriaceae* family (*Proteobacteria* phylum)

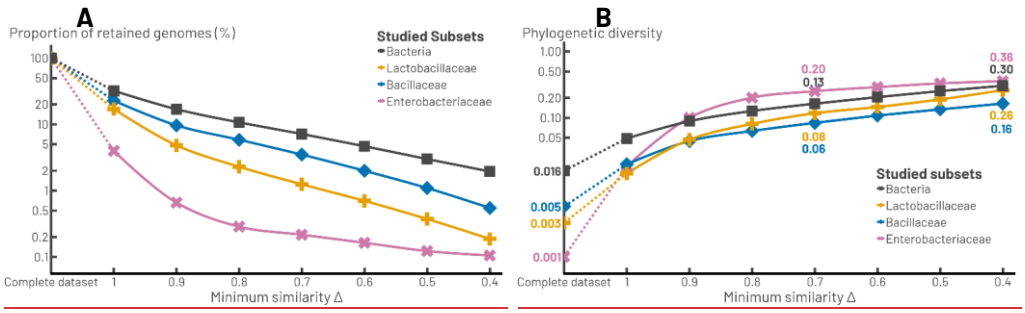
The *Enterobacteriaceae* represented another interesting case because their taxonomy was more unbalanced than that of the *Bacillaceae*. The *Enterobacteriaceae* family included 17,096 genomes from 178 species and 58 genera, which theoretically corresponded to an average of 88 and 294 genomes per species and per genus, respectively (). However, this was far from the case as six genera (*Klebsiella*, *Enterobacter*, *Salmonella*, *Shigella*, *Escherichia*, and *Citrobacter*) accounted for 92% of the genomes (15,728 out of 17,096). Consistently, even with the densest sampling ($\Delta = 1$), a few genomes (3.99%) were

kept, compared to 32.17%, 17.07%, and 22.80% for *Bacteria*, *Lactobacillaceae*, and *Bacillaceae*, respectively (A). The situation was even more extreme when $\Delta = 0.4$: as 0.11% of the genomes were retained, compared to 1.95%, 0.19%, and 0.55% for *Bacteria*, *Lactobacillaceae*, and *Bacillaceae*, respectively (A). From $\Delta \leq 0.7$, the six most represented genera were reduced to only 1 MPS-representative. Regarding the taxonomic density, one genome per species and genus was kept for *Enterobacteriaceae*, from $\Delta \leq 0.9$ and $\Delta \leq 0.7$ respectively, indicating that as for *Bacteria*, *Lactobacillaceae*, and *Bacillaceae*, most of the redundancy within genera and species was eliminated. However, dereplication was much higher, as 37.93% of the genera conserved when $\Delta = 0.7$, compared to 96.97%, and 86.11% for *Lactobacillaceae*, and *Bacillaceae*, respectively. This reflected a much lower inter-genera and inter-species diversity in *Enterobacteriaceae* than in the two others families. This might be due to biases in the delineation of taxa, which might reflect historical legacy and/or practical convenience, as in the case of the genera *Shigella* and *Escherichia* (Lan and Reeves, 2002). Regardless of the origin of these biases, taxonomy-based sampling would have led to an over-representation of *Enterobacteriaceae*, but also of higher taxa (i.e. *Enterobacterales* and *Gammaproteobacteria*) in the samples.

Considering phylogenetic diversity, *Enterobacteriaceae* were initially less diversified than *Bacteria*, *Lactobacillaceae*, and *Bacillaceae* (0.0008, compared to 0.0159, 0.0026, and 0.0046, respectively (C). As with other taxonomic groups, the phylogenetic diversity of *Enterobacteriaceae* increased as sampling density decreased. More precisely, although *Enterobacteriaceae* has the lowest initial phylogenetic diversity, the diversity of its MPS-samples is higher than for the other groups from $\Delta = 0.8$ to $\Delta = 0.4$ (C). This indicated that the true phylogenetic diversity of *Enterobacteriaceae* was hidden by its extreme genetic redundancy. And indeed, the *Enterobacteriaceae* included highly divergent strains and species, with very diverse lifestyles and habitats (e.g., insect endosymbionts, plant and animal pathogens, extremophiles…), some of which might have high rates of evolution. While being phylogenetically agnostic, but thanks to its ability to adapt the sampling density, MPS-Sampling succeeded in revealing the true phylogenetic diversity of *Enterobacteriaceae*, despite a very high initial redundancy.

Supplementary Material S42 – Reduction of three taxonomic families

A: Genomic reduction of the three taxonomic families *Lactobacillaceae*, *Bacillaceae* and *Enterobacteriaceae*, in comparison with *Bacteria*.
B: Phylogenetic diversity of the three taxonomic families *Lactobacillaceae*, *Bacillaceae* and *Enterobacteriaceae*, in comparison with *Bacteria*, computed by the length of all branches divided by the number of leaves.
All precise statistics are available in -S25.



a mis en forme : Droite : 1 cm

a mis en forme le tableau

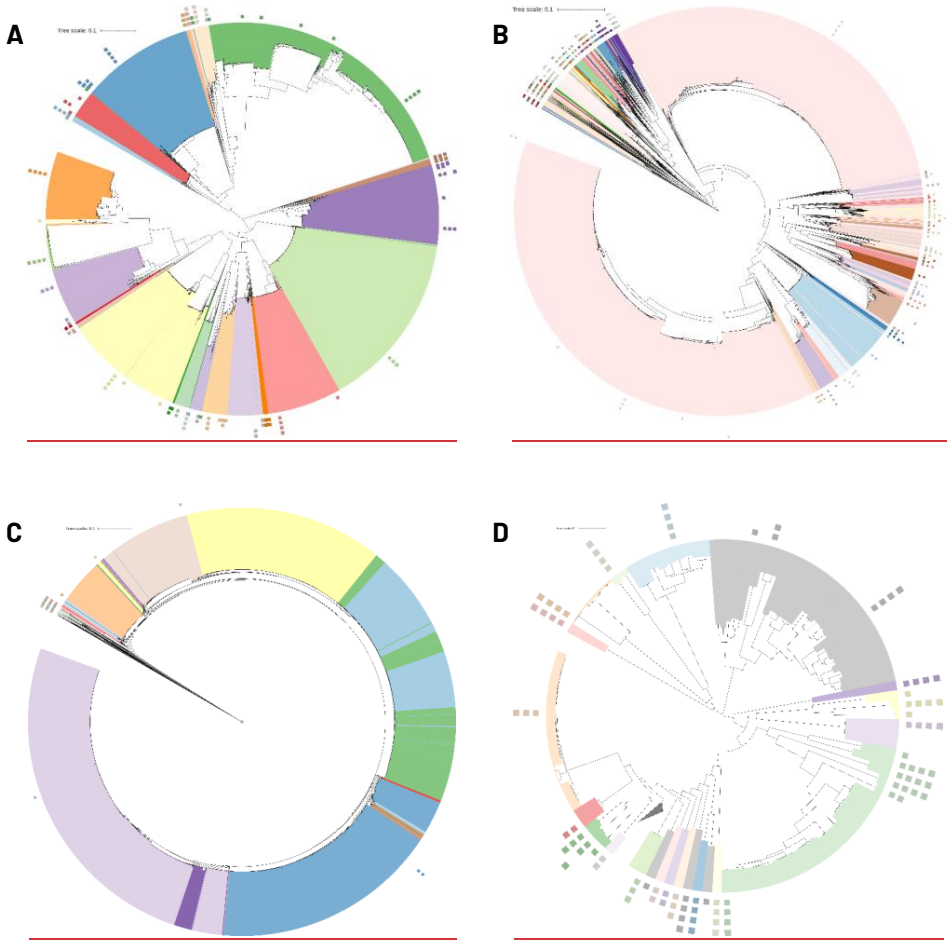
Supplementary Material S43 – Phylogenetic distribution of MPS-representatives of three taxonomic families

A: ML tree of the 6,410 *Lactobacillaceae* genomes present in the bacterial dataset. The tree has been inferred with r-prot sequences. From the innermost to the outermost circle: MPS-representatives corresponding to Δ=0.7 (80 MPS-representatives), Δ=0.6 (45 MPS-representatives), Δ=0.5 (24 MPS-representatives), and Δ=0.4 (12 MPS-representatives). Colors correspond to the 33 genera of *Lactobacillaceae*.

B: ML tree of the 7,113 *Bacillaceae* genomes present in the bacterial dataset. The tree has been inferred with r-prot sequences. From the innermost to the outermost circle: MPS-representatives corresponding to Δ=0.7 (249 MPS-representatives), Δ=0.6 (142 MPS-representatives), Δ=0.5 (78 MPS-representatives), and Δ=0.4 (39 MPS-representatives). Colors correspond to the 108 genera of *Bacillaceae*.

C: ML tree of the 17,096 *Enterobacteriaceae* genomes present in the bacterial dataset. The tree has been inferred with r-prot sequences. From the innermost to the outermost circle: MPS-representatives corresponding to Δ=0.7 (37 MPS-representatives), Δ=0.6 (28 MPS-representatives), Δ=0.5 (21 MPS-representatives), and Δ=0.4 (18 MPS-representatives). Colors correspond to the 58 genera of *Enterobacteriaceae*.

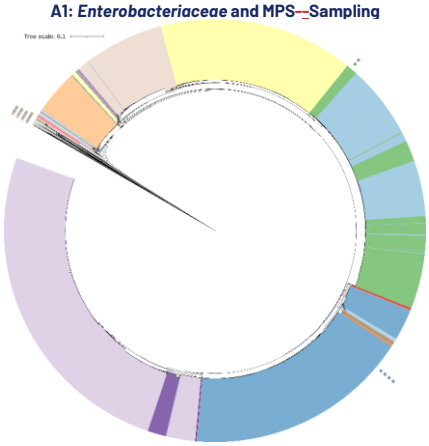
D: ML tree of the 17,096 *Enterobacteriaceae* genomes with 16,987 leaves collapsed. It highlights the 18 Candidatus genera contained within the *Enterobacteriaceae* genomes, representing the large majority of the MPS-samples: 27 out of 37 and 14 out of 18 MPS-representatives are Candidatus genomes, for Δ=0.7 and Δ=0.4 respectively.



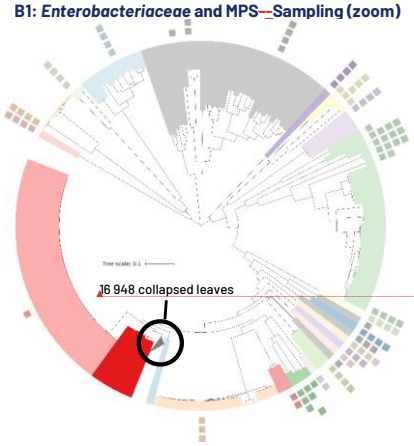
a mis en forme le tableau

Supplementary Material S44 – Phylogenetic distribution for *Enterobacteriaceaae*
Supplementary Material S4451 – Phylogenetic distribution for *Enterobacteriaceaae*
(legend on the next page)

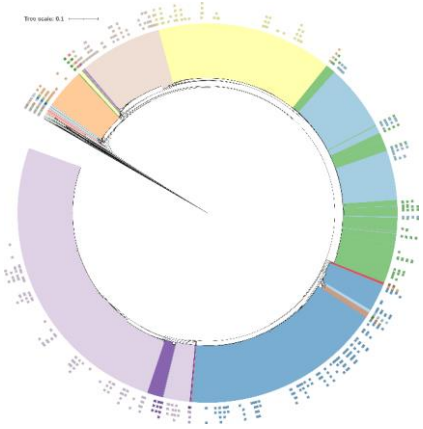
A1: *Enterobacteriaceae* and MPS—Sampling



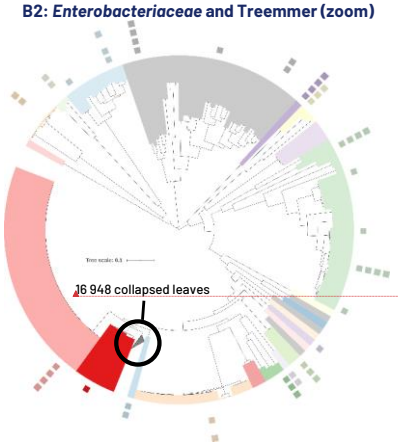
B1: *Enterobacteriaceae* and MPS—Sampling (zoom)



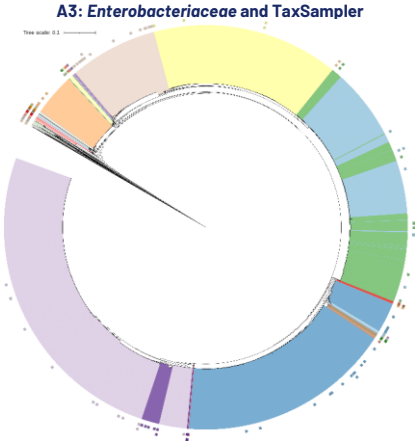
A2: *Enterobacteriaceae* and Treemmer



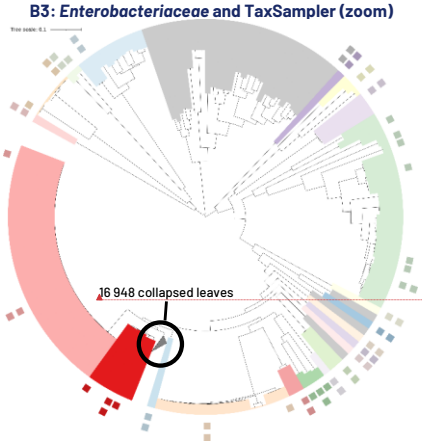
B2: *Enterobacteriaceae* and Treemmer (zoom)



A3: *Enterobacteriaceae* and TaxSampler



B3: *Enterobacteriaceae* and TaxSampler (zoom)



a mis en forme le tableau

a mis en forme : Anglais (États-Unis)

a mis en forme : Anglais (États-Unis)

a mis en forme : Anglais (États-Unis)

Phylogenetic tree inferred from the 17,096 genomes of *Enterobacteriaceae*. Leaves are colored according to the 58 genera.
A1: Four samples generated by MPS-Sampling are tagged around the phylogeny, for the respective values: $\Delta \in \{0.7; 0.6; 0.5; 0.4\}$, corresponding to 36, 28, 21 and 18 MPS-representatives, respectively.
A2: Four Treemmer-generated samples are tagged around the phylogeny, corresponding to 280, 195, 130 and 84 Treemmer-representatives respectively. The Treemmer samples studied are the same size as those from MPS-Sampling on Bacteria.
A3: Two samples generated by TaxSampler are tagged around the phylogeny, corresponding to 204 and 58 Tax-representatives respectively. The two TaxSampler samples correspond to both species and genus levels.

Tree of 17,096 *Enterobacteriaceae* genomes, with 16,948 collapsed leaves, in the small triangle at bottom left
B1: This triangle contains 2, 2, 1 and 1 MPS-representatives respectively.
The rest of the tree therefore contains 34, 26, 20 and 17 MPS representatives respectively.
B2: This triangle contains 263, 181, 119 and 77-Treemmer representatives respectively.
The rest of the tree therefore contains: 17, 14, 11 and 7 Treemmer representatives respectively.
B3: This triangle contains respectively: 168 and 36 Tax-representatives.
The rest of the tree therefore contains: 36 and 22 Tax-representatives respectively.

a mis en forme : Gauche : 2 cm, Haut : 2 cm

References

Adeolu,M. *et al.* (2016) Genome-based phylogeny and taxonomy of the 'Enterobacteriales': Proposal for enterobacterales ord. nov. divided into the families Enterobacteriaceae, Erwiniaceae fam. nov., Pectobacteriaceae fam. nov., Yersiniaceae fam. nov., Hafniaceae fam. nov., Morgane. *Int. J. Syst. Evol. Microbiol.*, **66**, 5575–5599.

Aguinis,H. *et al.* (2013) Best-Practice Recommendations for Defining, Identifying, and Handling Outliers. *Organ. Res. Methods*, **16**, 270–301.

Anaconda Documentation (2020) Anaconda Software Distribution.

Balaban,M. *et al.* (2019) TreeCluster: Clustering biological sequences using phylogenetic trees. *PLoS One*, **14**, 1–20.

Bateman,A. *et al.* (2017) UniProt: The universal protein knowledgebase. *Nucleic Acids Res.*, **45**, D158–D169.

Chun,J. *et al.* (2018) Proposed minimal standards for the use of genome data for the taxonomy of prokaryotes. *Int. J. Syst. Evol. Microbiol.*, **68**, 461–466.

Chun,J. and Rainey,F.A. (2014) Integrating genomics into the taxonomy and systematics of the Bacteria and Archaea. *Int. J. Syst. Evol. Microbiol.*, **64**, 316–324.

Dice,L.R. (1945) Measures of the Amount of Ecologic Association Between Species Author (s): Lee R . Dice Published by : Ecological Society of America Stable URL : http://www.jstor.org/stable/1932409. *Ecology*, **26**, 297–302.

Garcia,P.S. *et al.* (2021) A Comprehensive Evolutionary Scenario of Cell Division and Associated Processes in the Firmicutes. *Mol. Biol. Evol.*, **38**, 2396–2412.

Goris,J. *et al.* (2007) DNA-DNA hybridization values and their relationship to whole-genome sequence similarities. *Int. J. Syst. Evol. Microbiol.*, **57**, 81–91.

Gupta,R.S. *et al.* (2020) Robust demarcation of 17 distinct bacillus species clades, proposed as novel bacillaceae genera, by phylogenomics and comparative genomic analyses: Description of robertmurraya kyonggiensis sp. nov. and proposal for an emended genus bacillus limiting it o. *Int. J. Syst. Evol. Microbiol.*, **70**, 5753–5798.

Han,A.X. *et al.* (2019) Phylogenetic clustering by linear integer programming (PhyCLiP). *Mol. Biol. Evol.*, **36**, 1580–1595.

Harris,D. and Harris,S. (2012) Digital Design and Computer Architecture Kaufmann,M. (ed).

Jauffrit,F. *et al.* (2016) RiboDB Database: A Comprehensive Resource for Prokaryotic Systematics. *Mol. Biol. Evol.*, **33**, 2170–2172.

Köster,J. and Rahmann,S. (2012) Snakemake-a scalable bioinformatics workflow engine. *Bioinformatics*, **28**, 2520–2522.

Lan,R. and Reeves,P.R. (2002) Escherichia coli in disguise: Molecular origins of Shigella. *Microbes Infect.*, **4**, 1125–1132.

Land,M. *et al.* (2015) Insights from 20 years of bacterial genome sequencing. *Funct. Integr. Genomics*, **15**, 141–161.

Lassmann,T. (2020) Kalign 3: Multiple sequence alignment of large datasets. *Bioinformatics*, **36**, 1928–1929.

Letunic,I. and Bork,P. (2021) Interactive tree of life (iTOL) v5: An online tool for phylogenetic tree display and annotation. *Nucleic Acids Res.*, **49**, W293–W296.

Lewis,W.H. *et al.* (2020) Innovations to culturing the uncultured microbial majority. *Nat. Rev. Microbiol.*, **19**, 225–240.

Maayer,P. De *et al.* (2019) Reorganising the order Bacillales through phylogenomics. *Syst. Appl. Microbiol.*, **42**, 178–189.

Menardo,F. *et al.* (2018) Treemmer: A tool to reduce large phylogenetic datasets with minimal loss of diversity. *BMC Bioinformatics*, **19**, 1–8.

O'Leary,N.A. *et al.* (2016) Reference sequence (RefSeq) database at NCBI: Current status, taxonomic expansion, and functional annotation. *Nucleic Acids Res.*, **44**, D733–D745.

Ondov,B.D. *et al.* (2016) Mash: Fast genome and metagenome distance estimation using MinHash. *Genome Biol.*, **17**, 1–14.

Parks,D.H. *et al.* (2020) A complete domain-to-species taxonomy for Bacteria and Archaea. *Nat. Biotechnol.*, **38**, 1079–1086.

Parks,D.H. *et al.* (2022) GTDB: An ongoing census of bacterial and archaeal diversity through a phylogenetically consistent, rank normalized and complete genome-based taxonomy. *Nucleic Acids Res.*, **50**, D785–D794.
Parks,D.H. *et al.* (2017) Recovery of nearly 8,000 metagenome-assembled genomes substantially expands the tree of life. *Nat. Microbiol.*, **2**, 1533–1542.
Patel,S. and Gupta,R.S. (2020) A phylogenomic and comparative genomic framework for resolving the polyphyly of the genus bacillus: Proposal for six new genera of bacillus species, peribacillus gen. nov., cytobacillus gen. nov., mesobacillus gen. nov., neobacillus gen. nov., metabacillu. *Int. J. Syst. Evol. Microbiol.*, **70**, 406–438.
Philippe,H. *et al.* (2017) Pitfalls in supermatrix phylogenomics. *Eur. J. Taxon.*, **2017**, 1–25.
Pipes,L. and Nielsen,R. (2022) AncestralClust: clustering of divergent nucleotide sequences by ancestral sequence reconstruction using phylogenetic trees. *Bioinformatics*, **38**, 663–670.
Price,M.N. *et al.* (2010) FastTree 2 - Approximately maximum-likelihood trees for large alignments. *PLoS One*, **5**.
Qin,Q.L. *et al.* (2014) A proposed genus boundary for the prokaryotes based on genomic insights. *J. Bacteriol.*, **196**, 2210–2215.
Ramulu,H.G. *et al.* (2014) Ribosomal proteins: Toward a next generation standard for prokaryotic systematics? *Mol. Phylogenet. Evol.*, **75**, 103–117.
Slavík,P. (1996) A tight analysis of the greedy algorithm for set cover. *Proc. Annu. ACM Symp. Theory Comput.*, **Part F1294**, 435–441.
Song,I. *et al.* (1995) A Comparative Analysis of Entity-Relationship Diagrams. *J. Comput. Softw. Eng.*, **3**, 427–459.
Sørensen,T. (1948) A method of establishing groups of equal amplitude in plant sociology based on similarity of species content and its application to analyses of the vegetation on Danish commons.
Steinegger,M. and Söding,J. (2018) Clustering huge protein sequence sets in linear time. *Nat. Commun.*, **9**.
Steinegger,M. and Söding,J. (2017) MMseqs2 enables sensitive protein sequence searching for the analysis of massive data sets. *Nat. Biotechnol.*, **35**, 1026–1028.
Stephens,Z.D. *et al.* (2015) Big data: Astronomical or genomical? *PLoS Biol.*, **13**, 1–11.
Sultanpuram,V.R. and Mothe,T. (2016) Salipaludibacillus aurantiacus gen. Nov., sp. nov. a novel alkali tolerant bacterium, reclassification of Bacillus agaradhaerens as Salipaludibacillus agaradhaerens comb. nov. and Bacillus neizhouensis as Salipaludibacillus neizhouensis comb. nov. *Int. J. Syst. Evol. Microbiol.*, **66**, 2747–2753.
Yates,A.D. *et al.* (2022) Ensembl Genomes 2022: An expanding genome resource for non-vertebrates. *Nucleic Acids Res.*, **50**, D996–D1003.
Yutin,N. *et al.* (2012) Phylogenomics of Prokaryotic Ribosomal Proteins. *PLoS One*, **7**, e36972.
Zheng,J. *et al.* (2020) A taxonomic note on the genus Lactobacillus: Description of 23 novel genera, emended description of the genus Lactobacillus beijerinck 1901, and union of Lactobacillaceae and Leuconostocaceae. *Int. J. Syst. Evol. Microbiol.*, **70**, 2782–2858.