Accurate reconstruction of bacterial pan- and core- genomes with PEPPA

Authors: Zhemin Zhou* and Mark Achtman

Affiliations:

Warwick Medical School, University of Warwick, Gibbet Hill Road, Coventry, CV4 7AL, United Kingdom

*Corresponding author: Z.Z.: Zhemin.Zhou@warwick.ac.uk

ORCID:

ZZ: 0000-0001-9783-0366

MA: 0000-0001-6815-0070

Abstract

Bacterial genomes have been shaped by a complex evolutionary history, including extensive homologous recombination, horizontal gene transfer, gene loss, and gene duplications. A pan-genome, consisting of all genes within a defined set of bacterial genomes, can provide the basis for phylogenetic inferences and population studies. Here we introduce PEPPA, a pipeline that can construct a pan-genome from thousands of genetically diversified bacterial genomes. PEPPA implements a combination of tree- and synteny-based approaches to identify and exclude paralogous genes, which facilitates the ready construction of schemes for whole genome multi-locus sequence typing (wgMLST) and core genome MLST. PEPPA also implements similarity-based gene predictions that support consistent annotations of genes and pseudogenes in individual genomes, and which are nearly as accurate as expert manual curation. The PEPPA package includes PEPPA_parser, which calculates trees based on accessory gene content and MLST core genome allelic differences. We also present SimPan, a novel pipeline for simulating the evolution of a bacterial pan-genome. PEPPA was compared with four state-of-the-art pan-genome pipelines on both empirical and simulated datasets. It demonstrated greater accuracy and specificity than any other pipeline, and was almost as fast as any of them at calculating a pan-genome. To demonstrate its abilities, PEPPA was used to construct a *Streptococcus* pan-genome of >40,000 genes from 3,170 representative genomes spanning at least 80 species. The resulting gene and allelic trees provide an unprecedented overview of the genomic diversity of this genus.

Introduction

After the first bacterial genome was sequenced (Fleischmann *et al.* 1995), it soon became clear that the genomic contents differed between individual strains within a prokaryotic species. Variable genomic content is caused by the gain or loss of singleton ORFan genes (Daubin and Ochman 2004), genomic islands, selfish DNA (plasmids, bacteriophages, integrative chromosomal elements) and/or widespread horizontal gene transfer (Abby *et al.* 2012; Szöllösi *et al.* 2012; Croucher *et al.* 2014). The accessory genome consists of genes which are variably present in individual genomes, whereas

the core genome consists of those orthologs that are present in all genomes, and often contains phylogenetic signals reflecting the vertical accumulation of mutations. However, although these concepts were widely known, it was still a major shock when the concept of an open pan-genome was introduced (Tettelin *et al.* 2005), which corresponds to the sum of all genes in both the core and accessory genomes, and whose size increases indefinitely with the number of genomes sequenced.

The original concept of an open pan-genome was based on a genomic comparison of six strains of *Streptococcus agalactiae* (Tettelin *et al.* 2005). Very few of the subsequent pan-genome analyses included more than 100 genomes (Vernikos *et al.* 2015), in part because only few genomes had been sequenced. Furthermore, the pan-genome construction algorithms that were available at that time (OrthoMCL (Li *et al.* 2003); Panseq (Laing *et al.* 2010); PGAP (Zhao *et al.* 2012)) were not able to handle larger numbers of genomes: they were based on an initial all-against-all sequence comparison, which scales computationally with the squared number of gene sequences.

The numbers of genomes per microbial species have increased very dramatically over the last decade, and databases now exist that include 100,000s of assembled genomes from a single bacterial genus (Sanaa *et al.* 2019; Zhou *et al.* 2020). However, the question of an open pan-genome has not yet been revisited with these large datasets. Similarly, earlier publications indicated fewer and fewer genes within the core-genome as more and more genomes were examined, but this has also not been revisited with the larger datasets. Finally, genetic relationships that are based on single nucleotide polymorphisms are extremely difficult to calculate in real-time with very large datasets, and some large databases focus on core genome Multilocus sequence typing (cgMLST) (Mellmann *et al.* 2011; Moura *et al.* 2016; Alikhan *et al.* 2018; Frentrup *et al.* 2019; Zhou *et al.* 2020) for genotyping and recognizing relationships. cgMLST schemes within public databases calculate the allelic variants within a fixed set of core genes. Defining that fixed set is best performed on the basis of a large and representative dataset of genomes in ordert to avoid the necessity for database-wide changes as additional genomes are genotyped. Thus, the creation of cgMLST schemes depends on the availability of software pipelines that can elucidate the entire pan-genome from relatively

large numbers of genomes. Fortunately, three such pipelines have recently been developed (Roary (Page *et al.* 2015); PanX (Ding *et al.* 2018); PIRATE (Bayliss *et al.* 2019)).

We now address a second and largely unexplored problem for elucidating a pan-genome. Until recently, genome annotations in public repositories were inconsistent, resulting in different predictions from genome to genome of the locations of start codons of open reading frames, the identification of pseudogenes and the presence or absence of individual genes (Wozniak *et al.* 2014). The earliest pan-genome analyses (Tettelin *et al.* 2005; Hogg *et al.* 2007) addressed this problem by an independent round of TBLASTN gene-against-genome comparisons, but such inconsistencies between genome annotations are not directly addressed by the latest generation of pan-genome pipelines, except for LS-BSR (Sahl *et al.* 2014). Instead, the other current tools depend on the re-annotation of genomes with independent *ab initio* gene predictors, such as Prodigal (Hyatt *et al.* 2010) and PROKKA (Seemann 2014). The identification of pseudogenes requires comparison against intact analogs (Lerat and Ochman 2005), but this is not performed by these *ab initio* gene predictors. Instead, they annotate fragments of some pseudogenes as intact genes, leading to inconsistent annotations of orthologs and an overestimation of the size of the pan-genome.

In order to create reliable cgMLST schemes for EnteroBase (Frentrup *et al.* 2019; Zhou *et al.* 2020), we have developed PEPPA, a novel pipeline for calculating a pan-genome that deals specifically with the problems described above. Here we describe the algorithms implemented within PEPPA, and show that it outperforms other pan-genome methods with empirical as well as with simulated datasets. As a demonstration of its capacities, we also demonstrate that PEPPA can identify a pan-genome of >40,000 genes from 3170 representatives of *Streptococcus*, a highly diverse genus.

Results

A brief overview of PEPPA

The procedures performed by PEPPA can be sub-divided into four successive groups of operations (Supplemental Fig. S1), as follows. Details are presented in Appendix 1.

(1) Identifying representative gene sequences. The GFF3 format (Ensembl Release 98 2019) is a commonly used format for genomic annotations of genes, and PEPPA depends on this format for extracting genes from assembled genomes. It then iteratively assigns genes to clusters with Linclust, a linear gene clustering function implemented in MMSeqs2 (Steinegger and Soding 2017), and extracts a reduced gene set consisting of one representative gene sequence per cluster.

(2) Identifying all gene candidates in all genomes. Each representative gene sequence is aligned to all the genomes using both BLASTn (Altschul *et al.* 1990) and DIAMOND (Buchfink *et al.* 2015). All hits whose homology spans at least 50% of the length of each representative gene are extracted by PEPPA as putative gene candidates.

(3) Assignments to sets of orthologous genes. The putative gene candidates for each representative gene which form monophyletic subtrees according to their phylogenetic topologies are assigned to clusters of orthologs. To this end, the average genome similarities and their standard deviations are calculated from a lognormal distribution of the sequence similarities of the 1,000 most common sets of putative gene candidates. Trees are also constructed from the alignments of gene candidates for each representative gene using RapidNJ (Simonsen *et al.* 2011). A comparison of the pairwise similarities between the gene candidates with the similarities of their source genomes is used to calculate paralogous scores for each branch in the gene trees, and branches with a paralogous score of >1 (see Appendix 1.6) are iteratively pruned until none remain. The remaining subtrees correspond to monophyletic groups.

The genomic locations of multiple putative gene candidates from distinct monophyletic groups may overlap in some genomes. These conflicts are resolved by retaining only the monophyletic group in each conflicted region with the greatest information score (see Appendix 1.7), and eliminating all other gene candidates for that region.

The remaining gene candidates from each genome are ordered according to their genomic coordinates, and the final sets of orthologous genes are identified based on synteny. To this end, a minimum spanning tree is used to identify and remove recent gene duplications based on the genomic synteny of up to three upstream and three downstream neighbors of every gene candidate.

(4) Pseudogene prediction and outputs. The coding region of each remaining gene candidate in each genome is categorized as an intact coding sequence (CDS) or a pseudogene, depending on the size of the aligned reading frame relative to the representative gene. The evaluations of all genes are saved in GFF3 format, and the extent of matched regions to their representative genes are saved in FASTA format.

(5) Analytical results and cgMLST schemes. A separate tool, PEPPA_parser, generates analytical results such as rarefaction curves, gene presence matrices, gene presence trees, and core gene trees based on the GFF outputs from PEPPA. These have been the basis for all core genome MLST schemes in EnteroBase (Zhou *et al.* 2020). Because PEPPA calculates core genes from ortholog clusters based on large numbers of genomes, the cgMLST schemes avoid problems with paralogs that may plague cgMLST schemes that are based exclusively on single-BLAST hits (Appendix 3).

Comparisons of PEPPA with other recent pan-genome pipelines

We assessed the absolute performance of PEPPA, and also compared it with other, recently described pipelines for pan-genome construction (Roary (Page *et al.* 2015); PanX (Ding *et al.* 2018); PIRATE (Bayliss *et al.* 2019)) as well as with a classical, small-scale pipeline (OrthoMCL, (Li *et al.* 2003)). However, it was important to examine multiple aspects of genomic diversity for these comparisons because the evolutionary history of bacterial pan-genomes can be highly complex. The pan-genome reflects the accumulation of mutations during vertical inheritance, admixture due to homologous recombination with DNA from unrelated strains, the insertion by horizontal gene transfer (HGT) of novel genes from outside the species, as well as gene loss due to deletion events. Indels in intra-genic and inter-genic regions differ in their evolutionary dynamics due to differential levels of diversity and selection pressures. Furthermore, single copy orthologs need to be distinguished from paralogs which arise by the insertion of DNA regions at more than one location in a genome.

We were not aware of any pre-packaged simulation tools that can encompass this entire diversity of genetic changes. We therefore performed our first benchmarks on a pan-genome from 15 *Salmonella enterica* genomes, and compared the pan-genomes calculated from manual annotations which had been performed in order to ensure

consistency across the genomes (Nuccio and Baümler 2014) with pan-genomes based on automated annotations of the same genomes. Secondly, we designed a new simulation tool, SimPan, which uses SimBac (Brown *et al.* 2016) to simulate the dynamics of pan-genome evolution *via* recombination, HGT, gene gain and loss as well as the creation of paralogs (Appendix 2).

The recent pipeline LS-BSR (Sahl *et al.* 2014) was not included in the comparisons because it does not report genomic coordinates for predicted pan genes, and could therefore not be evaluated by our benchmarks. We also only report results from the best of three runs (minimum identity = 50, 80 or 95%) with Roary or PIRATE because the optimal value of this parameter differs for various levels of diversity ((Ding *et al.* 2018) and our own observations).

Benchmarking pan-genome pipelines on 15 curated genomes

Nuccio and Baümler re-annotated 15 complete genomes of *S. enterica* (Nuccio and Baümler 2014). They removed existing annotations for unreliable short genes, performed new BLASTN and TBLASTN alignments to identify previously non-annotated genes, corrected the start positions of falsely annotated genes and predicted the existence of pseudogenes based on their alignments with orthologous intact CDSs. The result of these efforts is a unique set of consistently annotated genomes from a single species, which we equated with the ‘ground truth’ for the individual pan-genome pipelines.

We compared the manual re-annotation with three sets of gene annotations for each of the 15 *S*. *enterica* genomes: (1) the original “submitter” annotation that had been submitted to GenBank, (2) an automated re-annotation from RefSeq (Haft *et al.* 2018) that was generated with PGAP (Tatusova *et al.* 2016), (3) a novel annotation that we performed for each genome using PROKKA (Seemann 2014), another popular bacterial annotation pipeline. Unreliable genes that had been eliminated by Nuccio and Baümler were removed from all three annotations for consistency. We then examined the degree of concordance between the pan-genomes calculated by each of the pipelines with the pan-genome published by Nuccio and Baümler by calculating the adjusted Rand index (ARI) (Rand 1971), which is a measure of similarity between clustering results.

All pipelines successfully calculated a pan-genome from each of the four annotations, except that PanX never completed the “Submitter” annotations. The PEPPA pan-genomes consistently yielded ARIs of approximately 0.98 relative to the manual pan-genome (Fig. 2A, histograms). This consistency is not surprising because PEPPA re-calculates gene annotations in a fashion which resembles that of the manual curation. All the other tools yielded lower ARI values, and these ARI values depended on the source of the annotations. The PROKKA annotations yielded ARI scores of approximately 0.97 with Roary, PanX, and OrthoMCL and slightly less with PIRATE. The ARI scores were 0.95-0.96 for the PGAP annotations from RefSeq, and even lower for the original annotations (Submitter). We also performed hierarchical clustering using the Neighbor-Joining algorithm on pair-wise comparisons of the ARI scores across all 14 pan-genomes (Fig. 3B). The three pan-genomes predicted by PEPPA formed a tight cluster with high pair-wise ARI (0.99), and which clustered tightly with the curated pan-genome (ARI=0.98). In contrast, the pan-genomes generated by the other pipelines clustered together according to annotation sources rather than pipeline methodology. For each of the three annotation sources, the pan-genome predicted by Roary was the most distinct whereas the results were more comparable between the pan-genomes predicted by OrthoMCL, PanX, and PIRATE. These results may reflect the fact that Roary differs from the other pipelines by an additional splitting of paralogs on the basis of synteny.

Predictions of pseudogenes. Pan-genomes contain pseudogenes in addition to intact CDSs, and identifying pseudogenes within pan-genomes is important because they are relevant to cell function (Goodhead and Darby 2015) and evolutionary biology (Koonin 2016). Within *Salmonella*, the accumulation of pseudogenes has been linked to host specialization (Parkhill *et al.* 2001; Holt *et al.* 2008; Nuccio and Baümler 2014; Zhou *et al.* 2014; Zhou *et al.* 2018b). The core genome defined by Nuccio and Baümler contained 2,838 CDSs that were intact in all 15 genomes and 783 others that were disrupted in at least one genome. PEPPA predicted marginally more intact CDSs, and slightly fewer pseudogenes, from all three annotations than were present in the manual annotations (Fig. 2A, circles). The numbers of pseudogenes within each of the 15

genomes were also very similar between the manual curations and PEPPA's automated predictions (Fig. 2C).

Roary, OrthoMCL and PanX do not predict disrupted genes at all. PIRATE report 'gene diffusion', a measure of the frequency with which intact CDSs in some genomes are split into two or more fragments in others. However, it did not detect any diffusion in the RefSeq and GenBank annotations, and only one example in the PROKKA annotation. Thus, PIRATE also only reported core CDSs. Similar to the ARI comparisons described above, the total numbers of predicted core CDSs varied according to the annotation source for all three pipelines other than PEPPA. The four pipelines reported 3,301-3,515 core CDS from the PROKKA annotations (Fig. 2A right). These numbers are similar to the total number of core plus pseudogene CDSs within the curated pan-genome, indicating that PROKKA has predicted most pseudogenes as intact CDSs. Roary, PIRATE and OrthoMCL only detected 2,418-2,484 core genes in the originally submitted genomes, providing further support for poor consistency in those annotations. In contrast, all four pipelines predicted 2,901-2,957 core CDSs from the RefSeq annotations, and these numbers were similar to the numbers of intact core CDSs in the curated pan-genome (2,838), or as predicted by PEPPA (2,918-2,961).

Inaccurate predictions of orthologs. We investigated the different numbers of predicted core CDSs between the four pipelines by identifying the inconsistent ortholog calls relative to the manually curated pan-genome (Nuccio and Baümler 2014). We designated as 'false splits' those cases where an ortholog cluster in the curated pan-genome was split into multiple orthologous clusters by a pipeline. Similarly, 'false merges' was used to designate pipeline assignments of genes from multiple orthologous clusters according to the curated pan-genome to a single orthologogous cluster. We also assigned the designation 'backbone genes' to 4,695 genes in the curated pan-genome which were descendants from the most recent common ancestor (MRCA) (Fig. 2D), and 'mobile genes' to the other 3,364 genes which were associated in one or more genomes with prophages, insertion sequences, plasmids or ICE/IMEs, and were absent in the MRCA (Fig. 2E). PEPPA made the fewest false splits and false merges of all five pipelines, followed by PanX (Fig. 2D). False merges were made four

times as often by all pipelines for mobile genes than for backbone genes, and false splits were up to two times as often (Fig. 2E *vs.* 2D). Roary generated the most false calls and PEPPA generated the fewest.

Generation of simulated datasets

The analyses just presented indicate that the backbone and mobile genes might differ in their evolutionary rates of gain and loss. However, we could not test the abilities of the pipelines to handle varying evolutionary rates because we were not aware of any tool that can simulate the evolution of a pan-genome based on independent rates of change of these two categories. We therefore created SimPan (https://github.com/zheminzhou/SimPan) in order to be able to simulate the genetic features of real bacterial pan-genomes (Supplemental Fig. S2 and Table S5; see Appendix 2 for details). In brief, SimPan uses SimBac (Brown *et al.* 2016) to generate a clonal genomic phylogeny, whose topology is subsequently disrupted by random homologous recombination until every single site of the backbone and mobile genes belongs to a distinct local tree. Place holders for backbone genes and mobile genes, and their surrounding intergenic regions, were incorporated into the simulations, and the marked regions were subjected to random deletion/insertion events until the average number of genes in each genome and in the core genome attained user-specified parameters. Finally, sequences of both genes and intergenic regions were subjected to short indels and converted into genes with INDELible (Fletcher and Yang 2009), which were then concatenated into whole genomes.

For the pan-genome simulations, we used parameters derived from the curated *S. enterica* pan-genome, and generated five sets of 15 simulated genomes (a to e) with SimPan, with each genome containing a mean of 3,621 core genes and 879 accessory genes. We arbitrarily assigned 5% of the backbone genes to paralogous clusters, and 40% of the mobile genes, and varied their mean percentage sequence identities between each set of simulations (Fig 3, inset). Simulation c was the simplest pan-genome construction scenario, with high sequence identity (98%) between genes in an ortholog cluster and low sequence identity (60%) between genes in a paralog cluster. Simulations a and b had decreasing levels of identity between orthologs to simulate

more diverse species while simulation d and e had increasing levels of identity between paralogs in order to simulate recent gene duplications.

Pipeline performance with simulated genomes

The pan-genomes calculated from the simulated data by PEPPA, Roary, PIRATE, PanX, and OrthoMCL were compared to the original pan-genome generated by SimPan (Fig. 3A). Once again, PEPPA outperformed all other tools (ARI ≥0.998). PEPPA correctly predicted all core genes in simulations b, c, and e, and only missed 2-3 core genes in the two remaining datasets. Roary was very close behind PEPPA on all simulated datasets by both criteria. However, Roary failed to identify any of the core genes as being present in multiple copies, likely due to its aggressive synteny-based paralogous splitting step. PIRATE performed almost as well with simulations c to e, but yielded lower ARI scores with the more divergent genomes in simulations a and b. In contrast, PanX and OrthoMCL yielded high ARI scores with the divergent genomes in simulations a and b, but poorer scores at higher levels of identity between paralogous genes (simulations d and e).

The pan-genomes created by PIRATE, PanX, and OrthoMCL contained more core genes when all multi-copy and single-copy core genes were counted than when only single-copy core genes were counted, suggesting potential false splits of paralog clusters. Indeed the frequency of false splits was particularly high for backbone genes with these three pipelines, and the frequency of false merges was high with Roary and OrthoMCL (Fig. 3B). All five pipelines made multiple false merges with mobile genes, possibly because of their predominance among paralogous clusters, and Roary made large numbers of false splits (Fig. 3C). Overall, PEPPA made fewer false splits than Roary and fewer false merges than the other three pipelines for both backbone and mobile genes, which explains its higher ARI scores.

The effects of missing gene annotations on the pan-genome. As shown above, inconsistent or inaccurate gene predictions within annotations are problematical for calculating reliable pan-genomes. We simulated this effect by randomly deleting 0.1%, 1% or 2% of the gene predictions and their annotations in simulation c (Fig 3D). Because PEPPA relies on its own assignments of individual genes to ortholog clusters,

it was unaffected by these manipulations. However, the missing data resulted in drastically reduced ARI scores (Fig. 3D. histograms) and core genome sizes (Fig. 3D, circles) for the other pipelines, and these became progressively worse with the proportion of missing annotations.

Pipeline speed. We generated 10 additional simulations of 20 to 200 genomes with the same parameters as simulation c, and measured the running times for all five pipelines to calculate a pan-genome with 4 processes on a server with 1 TB of memory and 40 CPU cores (Fig. 3E). OrthoMCL needed >24 h for ≥60 genomes. PanX was at least eightfold slower than the other three pipelines, and needed 500 min for 200 genomes, despite its activation of the divide-and-conquer algorithm on datasets with >50 genomes. Both Roary and PIRATE scaled well, and each completed the calculations on 200 genomes within 30 minutes. PEPPA is about twice as slow as either Roary or PIRATE, and needed 63 minutes for 200 genomes. The good scalability of these tools likely relates to the pre-clustering step, which reduces the number of genes in downstream comparisons. However, this pre-clustering step becomes less efficient with increased genetic diversity: in an independent simulation of 200 genomes with only 90% sequence identity, the runtime for all three pipelines increased by at least twofold relative to simulation c (PEPPA: 144 min; Roary: 132; PIRATE: 60).

A pan-genome for the genus *Streptococcus*

PEPPA can construct a pan-genome from thousands of genomes with high genetic diversities, and earlier versions of this pipeline were used to generate cgMLST schemes for the genera represented in EnteroBase (Alikhan *et al.* 2018; Frentrup *et al.* 2019; Zhou *et al.* 2020) as well as for ancient DNA analyses (Zhou *et al.* 2018b; Achtman and Zhou 2019). Here we address the genus *Streptococcus*, which includes very genetically diverse strains and highly significant zoonotic and human pathogens (Gao *et al.* 2014), to demonstrate how to use PEPPA to construct a pan-genome from representative genomes.

The pair-wise genetic distances between 14,115 draft and complete *Streptococcus* genomes in RefSeq (June 2019) (Supplemental Table S1) were calculated with MASH (Ondov *et al.* 2016), and those genomes were grouped into clusters by a single-linkage

clustering algorithm using a MASH distance cutoff of 0.002 (equivalent to 99.8% average nucleotide identity [ANI99.8%]). We chose one representative genome per cluster to create a dataset of 3170 genomes (Supplemental Table S2) that represents the entire taxonomic diversity of *Streptococcus.* PEPPA needed 12 days to construct a *Streptococcus* pan-genome from this dataset. The resulting pan-genome contains 41,026 genes, twice as many as a previous pan-genome that was based on 138 *Streptococcus* genomes (Gao *et al.* 2014). In agreement with the earlier conclusions by Gao *et al*., the rarefaction curve does not show any signs of a plateau, and the *Streptococcus* pan-genome continues to expand in size with each new genome (Fig 4A). Thus, the pan-genome remains open (Tettelin *et al.* 2005). Gao *et al*. estimated that the pan-genome would expand by 62 genes for each new genome, and we estimated a similar rate of 58 for a random set of 138 genomes. However, the growth rate has dropped dramatically with the increased number of genomes, and we estimate that the future expansion rate of the pan-genome is only 1.7 new genes for every newly added representative genome.

In contrast to the earlier studies (Gao *et al.* 2014), which defined a strict core genome of 278 orthologs, we find no evidence for a classical strict core genome for *Streptococcus*. Only 16 core genes were shared across all representative genomes (Fig 4B, inset), and each of them was disrupted in at least one *Streptococcus* genome in RefSeq. This has become a common problem for core genome schemes, especially because draft genomes consist of multiple contigs, which can in turn result in the random absence of multiple genes from the genome. Core genome schemes used for cgMLST are therefore usually based on a relaxed core, consisting of single-copy genes present in the large majority of representative isolates (Moura *et al.* 2016; Alikhan *et al.* 2018; Zhou *et al.* 2020). Our analyses identified 777 genes that were present in at least 3020 (95%) of the representative streptococcal genomes (Supplemental Table S4, Fig. 4B). However, most of the 777 genes were present in multiple copies in some genomes, leaving a final relaxed core of 284 single-copy genes that were suitable for identifying core genomic relationships and evolutionary history (Table 1; Supplemental Table S4).

The taxonomy of *Streptococcus* is a highly dynamic area of research (Kikuchi *et al.* 1995; Jensen *et al.* 2016; Dekker and Lau 2016; Velsko *et al.* 2018; Velsko *et al.* 2019; Kilian and Tettelin 2019; Zhou *et al.* 2020). Many *Streptococcus* species are currently defined exclusively by phenotypic markers, and multiple taxonomic assignments in RefSeq are incorrect (Beaz-Hidalgo *et al.* 2015; Gomila *et al.* 2015; Kilian and Tettelin 2019). We therefore initially ignored the taxonomic designations, and used the somewhat arbitrary cut-off of 95%ANI as a proxy for species diversity (Konstantinidis *et al.* 2017; Jain *et al.* 2018). Single-linkage clustering of the pair-wise similarities calculated between the 3170 representative genomes by FastANI (Jain *et al.* 2018) revealed 227 agglomerative clusters, which we designated MG_1 through MG_227 (Supplemental Table S2). For 29 clusters that each contains at least ten genomes, we also identified the dominant species designation in NCBI metadata, as shown in Supplemental Table S3. Information on their pan-genomic contents can be found in Supplementary Material and Supplemental Table S4.

Taxonomical clusters within *Streptococcus*

We used PEPPA_parser to generate a tree of the 3170 representative genomes based on the presence or absence of the 41,026 pan genes (Fig. 5A). We also generated a second tree with PEPPA_parser from the same genomes, but based on the allelic variations within the 284 relaxed core genes. The topology of the first tree reflects the similarities in pan-genomic content and the topology of the second tree reflects the sequence similarities within core genes. Not surprisingly, the detailed topologies differed somewhat between the two trees. In particular, the core gene tree (Fig, 5B) contains an unresolved, star-like radiation which we attribute to the lack of allelic identities between genes from highly diverse species. However, despite their differences in deep branching topology, both trees in Fig. 5 showed tight clustering of genomes within each of the 29 common taxonomic groupings. This tight clustering indicates that the topologies of both trees are congruent at the ANI95% levels. Both trees also supported some published taxonomic assignments of subspecies, for example, MG_29 corresponds to *S. gallolyticus* and includes its three subspecies, *gallolyticus, macedonicus,* and *pasteurianus* (Dekker and Lau 2016). Similarly, MG_2 corresponds to *S. dysgalatiae* and includes the two subspecies (*dysgalatiae* and *equisimilis* (Jensen and Kilian 2012).

Both *Streptococcus* trees also clustered high order branches according to the traditional taxonomical group names Mitis, Anginosus, Salivarius, Mutans, Bovis, and Pyogenic (Gao *et al.* 2014). They clustered *S. suis* together in a seventh phylogenetic branch, which we designate as Suis, and also clustered *S. acidominimus*, *S. minor*, *S. hyovaginalis, S. ovis,* and multiple other taxa into a novel, unnamed neighboring branch. With the help of PEPPA_parser, we identified a group-specific pan-genome for each of the seven named taxonomical groups. Similar to the *Streptococcus* pan-genome, each sub- pan-genome is open (Fig. 4C-D), and is growing at the rate of 2.5-12.7 new genes for each new representative genome. Unlike the entire genus, these seven named taxonomic groups possess a strict core genome, consisting of 241 to 1152 core genes (Fig. 4C-D, Table 1). However, once again, many core genes are present in multiple copies in some genomes, and would not suitable for a cgMLST scheme. After excluding multi-copy genes, the sizes of the group-specific, 95% relaxed core genome ranged from 400 (Mutans) to 665 (Suis) genes (Fig. 4C-D, Table 1).

Similar to prior observations (Kilian *et al.* 2008; Kilian and Tettelin 2019), there are numerous discrepancies between the ANI95% groups and the taxonomic designations in RefSeq. Some discrepancies reflect inaccurate metadata but others reflect true discrepancies between ANI95% clusters and taxonomic designations made by expert microbiologists. For example, *S. mitis* spans 50 distinct ANI95% clusters (Fig. 5, Supplemental Table S3). Similarly, *S. oralis* straddles multiple, distinct ANI95% clusters, as did each of the three *S. oralis* subspecies *oralis, tigurinus* and *dentisani* defined by Jensen *et al.* (Jensen *et al.* 2016). Further investigations will be needed to elucidate just how many biological species are truly present within the genus *Streptococcus.* We anticipate that the trees in Fig. 5 might be useful for such analyses.

Discussion

Appendix 1 describes the algorithms used by PEPPA for rapidly calculating accurate pan-genomes and relaxed core genomes. Similarly, Appendix 2 describes SimPan, a second pipeline, which simulates the evolution of pan-genomes by a combination of mutations, deletions, homologous recombination with related organisms, horizontal gene transfer from unrelated microbes plus the formation of pseudo-genes. PEPPA was

compared with other pipelines on manually curated genomic annotations as well as on evolved populations of simulated genomes, and was found to be superior in all comparisons. We then demonstrated the capabilities of PEPPA by calculating a pan-genome based on representatives of more than 14,000 draft and complete genomes of *Streptococcus* from RefSeq.

Comparison of PEPPA with other pan-genome pipelines. All state-of-the-art pan-genome pipelines need to be efficient in order to handle the computational demands of modern, large-scale comparative genomics. Roary (Page *et al.* 2015) and PIRATE (Bayliss *et al.* 2019) were the fastest of all the pipelines tested, and their computational runtime scaled approximately linearly with the number of samples (Fig. 3E). Either pipeline can predict a pan-genome from 200 genomes in less than one hour, likely reflecting the fact that both use a pre-clustering step to reduce the number of genes that are analyzed in the subsequent, very time-consuming all-against-all comparison. A similar divide-and-conquer heuristic has also been implemented in PanX (Ding *et al.* 2018), but it seems to be less computationally efficient (Fig. 3E). Despite their speed, these pipelines calculated pan-genomes as accurately as did the classical OrthoMCL pipeline (Fig. 2A, 3A), which was unable to handle more than 20 genomes in 600 min (Fig. 3E).

PEPPA accelerates its workflow with an initial pre-clustering step based on Linclust (Steinegger and Soding 2018) from the MMseqs package (Steinegger and Soding 2017). Linclust scales linearly with the number of genes, and was faster than CD-HIT, the clustering package used by Roary and PIRATE. The execution time of PEPPA also scaled linearly with the number of genomes (Fig. 3E). However, PEPPA was approximately twice as slow as Roary or PIRATE, possibly because it implements tree-based splitting of paralogs and similarity-based internal gene prediction. As described below, these two steps help PEPPA to provide more accurate pan-genomes with real *Salmonella* genomes (Fig. 2A) and with simulated data (Fig. 3A) than the four other pipelines.

Paralog splitting. The tree-based splitting of paralogs was incorporated into PEPPA in order to reduce the number of mistakes. Instead of relying on phylogenetic trees, Roary

and PIRATE use MCL (Enright *et al.* 2002) to split paralogous clusters. MCL uses a graph-based clustering approach to identify a strict optimal threshold that separates orthologous genes from paralogous genes. This approach can be very successful with closely related genomes, but it runs into problems when datasets contain both closely-related and distantly-related genomes because a single global optimal threshold does not exist for both extremes. PIRATE failed to split many paralogous clusters within real (Figs. 2D, E) and simulated (Figs. 3B, C) genomes, especially at higher levels of genetic diversity (Fig. 3A). Roary implements a synteny-based approach to identify and split unresolved paralog clusters. However, this approach also failed to correctly split orthologs into multiple clusters (Figs. 2D, E, 3B, C).

In contrast, PEPPA normalizes the pairwise genetic distances between the genes by comparing them to the genetic distances of the source genomes. This normalization helps to distinguish between the extreme genetic diversity introduced by HGT or gene duplications and the lower genetic diversity that accumulates during vertical descent. PEPPA then identifies an optimal threshold for each gene which is insensitive to the bursts of excessive genetic diversity, and uses that threshold to split paralogous branches in the gene trees. PanX also implements a tree-based algorithm for splitting paralogous clusters. However, that algorithm is not as sophisticated as the algorithm implemented in PEPPA, and seems to result in many false merges of paralogous genes (Figs. 2D, E, 3B, C).

Effects of internal annotations by PEPPA. Our analyses of 15 curated *Salmonella* genomes revealed an unanticipated strong impact of inconsistent annotations on the pan-genome predictions (Fig. 2A). Indeed, differences in annotation influenced the quality of the pan-genome more than pipeline algorithms (Fig. 2B), and can decrease the number of core genes by up to one-third (Fig. 2A). PEPPA avoids this problem because it implements a similarity-based gene-prediction step. Accordingly, the pan-genome predicted by PEPPA varied only slightly with different annotations (Fig. 2A, B). Similar findings with simulated genomes indicated that PEPPA is quite tolerant of missing annotations whereas the other pipelines are sensitive to them. Unlike the other pipelines, PEPPA automatically predicts pseudogenes based on the comparisons of

orthologs with intact coding sequences. As a result, PEPPA predicted numbers of pseudogenes in the 15 *Salmonella* genomes that were comparable to those in manual re-annotations. PEPPA thus supports an analysis of the accumulation of pseudogenes with time by individual bacterial lineages, and might be useful for studies concerning the importance of pseudogenes for potential adaptations to new environments. Draft genome assemblies based on 454 or IonTorrent sequencing include elevated numbers of single-base insertions and deletions due to inaccurate sequencing (Shao *et al.* 2013; Zhang *et al.* 2015). Including such genomes in an analysis reduces the quality of the pan-genome for all state of the art pipelines. However, PEPPA simply scores genes disrupted by artificial indels as frameshifts, making such inaccurate genomes easier to identify. For example, a public genome of *S. infantis* SK1302 was sequenced by 454 sequencing. An earlier pan-genome phylogeny of *Streptococcus* (Gao *et al.* 2014) separated this genomes from other *S. infantis* genomes, supposedly due to inconsistent gene predictions. However, PEPPA identified 523 pseudogenes genes in *S. infantis* SK1302, 326 of which were due to the frameshifts expected for 454 sequencing, and clustered SK1302 together with other *S. infantis* genomes (Fig. 5).

Alikhan *et al.* (2018) describes a pan-genome for all of *Salmonella* based on 537 genomes that had been derived by a precursor of PEPPA in 2015. That pan-genome had been used to develop a wgMLST scheme of 21,065 loci and a cgMLST scheme of 3002 genes. The same publication also described a reference set of 926 genomes that represented the diversity of almost 120,000 *Salmonella* genomes on the basis of rMLST. After completion of this manuscript, we became aware of a new publication (Park and Andam 2020) which used Roary to calculate a pan-genome of 84,041 *S. enterica* genes and only 2085 soft-core genes from those 926 representative genomes after re-annotation with Prokka. Such applications of Roary are discouraged by its documentation, which recommends against using it on diverse groups of organisms such as all *Salmonella.* The analyses presented here also indicated that the number of calculated pan-genes would be much too high and the number of soft core genes too low. We ran PEPPA on the same 926 representative genomes. A comparison with the Park and Andam results (Supplemental Table S7) fully confirmed our expectations, with 30,000 fewer pan-genes and 1200 more soft-core genes. The smaller pan-genome and

larger core-genome calculated by PEPPA have contributed greatly to the high resolution and reliability that EnteroBase has achieved.

Pan-genomes depend on sample size. Early analyses of pan-genomes were based on small numbers of genomic sequences (Tettelin *et al.* 2005), resulting in the conclusion based on 12 genomes that the pan-genome of *S. pyogenes* was closed (Tettelin *et al.* 2008). The same publication concluded that the pan-genome of *S. pneumoniae* was open, and would continue to expand indefinitely. However, a subsequent study of 44 genomes concluded that the pan-genome of *S. pneumoniae* was also closed (Donati *et al.* 2010). It is only very recently that large numbers of bacterial genomes are available for analysis, and that pipelines exist that can handle such large numbers. We therefore decided to examine the properties of the pan-genome from a large set of *Streptococcus* genomes because *Streptococcus* is well known for its diversity, and its taxonomy continues to be hotly debated.

We calculated a pan-genome from 3170 *Streptococcus* genomes that represent the genomic diversity of 14,115 draft and complete genomes. That pan-genome contains 41,026 genes, and is open and will continue to expand at a rate of 1.7 genes per novel, representative genome. However, the calculated rate of growth was over 25fold faster when a pan-genome of half that size was calculated from 138 genomes (Gao *et al.* 2014). We also calculated the pan-genomes and their expected rates of growth for the 29 most common ANI95% clusters within *Streptococcus* (Supplementary Material). All were open, including MG_1 (441 genomes of *S. pyogenes*) with one single exception, MG_41 (*S. sobrinus*). These inconsistencies with prior analyses suggested that the pan-genome status may be strongly dependent on the number of genomes, their sampling strategies, and possibly on the pipelines used to calculate pan-genomes.

Taxonomic insights. It has been clear since 2004 that the strict core genome of all prokaryotes is extremely small. Only 14-30 genes were present in all of 147 diverse genomes (Charlebois and Doolittle 2004), and almost all of those genes encoded ribosomal proteins (Weiss *et al.* 2018). However, it was still unexpected that the same would be true for a large collection of *Streptococcus* genomes. We only found 16 strict core genes in the representative set of 3170 genomes, and each of these was absent or

incomplete in one or more of the entire set of 14,115 genomes. Thus, when dealing with large datasets, a strict core genome can be a mirage at the genus level. Our examination of the traditional taxonomic groups within *Streptococcus* revealed an average of 746 strict core genes (range 241-1152), but most of these were present in multiple copies in some genomes, and therefore would not be suitable for classical phylogenetic analyses.

EnteroBase has successfully implemented core genome MLST for datasets of 100,000s of genomes within the genera *Salmonella* and *Escherichia* (Zhou *et al.* 2020). However, this is not strict core MLST but rather depends on a relaxed core, consisting of the presence of each gene in ≥95% of the genomes in a representative set of genomes. For *Streptococcus*, the relaxed core consists of 284 single-copy genes, and these would be suitable to establish an EnteroBase database. Here, we compared the topologies of a cgMLST tree of the 3170 representative *Streptococcus* genomes based on allelic identity of these relaxed core genes with a second tree based on the presence/absence of all 41,026 genes in the pan-genome. The two trees were largely congruent in the clustering of genomes from ANI95% clusters. In other words, ANI95% clusters were uniform according to either the pan-genome or a relaxed cgMLST scheme. The two trees were also largely congruent in the clustering of deep branches that correspond to traditional, broad streptococcal groupings such as Bovis or Mitis or Pyogenic, and even defined two new groupings at this level. The ANI95% clusters also corresponded well with some taxonomic designations of species and sub-species. However, in agreement with prior insights (Kilian *et al.* 2008; Jensen *et al.* 2016), other taxonomic groups such as *S. mitis* or *S. oralis* each encompass many ANI95% clusters, and do not correspond to a typical species. They do have topological cohesion, but at a much higher level of diversity than many other species, and even the so-called subspecies in these organisms are more diverse than a normal species.

As previously noted by others (Kilian *et al.* 2008; Jensen and Kilian 2012; Jensen *et al.* 2013; Jensen *et al.* 2016; Kilian and Tettelin 2019), the taxonomies of numerous *Streptococcus* genomes are misclassified in RefSeq (Supplemental Figure S3B). Misclassification has been ongoing for decades (Kikuchi *et al.* 1995) due to the

phenotypic heterogeneity of this species. The Mitis group is particularly heterogeneous (Kilian *et al.* 2008; Jensen *et al.* 2016) and difficult to study (Kilian and Tettelin 2019; Velsko *et al.* 2019). Similar problems also apply to other bacterial taxa such as *Pseudomonas* (Gomila *et al.* 2015) and *Aeromonas* (Beaz-Hidalgo *et al.* 2015). However, 227 ANI95% clusters which are defined here, and our confirmation of their existence here by independent phylogenetic approaches based on both cgMLST and gene presence, may provide a framework that can help to avoid such problems in the future, Furthermore, the test case of *Streptococcus* illustrates the power of PEPPA, which can now be used for defining the pan-genomes of other diverse genera.

Methods

***S. enterica* genomes.** We downloaded the assembly_summary_genbank.txt table and the assembly_summary_refseq.txt table from NCBI on 30th May of 2019 (ftp://ftp.ncbi.nlm.nih.gov/genomes/ASSEMBLY_REPORTS/). The first table summarizes all genomes uploaded into GenBank by their original authors and the second summarizes all the genomes in RefSeq. We used these tables as a source of the FTP links for each of the accession codes listed by Nuccio and Baümler (Nuccio and Baümler 2014) for genomic sequences of 15 *S. enterica* genomes. These 15 genomes were also annotated *ab initio* with PROKKA 1.12 (Seemann 2014). Nuccio and Baümler excluded some "unreliable" short genes from their manual re-curation. In order to exclude these genes in our analyses as well, the genomic coordinates of each gene in each of the three annotations (Submitter, RefSeq, PROKKA) were compared with the coordinates of "reliable genes" in Table S1 of Nuccio and Baümler. Only genes whose co-ordinates overlapped with those of a reliable gene by ≥90% were used here for further comparisons.

Preparation of simulated datasets.

All simulated datasets were generated using SimPan (Appendix 2) with the input parameters "--genomeNum 15 --aveSize 4500 --pBackbone 4000 --nMobile 10000 --nCore 3621 --pBackbone 0.05 --pMobile 0.40 --rec 0.1". Datasets a through e were generated with the additional parameters: (a) '--idenOrtholog 0.9 --idenParalog 0.6'; (b) '--idenOrtholog 0.95 --idenParalog 0.6'; (c) '--idenOrtholog 0.98 --idenParalog 0.6'; (d) '--idenOrtholog 0.98 --idenParalog 0.8'; (e) '--idenOrtholog 0.98 --idenParalog 0.9'. Ten other sets of simulated genomes that were used to evaluate running times were generated with the same parameters as dataset c but with the additional parameter '--genomeNum xxx', where xxx ranged from 20 to 200 by steps of 20.

Pan-genome pipelines.

The following versions of the individual pipelines and command lines were used for all benchmark datasets.

(1) PEPPA with a Git HEAD of f721513 was run in Python 3.6 environment as:

```
python PEPPA.py -t 4 -p PEPPA --pseudogene 0.9 --min_cds 45 *.gff
```

(2) Roary 3.6.0+dfsg-4 was installed as a Ubuntu APT package and run as:

```
roary -p 4 -o roary -f roary -i <identity> -s -v -y *.gff
```

Three runs of Roary were performed for each dataset with the additional parameters “-i 50”, “-i 80” or “-i 95”. The data reported here are from the runs with the parameter ‘-i 80’ because that consistently yielded the best ARI values.

(3) PIRATE with a Git HEAD of effc522 was downloaded from https://github.com/SionBayliss/PIRATE and run as:

```
PIRATE -i . -o PIRATE -s <steps> -t 4 -k "--diamond"
```

Three runs of PIRATE were performed for each dataset with the additional parameters “-s 50,60,70,80,90,95,98”, “-s 80,90,95,98” or “-s 95,98”. We report the data generated with “-s 80,90,95,98” which had the greatest ARI value, except for simulated dataset e, where “-s 95,98” had the greatest ARI.

(4) PanX v1.6.0 was downloaded from https://github.com/neherlab/pan-genome-analysis/releases and run in Python 2.7 environment as:

```
panX.py --folder_name panX --species_name panX --threads 4 --diamond_identity 80 --simple_tree --store_locus_tag
```

(5) OrthoMCL v2.0.9 was downloaded from https://orthomcl.org/ and run in multiple steps as described in https://currentprotocols.onlinelibrary.wiley.com/doi/full/10.1002/0471250953.bi0612s35.

Generating ANI95% clusters of *Streptococcus* genomes.

A summary table of all genomes deposited in RefSeq was downloaded on 20 June of 2019 (see *S. enterica* genomes above). 14,115 bacterial records that contained “Streptococcus” in the “organism_name” field were extracted from the table, and the

files for each record were downloaded as described above. MASH was used to measure the pairwise distances between the genomes with parameters of '-k 19 –s 10000'. The resulting matrix was used to cluster *Streptococcus* genomes with the AgglomerativeClustering function in the scikit-learn package (Pedregosa *et al.* 2011), with parameters linkage=single and distance_threshold=0.002. The function generated 3170 clusters. The genome with the greatest N50 value within each cluster was chosen as its representative genome. Pair-wise ANI values were calculated from the 3170 representative genomes with FastANI v1.2 (Jain *et al.* 2018), and these genomes were grouped into ANI95% clusters using the AgglomerativeClustering function with linkage=single and distance_threshold=0.05.

Data Access

Source code for PEPPA is deposited online at https://github.com/zheminzhou/PEPPA. Source code for SimPan is deposited online at https://github.com/zheminzhou/SimPan. Trees presented in Figure 5 are available separately: (A) https://achtman-lab.github.io/GrapeTree/MSTree_holder.html?tree=https://raw.githubusercontent.com/zheminzhou/PEPPA_data/master/Strep.gene_content.json; (B) https://achtman-lab.github.io/GrapeTree/MSTree_holder.html?tree=https://raw.githubusercontent.com/zheminzhou/PEPPA_data/master/Strep.CGAV.json.

Acknowledgements

This project was supported by the Wellcome Trust (202792/Z/16/Z) and EnteroBase development was funded by the BBSRC (BB/L020319/1). We gratefully acknowledge helps on the interpretation of *Streptococcus* taxonomy by Mogens Kilian, Daniel Jaén Luchoro and Francisco Salva Serra, and critical comments on the text by Nina Luhmann and Jane Charlesworth.

Author Contributions.

Z.Z. developed the pipelines, analyzed data and prepared the figures. M.A. and Z.Z. interpreted the results and wrote the manuscript.

Reference List

Abby SS, Tannier E, Gouy M, Daubin V. 2012. Lateral gene transfer as a support for the tree of life. *Proc Natl Acad Sci USA* **109**: 4962-4967.

Achtman M and Zhou Z. 2019. Analysis of the human oral microbiome from modern and historical samples with SPARSE and EToKi. *BioRxiv* 842542.

Alikhan N-F, Zhou Z, Sergeant MJ, Achtman M. 2018. A genomic overview of the population structure of *Salmonella*. *PLoS Genet* **14**: e1007261.

Altschul SF, Gish W, Miller W, Myers EW, Lipman DJ. 1990. Basic local alignment search tool. *J Mol Biol* **215**: 403-410.

Baumdicker F, Hess WR, Pfaffelhuber P. 2012. The infinitely many genes model for the distributed genome of bacteria. *Genome Biol Evol* **4**: 443-456.

Bayliss SC, Thorpe HA, Coyle NM, Sheppard SK, Feil EJ. 2019. PIRATE: A fast and scalable pangenomics toolbox for clustering diverged orthologues in bacteria. *Gigascience* **8**.

Beaz-Hidalgo R, Hossain MJ, Liles MR, Figueras MJ. 2015. Strategies to avoid wrongly labelled genomes using as example the detected wrong taxonomic affiliation for *Aeromonas* genomes in the GenBank database. *PLoS ONE* **10**: e0115813.

Blattner FR, Plunkett G, III, Bloch CA, Perna NT, Burland V, Riley M, Collado-Vides J, Glasner JD, Rode CK, Mayhew GF, et al. 1997. The complete genome sequence of *Escherichia coli* K-12. *Science* **277**: 1453-1462.

Brown T, Didelot X, Wilson DJ, De MN. 2016. SimBac: simulation of whole bacterial genomes with homologous recombination. *Microb Genom* **2**.

Buchfink B, Xie C, Huson DH. 2015. Fast and sensitive protein alignment using DIAMOND. *Nat Methods* **12**: 59-60.

Charlebois RL and Doolittle WF. 2004. Computing prokaryotic gene ubiquity: rescuing the core from extinction. *Genome Res* **14**: 2469-2477.

Chen L, Song N, Liu B, Zhang N, Alikhan N-F, Zhou Z, Zhou Y, Zhou S, Zheng D, Chen M, et al. 2019. Genome-wide identification and characterization of a superfamily of bacterial extracellular Contractile Injection Systems. *Cell Reports* **29**: 511-521.

Collins RE and Higgs PG. 2012. Testing the infinitely many genes model for the evolution of the bacterial core genome and pangenome. *Mol Biol Evol* **29**: 3413-3425.

Croucher NJ, Coupland PG, Stevenson AE, Callendrello A, Bentley SD, Hanage WP. 2014. Diversification of bacterial genome content through distinct mechanisms over different timescales. *Nat Commun* **5**: 5471.

Dalquen DA, Anisimova M, Gonnet GH, Dessimoz C. 2012. ALF--a simulation framework for genome evolution. *Mol Biol Evol* **29**: 1115-1123.

Daubin V and Ochman H. 2004. Bacterial genomes as new gene homes: the genealogy of ORFans in *E. coli*. *Genome Res* **14**: 1036-1042.

Dekker JP and Lau AF. 2016. An update on the *Streptococcus bovis* group: classification, identification, and disease associations. *J Clin Microbiol* **54**: 1694-1699.

Ding W, Baumdicker F, Neher RA. 2018. panX: pan-genome analysis and exploration. *Nucleic Acids Res* **46**: e5.

Donati C, Hiller NL, Tettelin H, Muzzi A, Croucher NJ, Angiuoli SV, Oggioni M, Dunning Hotopp JC, Hu FZ, Riley DR, et al. 2010. Structure and dynamics of the pan-genome of *Streptococcus pneumoniae* and closely related species. *Genome Biol* **11**: R107.

Enright AJ, Van DS, Ouzounis CA. 2002. An efficient algorithm for large-scale detection of protein families. *Nucleic Acids Res* **30**: 1575-1584.

Ensembl Release 98. 2019. GFF3 File Format - Definition and supported options. https://www.ensembl.org/info/website/upload/gff3.html.

Ferres I, Fresia P, Iraola G. 2019. simurg: simulate bacterial pangenomes in R. *Bioinformatics*.

Fleischmann RD, Adams MD, White O, Clayton RA, Kirkness EF, Kerlavage AR, Bult CJ, Tomb JF, Dougherty BA, Merrick JM, et al. 1995. Whole-genome random sequencing and assembly of *Haemophilus influenzae* Rd. *Science* **269**: 496-512.

Fletcher W and Yang Z. 2009. INDELible: a flexible simulator of biological sequence evolution. *Mol Biol Evol* **26**: 1879-1888.

Frentrup M, Zhou Z, Steglich M, Meier-Kolthoff JP, Göker M, Riedel T, Bunk B, Spröer C, Overmann J, Blaschitz M, et al. 2019. Global genomic population structure of *Clostridioides difficile*. *BioRxiv* 727230.

Gao X-Y, Zhi XY, Li HW, Klenk HP, Li WJ. 2014. Comparative genomics of the bacterial genus *Streptococcus* illuminates evolutionary implications of species groups. *PLoS One* **9**: e101229.

Gomila M, Pena A, Mulet M, Lalucat J, Garcia-Valdes E. 2015. Phylogenomics and systematics in *Pseudomonas*. *Front Microbiol* **6**: 214.

Goodhead I and Darby AC. 2015. Taking the pseudo out of pseudogenes. *Curr Opin Microbiol* **23**: 102-109.

Haegeman B and Weitz JS. 2012. A neutral theory of genome evolution and the frequency distribution of genes. *BMC Genomics* **13**: 196.

Haft DH, DiCuccio M, Badretdin A, Brover V, Chetvernin V, O'Neill K, Li W, Chitsaz F, Derbyshire MK, Gonzales NR, et al. 2018. RefSeq: an update on prokaryotic genome annotation and curation. *Nucleic Acids Res* **46**: D851-D860.

Hogg JS, Hu FZ, Janto B, Boissy R, Hayes J, Keefe R, Post JC, Ehrlich GD. 2007. Characterization and modeling of the *Haemophilus influenzae* core and supragenomes based on the complete genomic sequences of Rd and 12 clinical nontypeable strains. *Genome Biol* **8**: R103.

Holt KE, Parkhill J, Mazzoni CJ, Roumagnac P, Weill F-X, Goodhead I, Rance R, Baker S, Maskell D, Wain J, et al. 2008. High-throughput sequencing provides insights into genome variation and evolution in *Salmonella* Typhi. *Nature Genet* **40**: 987-993.

Hyatt D, Chen GL, Locascio PF, Land ML, Larimer FW, Hauser LJ. 2010. Prodigal: prokaryotic gene recognition and translation initiation site identification. *BMC Bioinformatics* **11**: 119.

Jain C, Rodriguez R, Phillippy AM, Konstantinidis KT, Aluru S. 2018. High throughput ANI analysis of 90K prokaryotic genomes reveals clear species boundaries. *Nat Commun* **9**: 5114.

Jensen A, Hoshino T, Kilian M. 2013. Taxonomy of the Anginosus group of the genus *Streptococcus* and description of *Streptococcus anginosus* subsp. *whileyi* subsp. nov. and *Streptococcus constellatus* subsp. *viborgensis* subsp. nov. *Int J Syst Evol Microbiol* **63**: 2506-2519.

Jensen A and Kilian M. 2012. Delineation of *Streptococcus dysgalactiae*, its subspecies, and its clinical and phylogenetic relationship to *Streptococcus pyogenes*. *J Clin Microbiol* **50**: 113-126.

Jensen A, Scholz CF, Kilian M. 2016. Re-evaluation of the taxonomy of the Mitis group of the genus *Streptococcus* based on whole genome phylogenetic analyses, and proposed reclassification of *Streptococcus dentisani* as *Streptococcus oralis* subsp. *dentisani* comb. nov., *Streptococcus tigurinus* as *Streptococcus oralis* subsp. *tigurinus* comb. nov., and *Streptococcus oligofermentans* as a later synonym of *Streptococcus cristatus*. *Int J Syst Evol Microbiol* **66**: 4803-4820.

Kikuchi K, Enari T, Totsuka K, Shimizu K. 1995. Comparison of phenotypic characteristics, DNA-DNA hybridization results, and results with a commercial rapid biochemical and enzymatic reaction system for identification of viridans group streptococci. *J Clin Microbiol* **33**: 1215-1222.

Kilian M, Poulsen K, Blomqvist T, Havarstein LS, Bek-Thomsen M, Tettelin H, Sorensen UB. 2008. Evolution of *Streptococcus pneumoniae* and its close commensal relatives. *PLoS ONE* **3**: e2683.

Kilian M and Tettelin H. 2019. Identification of virulence-associated properties by comparative genome analysis of *Streptococcus pneumoniae*, *S. pseudopneumoniae*, *S. mitis*, three *S. oralis* subspecies, and *S. infantis*. *MBio* **10**.

Konstantinidis KT, Rossello-Mora R, Amann R. 2017. Uncultivated microbes in need of their own taxonomy. *ISME J* **11**: 2399-2406.

Koonin EV. 2016. Splendor and misery of adaptation, or the importance of neutral null for understanding evolution. *BMC Biol* **14**: 114.

Kröger C, Dillon SC, Cameron AD, Papenfort K, Sivasankaran SK, Hokamp K, Chao Y, Sittka A, Hébrard M, Händler K, et al. 2012. The transcriptional landscape and small RNAs of *Salmonella enterica* serovar Typhimurium. *Proc Natl Acad Sci USA* **109**: E1277-E1286.

Laing C, Buchanan C, Taboada EN, Zhang Y, Kropinski A, Villegas A, Thomas JE, Gannon VP. 2010. Pan-genome sequence analysis using Panseq: an online tool for the rapid analysis of core and accessory genomic regions. *BMC Bioinformatics* **11**: 461.

Lerat E and Ochman H. 2005. Recognizing the pseudogenes in bacterial genomes. *Nucleic Acids Res* **33**: 3125-3132.

Li L, Stoeckert CJ, Jr., Roos DS. 2003. OrthoMCL: identification of ortholog groups for eukaryotic genomes. *Genome Res* **13**: 2178-2189.

Lobkovsky AE, Wolf YI, Koonin EV. 2013. Gene frequency distributions reject a neutral model of genome evolution. *Genome Biol Evol* **5**: 233-242.

Mellmann A, Harmsen D, Cummings CA, Zentz EB, Leopold SR, Rico A, Prior K, Szczepanowski R, Ji Y, Zhang W, et al. 2011. Prospective genomic characterization of the German enterohemorrhagic *Escherichia coli* O104:H4 outbreak by rapid next generation sequencing technology. *PLoS ONE* **6**: e22751.

Moura A, Criscuolo A, Pouseele H, Maury MM, Leclercq A, Tarr C, Bjorkman JT, Dallman T, Reimer A, Enouf V, et al. 2016. Whole genome-based population biology and epidemiological surveillance of *Listeria monocytogenes*. *Nat Microbiol* **2**: 16185.

Nuccio SP and Baümler AJ. 2014. Comparative analysis of *Salmonella* genomes identifies a metabolic network for escalating growth in the inflamed gut. *MBio* **5**: e00929-14.

Ondov BD, Treangen TJ, Melsted P, Mallonee AB, Bergman NH, Koren S, Phillippy AM. 2016. Mash: fast genome and metagenome distance estimation using MinHash. *Genome Biol* **17**: 132.

Page AJ, Cummins CA, Hunt M, Wong VK, Reuter S, Holden MT, Fookes M, Falush D, Keane JA, Parkhill J. 2015. Roary: rapid large-scale prokaryote pan genome analysis. *Bioinformatics* **31**: 3691-3693.

Park CJ and Andam CP. 2020. Distinct but intertwined evolutionary histories of multiple *Salmonella enterica* subspecies. *mSystems* **5**.

Parkhill J, Dougan G, James KD, Thomson NR, Pickard D, Wain J, Churcher C, Mungall KL, Bentley SD, Holden MT, et al. 2001. Complete genome sequence of a multiple drug resistant *Salmonella enterica* serovar Typhi CT18. *Nature* **413**: 848-852.

Pedregosa F, Vatoquaux G, Gramfort A, Michel V, Thirion B, Grisel O, Blondel M, Prettenhofer P, Weiss R, Dubourg V, et al. 2011. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research* **12**: 2825-2830.

Price MN, Dehal PS, Arkin AP. 2010. FastTree 2--approximately maximum-likelihood trees for large alignments. *PLoS One* **5**: e9490.

Rand WM. 1971. Objective criteria for the evaluation of clustering methods. *J Amer Statist Assoc* **66**: 846-850.

Sahl JW, Caporaso JG, Rasko DA, Keim P. 2014. The large-scale blast score ratio (LS-BSR) pipeline: a method to rapidly compare genetic content between bacterial genomes. *PeerJ* **2**: e332.

Sanaa M, Pouillot R, Vega FG, Strain E, Van Doren JM. 2019. GenomeGraphR: A user-friendly open-source web application for foodborne pathogen whole genome sequencing data integration, analysis, and visualization. *PLoS ONE* **14**: e0213039.

Sebaihia M, Wren BW, Mullany P, Fairweather NF, Minton N, Stabler R, Thomson NR, Roberts AP, Cerdeno-Tarraga AM, Wang H, et al. 2006. The multidrug-resistant human pathogen *Clostridium difficile* has a highly mobile, mosaic genome. *Nature Genet* **38**: 779-786.

Seemann T. 2014. Prokka: rapid prokaryotic genome annotation. *Bioinformatics* **30**: 2068-2069.

Shao W, Boltz VF, Spindler JE, Kearney MF, Maldarelli F, Mellors JW, Stewart C, Volfovsky N, Levitsky A, Stephens RM, et al. 2013. Analysis of 454 sequencing error rate, error sources, and artifact recombination for detection of Low-frequency drug resistance mutations in HIV-1 DNA. *Retrovirology* **10**: 18.

Simonsen M, Mailund T, Pedersen CNS. Inference of large phylogenies using Neighbour-Joining. 2011. Biomedical Engineering Systems and Technologies: 3rd International Joint Conference, BIOSTEC 2010. *Communications in Computer and Information Science* , 334-344. Springer Verlag.

Steinegger M and Soding J. 2017. MMseqs2 enables sensitive protein sequence searching for the analysis of massive data sets. *Nat Biotechnol* **35**: 1026-1028.

Steinegger M and Soding J. 2018. Clustering huge protein sequence sets in linear time. *Nat Commun* **9**: 2542.

Szöllösi GJ, Boussau B, Abby SS, Tannier E, Daubin V. 2012. Phylogenetic modeling of lateral gene transfer reconstructs the pattern and relative timing of speciations. *Proc Natl Acad Sci USA* **109**: 17513-17518.

Tatusova T, Ciufo S, Fedorov B, O'Neill K, Tolstoy I. 2014. RefSeq microbial genomes database: new representation and annotation strategy. *Nucleic Acids Res* **42**: D553-D559.

Tatusova T, DiCuccio M, Badretdin A, Chetvernin V, Nawrocki EP, Zaslavsky L, Lomsadze A, Pruitt KD, Borodovsky M, Ostell J. 2016. NCBI prokaryotic genome annotation pipeline. *Nucleic Acids Res* **44**: 6614-6624.

Tettelin H, Masignani V, Cieslewicz MJ, Donati C, Medini D, Ward NL, Angiuoli SV, Crabtree J, Jones AL, Durkin AS, et al. 2005. Genome analysis of multiple pathogenic isolates of *Streptococcus agalactiae*: Implications for the microbial "pan-genome". *Proc Natl Acad Sci USA* **102**: 13950-13955.

Tettelin H, Riley D, Cattuto C, Medini D. 2008. Comparative genomics: the bacterial pan-genome. *Curr Opin Microbiol* **11**: 472-477.

Velsko IM, Chakraborty B, Nascimento MM, Burne RA, Richards VP. 2018. Species designations belie phenotypic and genotypic heterogeneity in oral streptococci. *mSystems* **3**.

Velsko IM, Perez MS, Richards VP. 2019. Resolving phylogenetic relationships for *Streptococcus mitis* and *Streptococcus oralis* through core- and pan-genome analyses. *Genome Biol Evol* **11**: 1077-1087.

Vernikos G, Medini D, Riley DR, Tettelin H. 2015. Ten years of pan-genome analyses. *Curr Opin Microbiol* **23**: 148-154.

Weiss MC, Preiner M, Xavier JC, Zimorski V, Martin WF. 2018. The last universal common ancestor between ancient Earth chemistry and the onset of genetics. *PLoS Genet* **14**: e1007518.

Wozniak M, Wong L, Tiuryn J. 2014. eCAMBer: efficient support for large-scale comparative analysis of multiple bacterial strains. *BMC Bioinformatics* **15**: 65.

Zhang B, Penton CR, Xue C, Wang Q, Zheng T, Tiedje JM. 2015. Evaluation of the Ion Torrent personal genome machine for gene-targeted studies using amplicons of the nitrogenase gene *nifH*. *Appl Environ Microbiol* **81**: 4536-4545.

Zhao Y, Wu J, Yang J, Sun S, Xiao J, Yu J. 2012. PGAP: pan-genomes analysis pipeline. *Bioinformatics* **28**: 416-418.

Zhou Z, Alikhan N-F, Mohamed K, Fan Y, Agama Study Group, Achtman M. 2020. The EnteroBase user's guide, with case studies on *Salmonella* transmissions, *Yersinia pestis* phylogeny, and *Escherichia* core genomic diversity. *Genome Res* **30**: 138-152.

Zhou Z, Alikhan N-F, Sergeant MJ, Luhmann N, Vaz C, Francisco AP, Carrico JA, Achtman M. 2018a. GrapeTree: Visualization of core genomic relationships among 100,000 bacterial pathogens. *Genome Res* **28**: 1395-1404.

Zhou Z, Lundstrøm I, Tran-Dien A, Duchêne S, Alikhan N-F, Sergeant MJ, Langridge G, Fokatis AK, Nair S, Stenøien HK, et al. 2018b. Pan-genome analysis of ancient and modern *Salmonella enterica* demonstrates genomic stability of the invasive Para C Lineage for millennia. *Curr Biol* **28**: 2420-2428.

Zhou Z, McCann A, Weill F-X, Blin C, Nair S, Wain J, Dougan G, Achtman M. 2014. Transient Darwinian selection in *Salmonella enterica* serovar Paratyphi A during 450 years of global spread of enteric fever. *Proc Natl Acad Sci U S A* **111**: 12199-12204.

Appendix 1. PEPPA: Phylogeny Enhanced Prediction of a PAn-genome

The detailed workflow of the algorithm implemented in PEPPA consists of the following steps (Supplemental Fig. S1):

1.1 Iterative clustering of genes

Annotated genes are extracted from a set of genomes in the GFF3 format. Their nucleotide sequences are iteratively clustered using the Linclust module (Steinegger and Soding 2018) in MMseqs (Steinegger and Soding 2017) at the nucleotide sequence identity levels of 100% to 90% in steps of -1%. In each iteration, all except the longest sequence is discarded from each cluster, and the resulting set of those longest sequences is subjected to the next level of clustering. This procedure is equivalent to a hierarchical clustering of genes down to 90% identity, and results in a set of one representative gene for each 90% sequence homology cluster which is used for downstream analyses. The internal clustering results are used for the calculation of average genetic distances of genomes (section 1.5).

1.2 Build a gene graph from an all-against-all comparison of representative genes

Two all-against-all comparisons of all representative genes are computed using BLASTn (Altschul *et al.* 1990) and DIAMOND (Buchfink *et al.* 2015) and concatenated together. This yields an undirected graph with all representative genes as vertices, and edges between pairs of vertices between which at least 40% of the codons were aligned with at least 40% identity. This graph is used in sections 1.5 and 1.7.

1.3 Identifying representative genes in all genomes.

In order to cover genes that were missing in individual genome annotations, each representative gene sequence is aligned to all genome sequences using BLASTn and DIAMOND. The DIAMOND results are back-translated into nucleotide alignments. The alignment scores for alignments from both tools were re-calculated with a scoring scheme of +3 and -1 for each matched and mismatched site and -6 and -1 for the gap opening and extension penalties. The alignment from both tools with the greatest score

is kept when both tools align a representative gene to the same region of a genome, and otherwise both alignments are kept. PEPPA considers all alignments that span at least 50% of the length of a representative gene with 50% of identity as putative gene candidates. Each putative gene candidate is scored on information content, using a measure *s*, which is calculated as:

$$s = (l * i) * \sqrt{l * (r_q * r_r)^{1/2}}$$

where l is the length of the open reading frame of the putative gene candidate, *i* is its fraction of identity to the representative gene and r_q is the length of alignment of the gene candidate relative to the representative gene. r_r is calculated by comparing the genomic coordinates of the gene candidate with the genomic coordinates of each gene in the original GFF3 file given by the user. r_r equals to the length of the overlapped genomic region relative to the size of the gene candidate if at least 0.1 of the gene candidate overlaps with a gene in the original GFF3 annotation, 0.1 if it does not. These scores are saved for later use during the filtering of conflicting gene candidates in section 1.7.

1.4 Gene candidate alignment and phylogeny estimation

In order to facilitate accurate multi-sequence alignments, pseudo-sequences that are easy to handle are generated for each gene candidate. These consist of a copy of the corresponding reference gene sequence except that nucleotides which were absent in the above alignment are replaced by '-'. This procedure results in identities between the pseudo-sequences at all positions which were included in the alignments. A multiple alignment of these pseudo-sequences is used to calculate pair-wise genetic distances of the gene candidates that are based on percentage coverage, which are then used to build a RapidNJ tree (Simonsen *et al.* 2011) of all gene candidates for each representative sequence (Supplemental Fig. S1F). These alignments are also used in section 1.6 for the calculation of paralogous scores.

1.5 Genetic distances between genomes

PEPPA identifies paralogous genes by comparing the genetic distances of the gene candidates with the genetic distances of the genomes containing them. To this end,

PEPPA uses representative genes connected by edges in section 1.2 to calculate a proxy to the genetic distances of the genomes. Each discrete subgraph contains a unique set of similar genes, and is used to create a gene set. In order to exclude paralogs, PEPPA removes all genes from the set where more than one originated from a common genome, leaving only genes from different genomes within each set. PEPPA then selects up to 1,000 discrete subgraphs in descending order of numbers of genes as a conserved gene set for the estimation of average genetic distances between the genomes.

Pair-wise genetic distances are calculated between genes from each discrete subgraph according to their original similarities: (1) for two genes from the same 90% identity cluster, the genetic distance equals one minus the greatest identity during iterative clustering (section 1.1) that still clustered them together; (2) for two genes from different 90% identity clusters, the genetic distance equals to the value of the most distant edge in the shortest path connecting these two clusters (section 1.4). These pairwise genetic distances of genes are summarized over all subgraphs, and fitted to a log-normal distribution for each pair of genomes, yielding an estimate of average genetic distance ($\overline{D}$) and standard deviation (σ). In order to handle potential paralogs that are co-present in a common genome, that genomic distance is arbitrarily set to $\overline{D}$=0.005 and σ=2.

1.6 Phylogeny-based paralog splitting.

Each branch in a gene tree separates the genes into two sub-clusters. PEPPA scores gene as orthologs on the basis of whether the branch between them is compatible with their descent from a common ancestor (Supplemental Fig. S1E-H). This decision is reached by comparing the branch lengths for genetic distances between two sub-clusters of genes with the average genetic distance between the genomes carrying those genes.

Consider the simple scenario in which each sub-cluster contains only one gene (gene candidates i and j). PEPPA calculates the paralogous score for these two gene candidates as:

$$p_{i,j} = \frac{d_{i,j}}{e^{\overline{D}_{i,j}+3*\sigma_{i,j}}}$$

where $d_{i,j}$ is the genetic distance between the two genes, and $\overline{D}_{i,j}$ and $\sigma_{i,j}$ are the parameters for the log-normal distribution of the genetic distance between the two genomes carrying these genes (section 1.5). $p_{i,j}$>1 indicates a pair of paralogs that most likely diverged prior to the common ancestor of their genomes because the cumulative sequence differences between the genes are greater than 3σ of the average genetic distance of their genomes.

This calculation can be extrapolated to two sub-clusters of genes (*M* and *N*) defined by a branch with a weighted average of the paralogous scores for all genes in the two groups as:

$$\overline{p} = \frac{\sum_{i\epsilon M} \sum_{j\epsilon N} p_{i,j} * e^{-\overline{D}_{i,j}}}{\sum_{i\epsilon M} \sum_{j\epsilon N} e^{-\overline{D}_{i,j}}}$$

Here the *p* scores are weighted by the inverse of the genetic distances of the respective genomes, because paralogous genes are better indicated by comparing closely related genomes than more divergent ones. PEPPA splits each gene tree at the branch with the greatest $\overline{p}$ value if $\overline{p}$ >1 (Supplemental Fig. S1H). Branches in each of the sub-trees are iteratively tested and subjected to further splits until all branches have $\overline{p}$ ≤1. Each of the resulted sub-trees presents a monophyletic group.

1.7 Resolve conflicting gene candidates.

Gene candidates from distinct monophyletic groups can be assigned to overlapping regions in some genomes. In order to resolve such conflicts, an information score is assigned to each monophyletic group consisting of the sum of the information scores (section 1.2) for all gene candidates in the group. The monophyletic group with the greatest information score is kept for such a conflicted region, and all other gene candidates for that region are removed. As a special case, if the mapping of over 1/3 of the gene candidates in one monophyletic group overlaps with the gene candidates in a distinct monophyletic group with a higher information score, and the representative genes of these two monophyletic groups are connected by only one edge in the graph (section 1.4), the monophyletic group with the lower score is treated as a variant of the

other, and its gene candidates are merged into the monophyletic group with the higher score.

1.8 Use synteny to identify and remove recent duplicates

Paralogous genes with ≥99% identity that have recently arisen by gene duplications would not be detected by the previous phylogeny-based paralog analysis. However, when such duplicated genes candidates are located in distinct regions of the genome, they can be recognized by synteny with different neighboring genes. To this end, all gene candidates from each genome are ordered by their genomic coordinates, and up to three neighboring genes upstream and downstream of each gene candidate are extracted for synteny analyses. A minimum spanning tree is calculated for each monophyletic group based on the numbers of different neighbors between the individual gene candidates. The final set of orthologous genes consists of clusters of gene candidates within the minimum spanning tree with branch lengths of no more than 4 different neighboring genes.

1.9 Pseudogene identification.

Gene candidates were selected based on pseudo-genes created from a representative gene (section 1.2), and might include pseudogenes. PEPPA checks the integrity of the coding sequence in each gene candidate according to four criteria (Supplemental Fig. S1C): (1) the alignment covers ≥80% of the sequence of the representative gene; (2) ≥80% of the codons in the representative gene are not affected by a frameshift or nonsense mutation in the gene candidate; (3) a start codon is present within 60 bp upstream of the 5’ end of the alignment; (4) a stop codon is present within 600 bp downstream of the 3’ end. Candidate genes that pass all four criteria are considered to be coding genes and others are designated as pseudogenes.

Appendix 2. SimPan: A Pipeline for Simulating the evolution of Pan-genomes

The infinite many gene (IMG) model (Baumdicker *et al.* 2012) is arguably the simplest existing neutral mathematical model for the evolution of a pan-genome. In this model, accessory genes are inserted and deleted at fixed frequencies at nodes in a phylogenetic tree. The IMG model is the basis for several pan-genome simulators (Dalquen *et al.* 2012; Ferres *et al.* 2019) including the workflow in PanX (Ding *et al.* 2018). However, the IMG model does not match the U-shaped gene frequency distributions in multiple real bacterial pan-genomes (Collins and Higgs 2012; Lobkovsky *et al.* 2013). Several alternative models have also been proposed. They account for the U-shaped distribution as resulting from decreased gene insertion/deletion frequencies over long-term evolution, due to recent expansion of population size (Haegeman and Weitz 2012), long-term purifying selection (Lobkovsky *et al.* 2013; Zhou *et al.* 2014), or the presence of a special gene class that tends to be lost soon after it is acquired (Collins and Higgs 2012; Croucher *et al.* 2014). However, neither the IMG model itself nor these alternative models account for multiple, homoplastic insertions/deletions of a gene, nor large insertion/deletion events that span multiple consecutive genes.

SimPan was developed to simulate pan-genomes in a more realistic way by expanding the IMG model to account for these issues. (1) SimPan infers homoplastic gene insertion/deletion by using local trees generated by SimBac (Brown *et al.* 2016) that include homologous recombination events. (2) SimPan reshapes each local tree in a time-dependent manner, and reduces the long-term stability of gene insertions/deletions by penalizing the length of basal branches. (3) SimPan allows insertions/deletions of multiple consecutive genes.

2.1 Simulation of two gene pools

SimPan initiates a simulation by generating two gene pools according to input parameters: backbone genes (4,000) which are present in the ancestor of the population, and mobile genes (10,000) which are sources of horizontal genetic transfers. A random set of 5% percent of the backbone genes and 40% of the mobile genes are chosen to represent paralogous sources of recombination which resulted

from historical gene duplications, and are now only distantly related (60%-90% similarities). The phylogenies of these paralogous genes are also simulated using SimBac (Brown *et al.* 2016). Furthermore, SimPan assigns the sizes of the coding regions from a geometric distribution with an average value of 900 bp for each gene. SimPan also assigns the sizes for 5'-intergenic and 3'-intergenic regions surrounding its coding regions by drawing two additional numbers from a second geometric distribution with an average value of 50 bp for each gene. The total size of the sequences in the simulation equals the total sizes of backbone and mobile genes, multiplied by the total number of genomes.

2.2 Simulation of local trees and event trees

SimBac (Brown *et al.* 2016) simulates a clonal phylogeny of genomes, and then creates a local tree for every single site of the backbone and mobile genes that reflect the effects of random homogeneous recombination. SimPan converts each local tree into an event tree for later use to simulate gene insertion/deletions by resizing the branches in the trees according to their distances to the tips. Given a time-dependent factor λ=10, the new length (l') of each branch in the event tree is calculated as:

$$l' = \lambda * (e^{-\lambda * s} - e^{-\lambda * (s+l)})$$

Where l is the length of the branch in the local tree and s is its distance to the tips.

2.3 Simulation of insertion and deletion events

To simulate a deletion event, a point on the branches of a random event tree is chosen at random, and all descendant genomes after that point are subjected to deletions of a random block of backbone genes (default average length: 3 genes). The same process is repeated until the number of remaining core genes in any genome is equal or less than a user-defined number (default: 3,621). Similarly, for each insertion event, a set of genomes descended from a random point in an event tree are selected, and a block of consecutive mobile genes (default average length: 10) is inserted at a randomly chosen genomic location into these genomes. Insertions are also repeated iteratively until the average number of genes per genome is greater or equal than a user-defined number (default: 4,500).

2.4 Simulation of gene and genomic sequences

The coding and intergenic regions of the genes are simulated with the program INDELible (Fletcher and Yang 2009) from the original local trees generated by SimBac according to the M1 and HKY models, respectively. Additional short insertions and deletions (10 bp on average) are also simulated with INDELible at a rate of 100 mutations per event. INDELible does not simulate start and stop codons. Therefore, the first and last codons of each coding region are replaced by a random bacterial start and stop codon from sets of start codons (ATG: 83%; GTG: 14%; TTG: 3%) and stop codons (TAA: 63%; TGA: 29%; TGA: 8%) that represent their empirical distributions in *E. coli* K-12 (Blattner *et al.* 1997). All gene sequences are then concatenated into genomic sequences, and stored as files in FASTA format for the generation of annotated genomes in GFF (Ensembl Release 98 2019) format.

Appendix 3. Using PEPPA for whole-genome MLST schemes

PEPPA has been used to set up whole genome multi-locus sequence typing (wgMLST) schemes for multiple genera in EnteroBase. These wgMLST schemes can be used to identify the presence of interesting biologically meaningful genes in a bacterial population (Chen *et al.* 2019), and also for further selection of a subset of conserved loci for a core genome MLST (cgMLST) scheme (Alikhan *et al.* 2018; Zhou *et al.* 2020). Here we describe manual curations that were needed according to our experience to establish a reliable wgMLST scheme with PEPPA.

3.1 Selection of a reliable set of reference genomes

Unlike other pan-genome pipelines, PEPPA re-annotates all genomes to ensure the consistency of gene annotations. By default, the input genes are iteratively clustered and the longest sequence within each cluster is chosen as a cluster representative (section 1.1). The alignments of the representatives from all genomes determine the start and end coordinates of the final pan-genes. This approach works well on input genomes that have been automatically annotated without manual curations. However, genome sets that are used to build a wgMLST scheme will often contain reference genomes whose annotations have been curated by experts and are widely accepted by the community (Blattner *et al.* 1997; Sebaihia *et al.* 2006). In other cases, annotations have even been checked by experimental data such as transcriptomics (Kröger *et al.* 2012). The annotated genes in such genomes are likely to be more reliable than those predicted automatically by PEPPA, and their names are well known by the scientific community. PEPPA therefore allows users to incorporate such prior knowledge in the pan-genome, by defining a flexible priority order for the input genomes using the “-P” (Priority) parameter. When PEPPA clusters genes from genomes of different priority levels, genes from genomes with the lowest priority numbers will be preferably selected as representatives of the gene cluster. In the case of multiple genomes with identically low priorities, the longest gene will be selected. For example, we assigned each of the representative *Streptococcus* genomes into one of five priority levels before constructing the pan-genome (Supplemental Table S2), based on a combination of its

"RefSeq category" (Tatusova *et al.* 2014) and assembly status. The five priority levels were: 1: a reference genome consisting of manually selected 'gold standard' complete genomes with high-quality annotation within RefSeq; 2: a complete representative genome; 3: a draft representative genome; 4: other complete genomes; 5: other draft genomes. These priority levels resulted in the representative genes for 774 of the 777 relaxed core genes having originated from genomes in priority level 1 (Supplemental Table S4).

3.2 Identification of genes that are too similar

Each pan-gene identified by PEPPA is assigned to a group of orthologs that has been screened for paralogs according to phylogeny- and synteny-based approaches. However, this approach is not feasible for an MLST nomenclature server that assigns sequences to loci and alleles from newly assembled genomes. Instead, the EnteroBase nomenclature server calls MLST loci and alleles in a new genome by aligning a pre-defined set of representative sequences onto that genome (Zhou *et al.* 2020). Due to sequence diversity and horizontal gene transfer, novel genomes can include sequence variants that will align to multiple similar pan genes, resulting in overlapping gene calls for multiple representative gene sequences. Such related pan genes can be identified by the EnteroBase function MLSTdb in the EToKi package (https://github.com/zheminzhou/EToKi). EToKi MLSTdb applies an all-against-all comparison on all gene sequences generated by PEPPA, and removes pan genes that have ≥70% identity at either their nucleotide or amino acid level. All wgMLST schemes hosted by EnteroBase have been subjected to this final cleaning before being made publicly available.

Table 1. Summary statistics of the pan-genome of *Streptococcus* genus and seven species groups.

	Genomes	Number per genome			95% of genomes		All genomes		
Traditional Group		**Genes**	**CDSs**	**% pseudogenes**	**All genes**	**Single copy**	**Strict core genes**	**Strict core CDSs**	**Pan genes**
Streptococcus	3,170	1,823	1,970	7.5	777	284	16	0	41,026
Anginosus	103	1,861	1,733	6.9	1,235	661	1,152	773	6,850
Bovis	82	1,905	1,765	7.3	1,211	532	899	438	7,463
Mitis	1,514	2,038	1,863	8.6	1,105	637	241	7	16,948
Mutans	169	1,857	1,720	7.4	1,167	400	671	115	6,621
Pyogenic	815	1,834	1,736	5.4	999	426	489	94	13,075
Salivarius	100	1,913	1,738	9.2	1,297	602	1,026	553	5,975
Suis	347	2,117	1,969	7.0	1,299	665	743	432	11,395

Strict core refers to the number of genes that were found by DIAMOND and BLASTn in all genomes. Strict core CDSs refers to the number of genes which are not a pseudogene in any genome.

Figure Legends

Figure 1. A brief overview of the workflow for PEPPA. (A) Flow chart indicating the four cascading groups of operations from top to bottom. (B) Cartoon of similarity-based prediction of gene candidates (left) and phylogeny-based paralog splitting (middle and right). The tree was split at the red branch (right) to separate gene candidates into two sub-clusters. The gene pairs in the same sub-cluster had low paralogous scores (blue quadrilaterals and arrows at left) whereas gene pairs between the sub-clusters had high paralogous scores (yellow). (C) Flow chart of the pseudogene identification. The detailed workflow of the algorithm implemented in PEPPA can be found in Supplemental Fig. 1 and Appendix 1.

Figure 2. Comparison of pan-genome predictions for 15 *Salmonella* genomes with a manually curated pan-genome (Nuccio and Baümler 2014). (A) The adjusted Rand index *versus* the manual curation (ARI; histogram) and the sizes of core genomes (circles) in each of the pan-genomes after annotation by Prokka (Seemann 2014), after reannotation in RefSeq with PGAP (Tatusova *et al.* 2016), and as originally submitted to NCBI (Submitter). *indicates that PanX failed to run on the "Submitter" annotations. (B) A Neighbour-joining tree (left) of the pairwise ARI scores (heatmap at the right) between the predicted pan-genomes and the curated pan-genome. The annotation source is indicated within grey shadows at the left except for PEPPA where it is listed at the tips. Colors are as in part A. (C) Histogram of the numbers of pseudogenes (Y-axis) in each of the genomes (X-axis) in the curated pan-genome (grey) and pan-genome predicted by PEPPA (orange). A dashed line separates the two *Salmonella* pathovar groups described by Nuccio and Bäumler. (D, E) Histograms of the average numbers of false splits (top) and merges (bottom) of ortholog groups by the individual pipelines (X-axis) in backbone (D) or mobile (E) genes.

Figure 3. Comparison of the pan-genome pipelines with simulated data generated by SimPan. (A) The adjusted Rand index (ARI; histogram) and the sizes of core genomes (circles) in the pan-genomes produced by SimPan simulations a, b, c, d, e (inset). Left: pan-genome produced by the simulations. Other histograms, pan-genomes calculated by five pipelines. (B) Numbers of failed splits (top) and false merges (bottom) of ortholog groups by five pipelines with backbone genes. (C) Numbers of failed splits (top) and false merges (bottom) of ortholog groups by five pipelines with mobile genes. (D) The adjusted Rand index (ARI; histogram) and the sizes of core genomes (circles) in the pan-genomes produced by SimPan simulation c after random deletions of 0%, 0.1%, 1% and 2% of the gene annotations. Other details as in part A. (E) Runtime for each pipeline (Y-axis) versus number of genomes in simulated datasets (X-axis). Runs which exceed 600 minutes are not shown.

Figure 4. Rarefaction curves of pan- and core-gene numbers in *Streptococcus* and its seven major taxonomic subgroups. (A) Rarefaction curves created with PEPPA_parser for the accumulations of pan genes and core genes of 3,170 *Streptococcus* representative genomes from 300 random permutations. (B) The frequencies of pan genes (Y-axis) by the numbers of genomes that carried that many genes (X-axis). The inset shows the relaxed core genes present in ≥ 95% of the genomes. (C) Rarefaction curves of genomes in the Mitis, Pyogenic and Suis groups. (D) Rarefaction curves of genomes in the Mutans, Anginosus, Salivarius and Bovis groups. The dark lines in figures A, C and D indicate median values and the shadows indicate 95% confidence intervals.

Figure 5. Phylogenies of 3,170 *Streptococcus* genomes based on accessory gene content (A) and allelic variation in relaxed core genes (B). (A) A FastTree (Price *et al.* 2010) phylogeny based on binary information of the presence and absence of accessory genes in. (B) A RapidNJ (Simonsen *et al.* 2011) phylogeny based on numbers of identical sequences (alleles) of 284 single copy, relaxed, core genes that are present in ≥95% of *Streptococcus* genomes. These trees are represented in GrapeTree (Zhou *et al.* 2018a). The sizes of the circles in A and B are proportional to the numbers of genomes they encompass, and are color-coded by 29 common ANI95% clusters as shown in the inset. Many *Streptococcus* species have been assigned to one of six traditional taxonomic groups whose names are shown outside colored arcs. These trees define from the Suis group which contains *S. suis*. A black arrow in Figure B shows the root of the tree, where multiple branches radiate directly outwards due to lack of resolution of cgMLST for such distant taxa. All ANI95% cluster information can be found in Supplemental Table S3. Interactive versions of the trees can be found at (A) https://achtman-lab.github.io/GrapeTree/MSTree_holder.html?tree=https://raw.githubusercontent.com/zheminzhou/PEPPA_data/master/Strep.gene_content.json; (B) https://achtman-lab.github.io/GrapeTree/MSTree_holder.html?tree=https://raw.githubusercontent.com/zheminzhou/PEPPA_data/master/Strep.CGAV.json.

Supplemental Figure S1. The detailed workflow of PEPPA algorithm. (A-D) The four successive groups of operations as shown in Figure 1A. The operations of 1.1 to 1.9 correlate with sub-headings in Appendix 1. (E-H) Cartoon of an example for the phylogeny-based paralog splitting. (E) A homologous group of five gene candidates (circles) derived from alignments of a representative gene (R; red dot) on three genomes (1-3). (F) RapidNJ tree of the homologous group. (G) Compatible (black) and incompatible (red) pairs of gene candidates by their paralogous scores. (H) Split of gene tree on the branch with the greatest paralogous score.

Supplemental Figure S2. The detailed workflow of SimPan. The operations initiated with 2.x and in bold fonts are sub-titles that corresponds to sub-titles in Appendix 2. The words in blue are the names of the parameters to be specified by the users (Table S5).

Supplemental Figure S3. Mapping of Species label for each *Streptococcus genome* onto phylogenies generated by accessory gene contents (A) and by allelic variations in core genes (B). The trees in A and B are the same as in Figure 5, except that the nodes are color-coded by Species labels deposited in GenBank, as shown in the Key. Some species labels that are inconsistent with both phylogenies are highlighted with colored arrows.

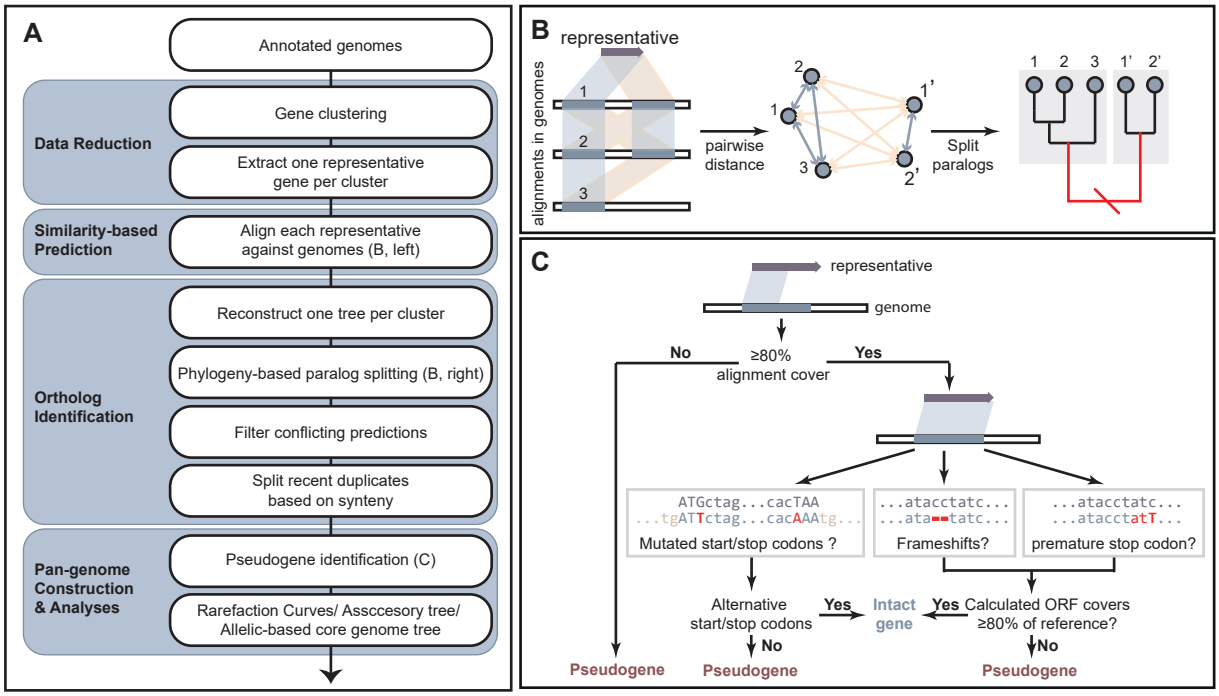

Figure 1. A brief overview of the workflow for PEPPA. (A) Flow chart indicating the four cascading groups of operations from top to bottom. (B) Cartoon of similarity-based prediction of gene candidates (left) and phylogeny-based paralog splitting (middle and right). The tree was split at the red branch (right) to separate gene candidates into two sub-clusters. The gene pairs in the same sub-cluster had low paralogous scores (blue quadrilaterals and arrows at left) whereas gene pairs between the sub-clusters had high paralogous scores (yellow). (C) Flow chart of the pseudogene identification. The detailed workflow of the algorithm implemented in PEPPA can be found in Supplemental Fig. 1 and Appendix 1.

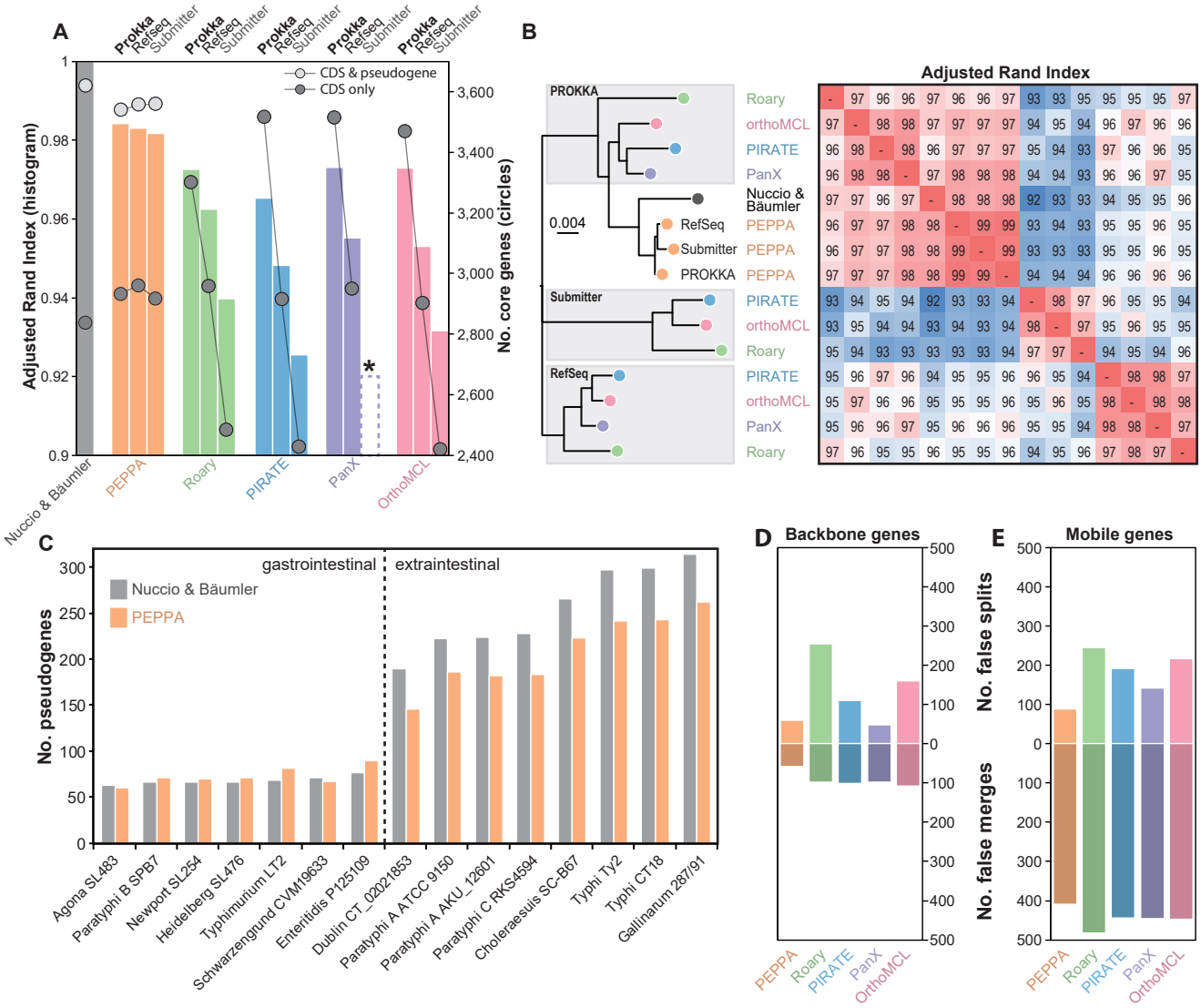

Figure 2. Comparison of pan-genome predictions for 15 *Salmonella* genomes with a manually curated pan-genome (Nuccio and Baümler 2014). (A) The adjusted Rand index *versus* the manual curation (ARI; histogram) and the sizes of core genomes (circles) in each of the pan-genomes after annotation by Prokka (Seemann 2014), after reannotation in RefSeq with PGAP (Tatusova et al. 2016), and as originally submitted to NCBI (Submitter). *indicates that PanX failed to run on the "Submitter" annotations. (B) A Neighbour-joining tree (left) of the pairwise ARI scores (heatmap at the right) between the predicted pan-genomes and the curated pan-genome. The annotation source is indicated within grey shadows at the left except for PEPPA where it is listed at the tips. Colors are as in part A. (C) Histogram of the numbers of pseudogenes (Y-axis) in each of the genomes (X-axis) in the curated pan-genome (grey) and pan-genome predicted by PEPPA (orange). A dashed line separates the two *Salmonella* pathovar groups described by Nuccio and Bäumler. (D, E) Histograms of the average numbers of false splits (top) and merges (bottom) of ortholog groups by the individual pipelines (X-axis) in backbone (D) or mobile (E) genes.

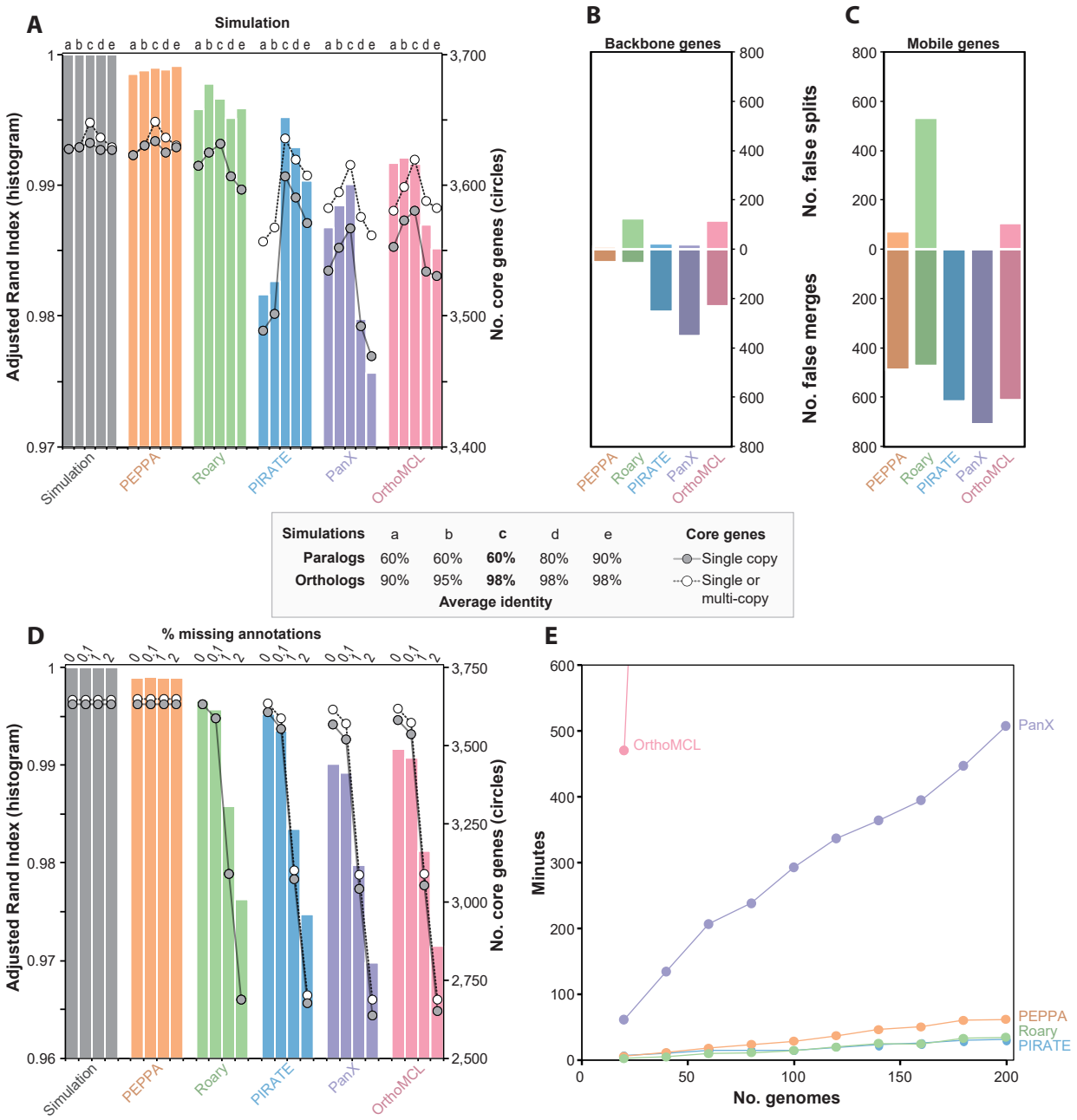

Figure 3. Comparison of the pan-genome pipelines with simulated data generated by SimPan. (A) The adjusted Rand index (ARI; histogram) and the sizes of core genomes (circles) in the pan-genomes produced by SimPan simulations a, b, c, d, e (inset). Left: pan-genome produced by the simulations. Other histograms, pan-genomes calculated by five pipelines. (B) Numbers of failed splits (top) and false merges (bottom) of ortholog groups by five pipelines with backbone genes. (C) Numbers of failed splits (top) and false merges (bottom) of ortholog groups by five pipelines with mobile genes. (D) The adjusted Rand index (ARI; histogram) and the sizes of core genomes (circles) in the pan-genomes produced by SimPan simulation c after random deletions of 0%, 0.1%, 1% and 2% of the gene annotations. Other details as in part A. (E) Runtime for each pipeline (Y-axis) versus number of genomes in simulated datasets (X-axis). Runs which exceed 600 minutes are not shown.

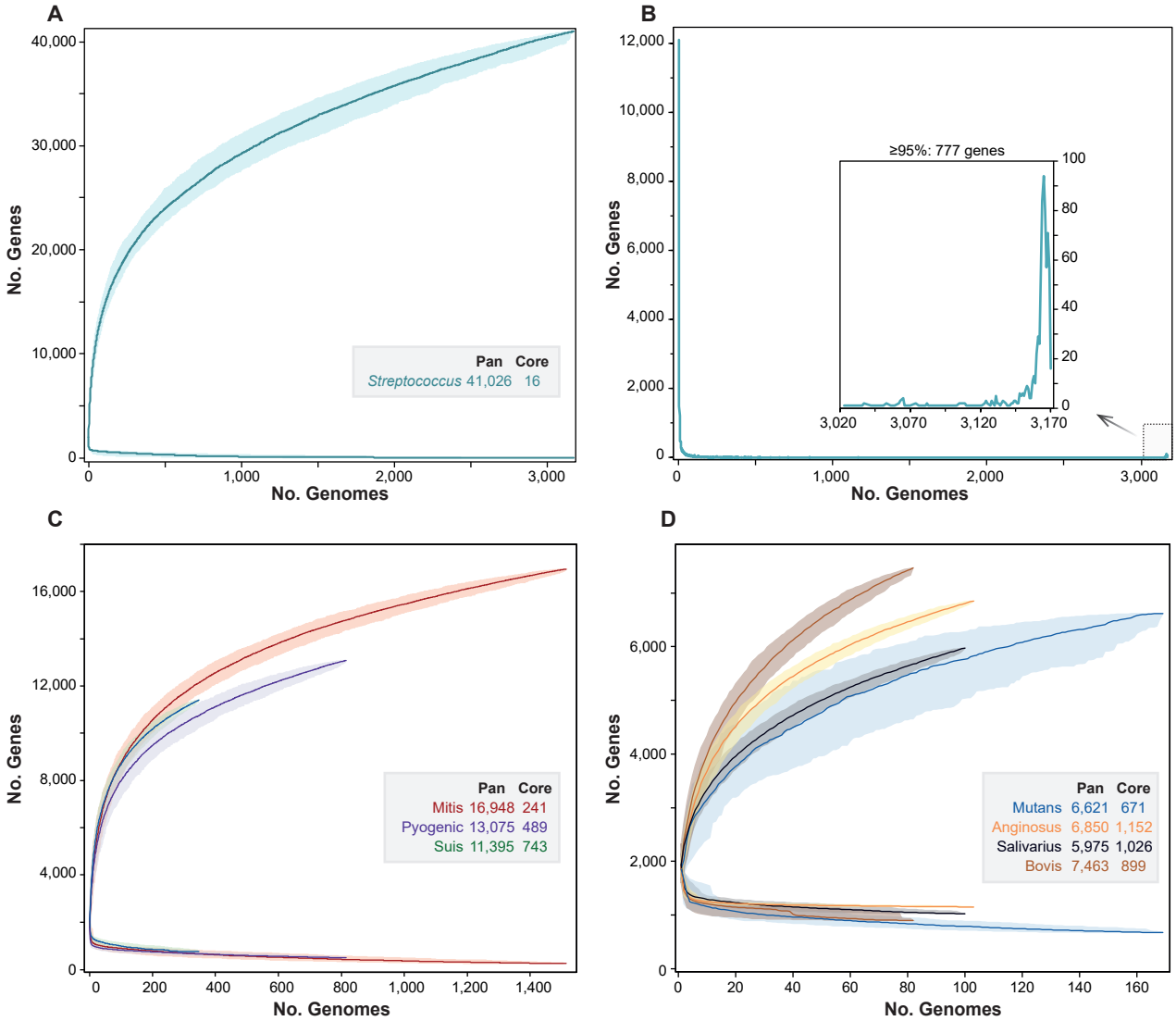

Figure 4. Rarefaction curves of pan- and core-gene numbers in *Streptococcus* and its seven major taxonomic subgroups. (A) Rarefaction curves created with PEPPA_parser for the accumulations of pan genes and core genes of 3,170 *Streptococcus* representative genomes from 300 random permutations. (B) The frequencies of pan genes (Y-axis) by the numbers of genomes that carried that many genes (X-axis). The inset shows the relaxed core genes present in ≥ 95% of the genomes. (C) Rarefaction curves of genomes in the Mitis, Pyogenic and Suis groups. (D) Rarefaction curves of genomes in the Mutans, Anginosus, Salivarius and Bovis groups. The dark lines in figures A, C and D indicate median values and the shadows indicate 95% confidence intervals.

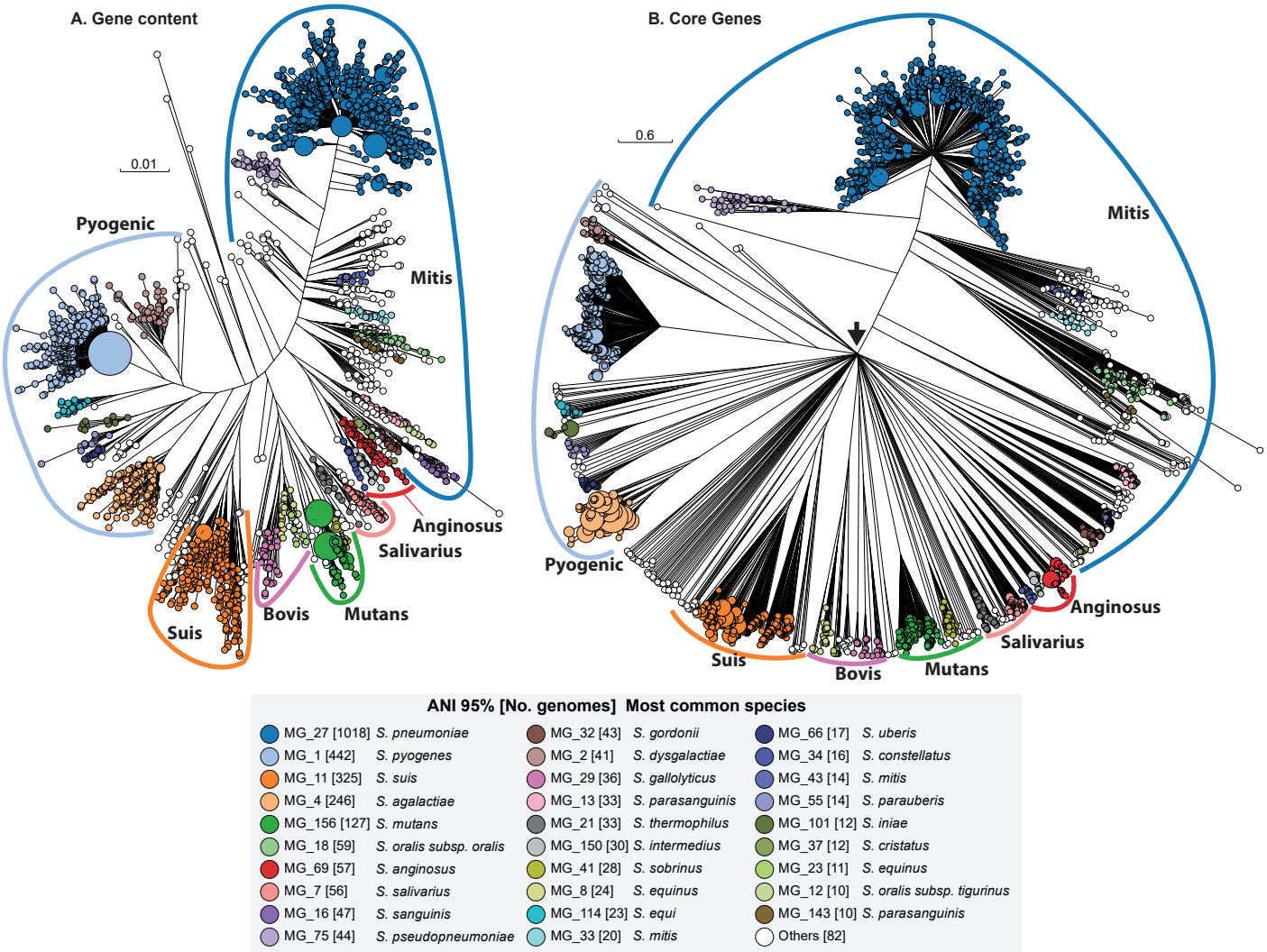

Figure 5. Phylogenies of 3,170 *Streptococcus* genomes based on accessory gene content (A) and allelic variation in relaxed core genes (B). (A) A FastTree (Price *et al*. 2010) phylogeny based on binary information of the presence and absence of accessory genes in. (B) A RapidNJ (Simonsen *et al*. 2011) phylogeny based on numbers of identical sequences (alleles) of 284 single copy, relaxed, core genes that are present in ≥95% of *Streptococcus* genomes. These trees are represented in GrapeTree (Zhou *et al.* 2018a). The sizes of the circles in A and B are proportional to the numbers of genomes they encompass, and are color-coded by 29 common ANI95% clusters as shown in the inset. Many *Streptococcus* species have been assigned to one of six traditional taxonomic groups whose names are shown outside colored arcs. These trees define from the Suis group which contains *S. suis*. A black arrow in Figure B shows the root of the tree, where multiple branches radiate directly outwards due to lack of resolution of cgMLST for such distant taxa. All ANI95% cluster information can be found in Supplemental Table S3. Interactive versions of the trees can be found at (A) https://achtman-lab.github.io/GrapeTree/MSTree_holder.html?tree=https://raw.githubusercontent.com/zheminzhou/PEPPA_data/master/Strep.gene_content.json; (B) https://achtman-lab.github.io/GrapeTree/MSTree_holder.html?tree=https://raw.githubusercontent.com/zheminzhou/PEPPA_data/master/Strep.CGAV.json.