Systems-based approach for optimization of a scalable bacterial ST mapping assembly-free algorithm

Natasha Pavlovikj[1¶], Joao Carlos Gomes-Neto[2,3¶], Jitender S. Deogun[1], Andrew K. Benson[2,3*]

[1] Department of Computer Science and Engineering, University of Nebraska-Lincoln, Lincoln, Nebraska, United States of America

[2] Department of Food Science and Technology, University of Nebraska-Lincoln, Lincoln, Nebraska, United States of America

[3] Nebraska Food for Health Center, University of Nebraska-Lincoln, Lincoln, Nebraska, United States of America

* Corresponding Author:

E-mail: abenson1@unl.edu (AKB)

¶ These authors contributed equally to this work.

Abstract

Epidemiological surveillance of bacterial pathogens requires real-time data analysis with a fast turn-around, while aiming at generating two main outcomes: 1) Species level identification; and 2) Variant mapping at different levels of genotypic resolution for population-based tracking, in addition to predicting traits such as antimicrobial resistance (AMR). With the recent advances and continual dissemination of whole-genome sequencing technologies, large-scale population-based genotyping of bacterial pathogens has become possible. Since bacterial populations often present a high degree of clonality in the genomic backbone (i.e., low genetic diversity), the choice of genotyping scheme can even facilitate the understanding of ancestral relationships and can be used for prediction of co-inherited traits such as AMR. Multi-locus sequence typing (MLST) fits that purpose and can identify sequence types (ST) based on seven ubiquitous genome-scattered loci that aid in genotyping isolates beneath the species level. ST-based mapping also standardizes genotyping across laboratories and is used by laboratories worldwide. However, algorithms for inferring ST from Illumina paired-end sequencing data typically rely on genome assembly prior to classification. Genome assembly is computationally intensive and is a bottleneck for speed and scalability, which are important aspects of genomic epidemiology. The stringMLST program uses an assembly-free, kmer-based algorithm for inferring STs, which can overcome the speed and scalability bottlenecks. Here we have systematically studied the accuracy and scalability of stringMLST relative to the standard MLST program across a wide array of phylogenetically divergent

Public Health-relevant bacterial pathogens. Our data shows that optimal kmer length for stringMLST is species-specific and that genome-intrinsic and -extrinsic features can affect performance and accuracy of the program. While suitable parameters could be identified for most organisms, there were a few instances where this program may not be directly deployable in its current format. More importantly, we integrated stringMLST into our freely available and scalable hierarchical-based population genomics platform, ProkEvo, and further demonstrated how the implementation facilitates automated, reproducible bacterial population analysis. The ProkEvo implementation provides a rapidly deployable genomic epidemiology tool for ST mapping along with other pan-genomic data mining strategies, while providing specific guidance on how to optimize stringMLST performance for a wide variety of bacterial pathogens.

Introduction

Modern epidemiological investigation of bacterial pathogens relies on rapid genomic characterization of new isolates routinely received by Public Health laboratories, along with bioinformatics programs for classification/comparison of genomic data from new isolates to existing data from many thousands of isolates [1][2]. While surveillance, source-tracking, and attribution are primary goals for use of whole-genome sequencing (WGS) data by Public Health agencies, there is growing interest in using the WGS data from thousands to hundreds of thousands of isolates of a given pathogen to study hierarchical genotypic classification at different levels of resolution, and prediction of population-specific traits

contributing to virulence, ecological fitness, or antimicrobial resistance (AMR) [3].

Evolutionarily-related bacterial populations (e.g., species and subtypes of a species) share common genotypic backbones and the degree of clonality of a population provides a context for studying co-inheritance of core- and accessory-genes [4][5][6][7]. Core loci are shared by at least 99% of the genomes, whereas accessory loci represent a sparse ensemble that are not shared by all variants of a population, but jointly, the core and accessory genomic content form a given species forms its pan-genome [66]. Population genetic analysis of bacterial species using multi-locus methods have shown that multiple types of genotyping schemes can provide sufficient resolution for classifying populations based on genotypes beneath the species level [8][22][79][80][81], and the genetic relationships between different populations reveal the pattern of population diversification and structuring. This optimal level of genotypic resolution can be considered an informative genotypic unit that facilitates both ecological and epidemiological inquiries [57][82][83].

Multi-locus sequence typing (MLST) is a well-established and widely used genotyping technique that classifies bacterial genomes into sequence types (ST) [8][81]. ST classification based on traditional MLST is usually inferred from seven loci that are found ubiquitously in the species and are scattered around the genome. Highly curated, species-specific databases of allelic variation at MLST loci and distributions of STs are publicly available [27][30][67]. Historically, sequences for MLST loci were generated from locus-specific polymerase chain

93 reaction (PCR) assays, but now are typically inferred from assembled WGS data
94 [8][9][44].
95 ST-based classification provides useful and relevant genotypic units for
96 epidemiological surveillance, population genetic analysis, and evolutionary
97 inference. Relative to surveillance, ST-based genotyping standardizes the
98 nomenclature for intra- and inter-laboratorial diagnostics and epidemiological
99 inquiries worldwide [9][10][11]. With respect to evolutionary inference, isolates
100 sharing alleles at five or more loci are more likely to be ancestrally related and are
101 commonly classified as members of a clonal complex (i.e., group of STs that have
102 shared a common ancestor very recently) [8][9][28][44]. The ST and clonal
103 complexes provide a point of reference for genetic analysis (e.g., Linkage
104 Disequilibrium - LD) to track inheritance/co-variance of accessory loci among
105 different STs or clonal complexes. For example, LD analyses of pan-genomic
106 content can inform phenotypic predictions for inheritance of unique traits in an ST
107 related to virulence, environmental stability, or AMR [12][13][14].
108 Recent studies have shown that LD between specific sets of accessory genes
109 such as AMR genes is an intrinsic property of some STs of specific bacterial
110 pathogens, and rapid identification of a given ST from WGS data can quickly
111 provide accurate prediction of AMR profiles [14][15]. There are important
112 applications of this innovative strategy for treatment of patients and there is
113 growing interest in defining LD relationships of AMR genes and virulence-
114 associated genes with STs. A limiting factor in ST-based classification from WGS
115 data, however, is the dependence of on genome assembly, most often from

116 Illumina-based short-read data [27][30][67]. This bottleneck can also hinder ST-
117 based surveillance efforts when many thousands of genomes are involved in a
118 real-time analysis [16][21]. One approach to overcome the computational
119 bottleneck is to use kmer-based ST classification directly from Illumina paired-
120 end raw reads (assembly-free). The stringMLST algorithm was recently
121 developed for this purpose and shown to provide computationally efficient,
122 assembly-free ST classifications [17]. However, stringMLST has not been
123 broadly evaluated across data from multiple Public Health-related bacterial
124 pathogenic species. Given the remarkable variation in the intrinsic features of
125 genomes from different species (e.g., compositional features, abundances of
126 repetitive sequences, mutation rates, and rates of recombination); it is likely that
127 parameters for assembly-free algorithms, such as stringMLST, will need to be
128 empirically optimized for genetically diverse species and even subtypes (e.g.,
129 serotypes) of pathogens.

130 To test this hypothesis, we systematically examined the performance of
131 stringMLST across a phylogenetically diverse set of bacterial pathogens that are
132 of primary interest to Public Health. Our systematic approach compared accuracy
133 of classifications using the standard MLST program vs. stringMLST at varying
134 kmer lengths across many thousands of genomes from 15 different pathogenic
135 species from three highly divergent phyla. Performance was first evaluated across
136 a broad spectrum of phylogenetic diversity using genomes from isolates of
137 *Acinetobacter baumannii (*Phylum: Proteobacteria)*, Clostridium difficile*
138 *(*Phylum: Firmicutes)*, Enterococcus faecium (*Phylum: Firmicutes)*, Escherichia*

139 *coli (*Phylum: Proteobacteria)*, Haemophilus influenzae (*Phylum: Proteobacteria)*,*
140 *Helicobacter pylori (*Phylum: Proteobacteria)*, Klebsiella pneumoniae (*Phylum:*
141 Proteobacteria)*, Mycobacterium tuberculosis (*Phylum: Actinobacteria)*, Neisseria*
142 *gonorrhoeae (*Phylum: Proteobacteria)*, Pseudomonas aeruginosa (*Phylum:*
143 Proteobacteria)*, Streptococcus pneumoniae (*Phylum: Firmicutes)*, Campylobacter*
144 *jejuni* (Phylum: Proteobacteria), *Listeria monocytogenes (*Phylum: Firmicutes),
145 *Salmonella enterica* (Phylum: Proteobacteria), and *Staphylococcus aureus*
146 (Phylum: Firmicutes). To evaluate performance within diverse populations of a
147 single species, we also measured performance of different kmer lengths across 23
148 of the most relevant serovars of *Salmonella enterica* subsp. *enterica* lineage I (*S.*
149 *enterica*). Our results show that optimal performance of stringMLST can, for
150 some species, be achieved with a single kmer length. However, several different
151 bacterial species, and even different serovars of *S. enterica,* require species or
152 serovar-specific kmer lengths for accurate ST classifications. Based on these
153 findings, we further implemented stringMLST into the scalable ProkEvo platform
154 [21], which produces hierarchical genotypic classifications from raw WGS data,
155 and demonstrated how use of species-specific kmer settings for stringMLST
156 enhances computational performance of the platform.
157 [4][5][6][7][66][8][27][30][67][8][9][44] [14][15][27][30][67][16][21][17] [21]

158 # Materials and methods

159 This systems-based comparison between mlst and stringMLST was centered at
160 capturing their differences in computational and statistical performances, and was

161 accomplished through the following steps: 1) Narrow-scope comparative analysis
162 across four phylogenetic distinct pathogens species; 2) Further examination of
163 algorithmic performance within a single ecologically diverse bacterial species;
164 and 3) Wide-scope comparison between phylogenetic divergent pathogenic
165 species with Public Health relevance and with databases available on pubMLST
166 (https://pubmlst.org/) for direct contrast between stringMLST and mlst.
167

168 **Datasets used for narrow-scope analysis**

169 WGS data from four major bacterial pathogens, including *Campylobacter jejuni*,
170 *Listeria monocytogenes*, *Salmonella enterica* subps. *enterica* lineage I (*S.*
171 *enterica*) and *Staphylococcus aureus,* were selected to be used in this first part of
172 the study. Our basis for that choice was due to three *a priori* defined criteria: 1)
173 Select bacterial species from two main phylogenetic divergent Phyla: Firmicutes
174 (*L. monocytogenes* and *S. aureus*) and Proteobacteria (*C. jejuni* and *S. enterica*);
175 2) Select zoonotic pathogens that continually cause human illnesses worldwide
176 [18]; and 3) Consider their epidemiological relevance according to the Centers for
177 Disease Control and Prevention (CDC) [19]. Specifically for *S. enterica*, 20 of the
178 CDC most investigated serovars were represented in the dataset, which includes:
179 *S.* Agona, *S.* Anatum, *S.* Braenderup, *S.* Derby, *S.* Dublin, *S.* Enteritidis, *S.* Hadar,
180 *S.* Heidelberg, *S.* Infantis, *S.* Javiana, *S.* Johannesburg, *S.* Kentucky, *S.* Mbandaka,
181 *S.* Montevideo, *S.* Muenchen, *S.* Newport, *S.* Schwarzengrund, *S.* Senftenberg, *S.*
182 Thompson, and *S.* Typhimurium [20]. All publicly available raw paired-end
183 Illumina reads for these organisms were downloaded from NCBI using parallel-

184 fastq-dump [58]. Genomes used for all analyses were randomly selected from a
185 previously downloaded samples of isolates containing *C. jejuni* (n = 21,919
186 genomes), *L. monocytogenes* (n = 19,633 genomes), *S. enterica* (n = 25,284
187 genomes), and *S. aureus* (n = 11,990 genomes) that were processed through the
188 computational platform ProkEvo [21]. Specifically, our study design was
189 comprised of random sampling of 600 genomes from each species, except for *S.*
190 *enterica* for which 600 genomes were randomly drawn per serovar (list of all 20
191 serovars is shown in S1 Table). For each species and all *S. enterica* serovars, all
192 ~600 genomes were randomly split into three independent batches, with ~200
193 genomes each. The batches were created to measure the degree of variation in
194 classification accuracy when comparing the two ST-based genotyping programs.
195 While for the majority of *S. enterica* serovars there were a total of 600 genomes
196 available, the total number of raw reads publicly available on NCBI and
197 ultimately used for the analyses for *S.* Agona, *S.* Derby, *S.* Johannesburg, *S.*
198 Mbandaka and *S.* Senftenberg was 565, 590, 534, 535 and 563 respectively. The
199 final total number of genomes used per species was *C. jejuni* (n = 600), *L.*
200 *monocytogenes* (n = 600), *S. enterica* (n = 11,787), and *S. aureus* (n = 600). Text
201 file containing all genome NCBI-SRA identifications is available here,
202 https://figshare.com/articles/dataset/_/16735411.
203

204 # Software tools

205 **mlst**

206 mlst is a standard approach for scanning genome assemblies against traditional
207 PubMLST typing schemes [22]. The genome assemblies can be in
208 FASTA/GenBank/EMBL formats [22]. mlst (version 2.16.2) was installed using
209 Anaconda, a package and environment manager that supports maintaining and
210 installing various open-source conda packages [26]. mlst uses genome assemblies
211 as an input. In order to generate assemblies from the raw Illumina paired-end
212 reads, multiple pre-processing steps were performed. Quality trimming and
213 adapter clipping were performed using Trimmomatic [50], while FastQC was
214 used to check and verify the quality of the trimmed reads [51]. The paired-end
215 reads were assembled *de novo* into contigs using SPAdes with the default
216 parameters [52]. The quality of the assemblies was evaluated using QUAST [53].
217 The information obtained from QUAST was used to discard assemblies with 0 or
218 more than 300 contigs, or assemblies with N50 value of less than 25,000 [21].
219 Finally, the assemblies that passed the quality control were used with mlst, where
220 they are categorized into specific variants based on the allele combinations from
221 seven ubiquitous, house-keeping genes [22]. A list of the exact versions of the
222 bioinformatics tools used for generating assemblies for mlst are shown on S2
223 Table. We used mlst with the default options (e.g., *mlst --legacy --scheme*
224 *<scheme> --csv <assembly.fasta> > <output>*) and the following schemes:
225 "senterica" (for *S. enterica*), "campylobacter" (for *C. jejuni)*, "lmonocytogenes"
226 (for *L. monocytogenes*), "saureus" (for *S. aureus*). The distribution of mlst comes

227 with set of pre-downloaded ST schemes. More details about these MLST
228 schemes, such as the number of alleles in the seven genes and the number of ST
229 classifications available are shown on S3 Table. To obtain the ST classifications
230 of all datasets, mlst was run as part of the computational platform ProkEvo [21].
231 Additionally, a separate run of the mlst program was used to conduct a pairwise
232 comparison between the computational performance (runtime and memory usage)
233 of mlst and stringMLST. The used mlst script can be found here,
234 https://github.com/npavlovikj/MLST_stringMLST_analyses/blob/main/scripts/ml
235 st.submit.
236

237 **stringMLST**

238 stringMLST is an assembly- and alignment-free rapid tool for ST-based
239 classification of Illumina paired-end raw reads based on kmers [17]. For the
240 analyses performed in this paper, we used stringMLST version 0.6.3. stringMLST
241 was installed using Anaconda [26]. The first step of using stringMLST was to
242 download the respective MLST scheme from PubMLST. In order to do this, a
243 species name and a kmer length were needed. The default kmer length used and
244 suggested by the developers of stringMLST for reads with lengths between 55 and
245 150 base pairs or nucleotides is 35 (common read length for Illumina paired-end
246 reads) [17]. We used stringMLST with the default options (e.g., *stringMLST.py --*
247 *getMLST --species=<species_name> -P <output_prefix> -k <kmer>*) and the
248 following species names, “Salmonella enterica”, “Campylobacter jejuni”,
249 “Listeria monocytogenes”, “Staphylococcus aureus”, and kmer lengths of 10, 20,

30, 35, 45, 55, 65, 70, 80, 90 independently (https://github.com/npavlovikj/MLST_stringMLST_analyses/blob/main/scripts/stringMLST_dbs.submit). More details about the downloaded MLST schemes, such as the number of alleles in the seven genes and the number of ST classifications available are shown on S3 Table. After the MLST scheme was downloaded and prepared, the final step was to run "stringMLT.py –predict" for the ST classification. For this, we ran stringMLST with the databases *a priori* created and the respective paired-end raw reads and kmer lengths of 10, 20, 30, 35, 45, 55, 65, 70, 80, 90 independently (e.g., *stringMLST.py --predict -d <directory_raw_reads> -p -r -t -x -P <database_prefix> -k <kmer> -o <output>*) (https://github.com/npavlovikj/MLST_stringMLST_analyses/blob/main/scripts/stringMLST.submit). Our choice of using an increasing gradient of kmer lengths was to evaluate whether the kmer length parameter could be optimized to enhance ST-based classification accuracy across bacterial species. Lastly, stringMLST was also integrated as part of the computational platform ProkEvo for a rapid ST-based genotyping as part of a hierarchical genotypic scheme [21][57]. This implementation can be found here, https://github.com/npavlovikj/MLST_stringMLST_analyses/tree/main/Prokevo_stringMLST.

ProkEvo-based MLST classifications

272 In order to compare the ST-based classification accuracy and conduct other
273 statistical analysis (e.g., identifying major contributing factors influencing ST-
274 based classifications) between mlst version 2.16.2 (assembly-dependent) and
275 stringMLST version 0.6.3 (assembly-independent), all initial ST calls for all
276 selected genomes, across all four species (*C. jejuni*, *L. monocytogenes*, *S. aureus*
277 and *S.* Typhimurium), were done using mlst [22] through the computational
278 platform ProkEvo [21]. In brief, ProkEvo uses bacterial Illumina raw paired-end
279 sequences as an input, and the following steps are sequentially done prior to ST-
280 based genotyping using mlst: Trimmomatic for sequence trimming [50], FastQC
281 for quality control of the trimmed reads [51], SPAdes for *de novo* genome
282 assembly [52], and QUAST for quality assessment of the genome assemblies
283 [53]. More information on how to install and use ProkEvo for hierarchical
284 bacterial population genomic analyses can be found here,
285 https://github.com/npavlovikj/prokevo.

286

287 Genome-intrinsic and –extrinsic factors that can
288 influence algorithmic performance

289 Both genome-intrinsic and –extrinsic factors were considered to determine their
290 contribution on the accuracy of ST classifications when comparing mlst vs.
291 stringMLST. The genome-intrinsic variables considered in these analyses were:
292 number of contigs per genome, total number of nucleotides per genome (genome
293 length), GC% content per genome, and dinucleotide composition of genomes. The

294 number of contigs per genome, as well as the genome length, were calculated
295 using the assembled contigs from SPAdes [52]. The number of contigs was
296 calculated for each genome using the Linux "grep" utility (e.g., *grep ">"*
297 *assembly.fasta | wc -l*). The total number of nucleotides per genome was
298 calculated using the "getlengths" function from the AMOS package [54]. For this
299 analysis, we used AMOS v3.1. "getlengths" provides the length for each contig,
300 and a custom Bash script was used to summarize these values per genome. The
301 GC% content was calculated using the program FastQC [51]. FastQC is used to
302 check and verify the quality of the raw Illumina paired-end raw reads. With each
303 pair of raw reads from all datasets, FastQC v0.11 was used. One of the statistics
304 checked for read quality is GC% and this value was extracted with custom Bash
305 script from the file "fastqc_data.txt" once the FastQC output was generated. Since
306 FastQC outputs the GC% per read, the average of both reads was calculated as the
307 final read GC%. The dinucleotide composition of the genomes was calculated
308 with the function "compseq" from the EMBOSS package [55]. "compseq"
309 calculates the frequency of words of a specific length (e.g., length is 2 in the case
310 of dinucleotides) from given input genome sequences. For these analyses we used
311 EMBOSS v6.6 with the command "*compseq -word 2 -outfile <output>*
312 *assembly.fasta*" for all datasets and genomes *a priori* assembled with SPAdes
313 [52]. Next, a customized Bash script was used to count the total number of
314 occurrences of each dinucleotide for each genome across all bacterial species.
315 Finally, all these outputs were merged per genome using custom Python script to
316 facilitate statistical analyses and data visualization. The used scripts can be found

here, https://github.com/npavlovikj/MLST_stringMLST_analyses/tree/main/scripts.

The genome-extrinsic variables used in the analyses presented in this paper were the total count of unique STs per database and the total count of unique alleles across all seven loci used for ST classification across all bacterial species. These genome-extrinsic variables were extracted from the PubMLST databases for both stringMLST and mlst using custom Bash scripts. While the first step of stringMLST is to download the most up-to-date available MLST scheme from PubMLST, the distributed version of mlst comes with set of pre-downloaded ST and allelic schemes. For each MLST scheme, the mlst distribution has a separate directory with 8 files - seven are ".tfa" files with the fasta sequences of the alleles for each locus, and one file (e.g., senterica.txt) contains the ST information (i.e., the total number of STs mapped including their specific allelic composition across all seven loci for that given species). To calculate the total number of unique STs, we used the Linux utility "wc" with the text file with ST information (e.g., *wc -l senterica.txt*). To calculate the total count of unique alleles across the seven loci, the "grep" Linux utility was used with the seven ".tfa" files (e.g., *grep ">" *.tfa | wc -l*). All calculations were done per bacterial species. The downloaded MLST scheme with stringMLST is in a separate directory for each organism and used kmer length. This directory had 12 files - seven are ".tfa" files with fasta sequences for all alleles across all seven loci, and one file has the ST profiles (e.g., Salmonella_enterica_profile.txt), while the remaining files contained information about the extracted kmers and additional config and log information.

340 Similarly, the total number of unique STs for stringMLST was counted using the
341 Linux utility "wc" with the text file with ST profile information (e.g., *wc -l*
342 *Salmonella_enterica_profile.txt*), and the total count of unique alleles per loci was
343 extracted using the "grep" Linux utility with the seven ".tfa" files (e.g., *grep ">"*
344 **.tfa | wc -l*). Similarly, all ST and allelic counts were carried out per bacterial
345 species. With stringMLST, the MLST schemes are downloaded and prepared
346 separately for each different kmer length used. However, the kmer length did not
347 affect the number of STs and unique alleles per organism. Thus, these values
348 remained the same across organisms and kmer lengths for stringMLST.
349

350 Kmer-based distribution across ST programs

351 In order to assess the potential impact of random mapping or occurrence of kmers
352 of different lengths across different bacterial species, we randomly chose 100 raw
353 Illumina paired-end reads from the initial *C. jejuni*, *L. monocytogenes*, *S. aureus*
354 and *S.* Typhimurium (major representative zoonotic serovar of *S. enterica*)
355 isolates. For each read, we extracted all unique kmers of length 10, 20, 30, 35, 45,
356 55, 65, 70, 80 and 90 respectively, and counted their occurrence in the
357 corresponding raw reads. This was done using DSK v2.2.0 [56]
358 (https://github.com/npavlovikj/MLST_stringMLST_analyses/blob/main/scripts/ds
359 k.submit). Next, the total number of kmer frequency was summarized per
360 organism and kmer length, and the mean value was calculated to examine the
361 distribution of different kmers across the raw reads. For each database created
362 with stringMLST, a file with the kmer frequency for the used ST scheme was

363 generated. Using the kmers generated from the raw reads and the stringMLST
364 database, a relative frequency of the common kmers was calculated (calculated as
365 a ratio between the common kmers and the unique kmers from all the kmers
366 generated between the raw reads and the stringMLST database, e.g.,
367 *(common_kmers/unique_total_observations)*100*). The code used for this can be
368 found in our GitHub repository
369 (https://github.com/npavlovikj/MLST_stringMLST_analyses/tree/main/figures_c
370 ode).
371

372 Agreement in ST classification between programs

373 In order to assess the overall accuracy of stringMLST compared to the standard
374 mlst approach for ST calls, a percentage of agreement in ST classification was
375 calculated. For this, the initial dataset composed of 600 genomes from either *C.*
376 *jejuni*, or *L. monocytogenes*, or *S. aureus* was selected, in addition to a total of
377 11,787 genomes across twenty zoonotic serovars of *S. enterica* (~600 genomes
378 per serovar, S1 Table). The program stringMLST was run with increasing kmer
379 lengths ranging from 10 to 90 nucleotides. If both stringMLST and mlst produced
380 identical ST calls, either "good" or "bad" ones, the call was a match. A "good"
381 and "bad" call represent ST with a number or a missing/blank value, respectively.
382 The remaining combinations were classified as a mismatch. Next, the percentage
383 of agreement (concordance) was calculated with custom R base script
384 (https://github.com/npavlovikj/MLST_stringMLST_analyses/tree/main/figures_c
385 ode).

386

387 ## Computational platforms

388 All computational analyses performed for this paper were done on Crane - one of
389 the high-performance computing clusters at the University of Nebraska-Lincoln
390 Holland Computing Center [23]. Crane is Linux cluster, having 548 Intel Xeon
391 nodes with RAM ranging from 64 GB to 1.5 TB. The scalability of ProkEvo with
392 stringMLST was tested on the Open Science Grid (OSG), a distributed, high-
393 throughput computational platform for large-scale scientific research [24][25].
394 OSG is a national consortium of more than 100 academic institutions and
395 laboratories that provide storage and tens of thousands of resources to OSG users.
396 These sites share their idle resources via OSG for opportunistic usage. The OSG
397 resources are Linux-based, and due to the different sites involved, the hardware
398 specifications of the resources are different and vary.

399

400 ## Computational performance

401 To evaluate the computational performance of stringMLST in comparison to the
402 mlst program, we assessed the runtime and memory usage of both programs. For
403 this, we chose four different datasets, *C. jejuni*, *L. monocytogenes*, *S. aureus* and
404 *S.* Typhimurium (major representative zoonotic serovar of *S. enterica*), with three
405 different batches of 200 genomes each, with a total of 600 genomes each. We ran
406 mlst with all required steps, such as quality trimming and adapter clipping, *de*
407 *novo* assembly, and assembly discarding on each dataset (see Section Software
408 tools: mlst for more detailed description). Separately, we ran stringMLST with a

409 range of 10 different kmer lengths (10, 20, 30, 35, 45, 55, 65, 70, 80, 90) on each
410 dataset. For each organism, the runtime was calculated as an average of all 200
411 genomes per batch. In general, the runtime depends on multiple factors, such as
412 the specification and capabilities of the used computational platform. Since the
413 runtime can vary depending on these various factors, average statistics were used
414 to show the central tendency of the runtime when comparing stringMLST vs.
415 mlst. The runtime was calculated using the "date" command integrated in the
416 Unix operating systems (e.g., *t=`date +%s`; mlst --legacy --scheme senterica --*
417 *csv assembly.fasta > <output>; tt=`date +%s`; total_time=$((tt-t))*). For each
418 organism, the memory was calculated as the maximum memory recorded from all
419 200 genomes per batch, since all genomes were analyzed separately and
420 concurrently. In the case of mlst, the recorded memory was the maximum
421 memory of all the steps ran prior to mlst, such as trimming, *de novo* assembly,
422 quality checking, filtering, and ST typing. The memory used for these steps
423 considerably varies from a few MBs to a few GBs (e.g., filtering vs. *de novo*
424 assembly), and since the memory is a physical limitation of the computational
425 platform, the maximum used memory was calculated for each organism and
426 batch. The memory used was calculated using the "cgget" command that tracks
427 various parameters from the Linux Control Groups (cgroups) per running job
428 (e.g., *mlst --legacy --scheme senterica --csv assembly.fasta > <output>; r=`cgget*
429 *-r memory.usage_in_bytes /slurm/uid_${UID}/job_${SLURM_JOBID}/`;*
430 *mem=`echo $r | awk -F: '{print $3}'*).
431

Incorporating stringMLST in ProkEvo

ProkEvo is a freely available and scalable computational platform capable of facilitating bacterial population genomics analyses while combining various independent algorithms in a portable pipeline [21]. One of the advantages of ProkEvo is its ability to facilitate the addition and removal of new steps and programs without disrupting its workflow. For instance, more details about adding new programs to ProkEvo are given here https://github.com/npavlovikj/ProkEvo/wiki/4.1.-Add-new-bioinformatics-tool-to-ProkEvo. By following these instructions, we were able to successfully add stringMLST to the current ProkEvo platform. The ultimate description of how stringMLST was integrated into ProkEvo can be found here, https://github.com/npavlovikj/MLST_stringMLST_analyses/tree/main/Prokevo_stringMLST.

Comparison between mlst and stringMLST performance using ProkEvo

In order to compare the performance/accuracy of MLST and stringMLST as part of the ProkEvo platform, two subsets of the *C. jejuni*, *L. monocytogenes*, *S.* Typhimurium and *S. aureus* datasets used in this paper were selected. One subset was composed of 100 randomly selected genomes, while the second one contained 1,000. The subsets were randomly selected from the original datasets used in this paper. As part of ProkEvo, stringMLST was run with the default kmer

454 length of 35. The ProkEvo workflows with mlst and stringMLST and the two
455 datasets were individually run on Crane - one of the high-performance computing
456 clusters at the Holland Computing Center. Once the four workflows finished, the
457 performance of ProkEvo with mlst and stringMLST and the datasets with 100 and
458 1,000 genomes, respectively, was compared using: i) the total running time; ii) the
459 percentage of non-classified STs; and iii) the percentage of agreement between
460 programs. Since ProkEvo is an automated platform, a list of NCBI-SRA
461 identifications was provided with the ProkEvo implementations with both mlst
462 and stringMLST. In brief, ProkEvo manages all the dependencies and
463 intermediate steps, and produces the final ST classification as an output.
464

465 stringMLST-based kmer length optimization across
466 phylogenetic divergent bacterial pathogens

467 To identify the optimal species-specific kmer length that minimizes the frequency
468 of ST miscalls, we ran stringMLST with a range of different kmer lengths across
469 phylogenetic divergent pathogenic species. First, we chose twenty-three *S.*
470 *enterica* serovars (*S.* Agona, *S.* Anatum, *S.* Braenderup, *S.* Derby, *S.* Dublin, *S.*
471 Enteritidis, *S.* Hadar, *S.* Heidelberg, *S.* Infantis, *S.* Javiana, *S.* Johannesburg, *S.*
472 Kentucky, *S.* Mbandaka, *S.* Montevideo, *S.* Muenchen, *S.* Newport, *S.*
473 Oranienburg, *S.* Poona, *S.* Saintpaul, *S.* Schwarzengrund, *S.* Senftenberg, *S.*
474 Thompson, *S.* Typhimurium), and for each dataset we randomly selected 100
475 paired-end Illumina reads from NCBI-SRA. Second, for each dataset we ran mlst
476 and stringMLST with kmer lengths ranging from 20, 30, 35, 40, 45, 50, 55, 60,

477 65, 70, 80, 90. The kmer length of 10 was excluded due to its poor performance in
478 previous analyses. Additionally, we use data from fourteen pathogens with Public
479 Health relevance and with available MLST schemes, to widen the scope of the
480 analysis and assess the necessity of fine-tunning the kmer length on a more
481 broadly selected collection of species. In particular, we chose the following
482 pathogens: *Acinetobacter baumannii, Clostridioides difficile, Enterococcus*
483 *faecium, Escherichia coli, Haemophilus influenzae, Helicobacter pylori,*
484 *Klebsiella pneumoniae, Mycobacterium tuberculosis, Neisseria gonorrhoeae,*
485 *Pseudomonas aeruginosa, Streptococcus pneumoniae, Campylobacter jejuni,*
486 *Listeria monocytogenes,* and *Staphylococcus aureus*. For each pathogen, we
487 randomly selected and downloaded 1,000 paired-end reads from NCBI-SRA and
488 processed these reads separately with mlst and stringMLST. stringMLST was run
489 with kmer lengths ranging from 20, 30, 35, 45, 55, 65, 70, 80, 90 and different
490 schemes for the different pathogens. Similar to the *S. enterica* datasets, the kmer
491 length of 10 was excluded from the analysis.
492 Across all datasets, the percentage of ST miscalls was calculated for
493 stringMLST for each kmer length, whereby miscalls were defined as "bad" ST
494 calls - calls with a missing or blank values. Next, for each dataset, the kmer length
495 that equated with the lowest percentage of ST miscalls was recorded. For some
496 datasets, multiple kmer lengths generated an identical lowest percentage for ST
497 miscalls. In this case, we applied a two-folded approach to select the most optimal
498 kmer length: 1) if kmer of length 35 was part of the array of kmer lengths that
499 showed the most optimal results, we recorded kmer 35 as the optimal kmer length

500 since that is the default and recommended value for stringMLST (parsimonious
501 approach); or 2) if kmer of length 35 was not part of the kmer lengths that showed
502 the most optimal results, we recorded the kmer with the highest value as the most
503 optimal one, since in general our analysis showed that longer kmers consumed
504 less computational resources and speed up the entire analysis. Ultimately, the
505 optimal kmer length and the percentage of ST miscalls were visualized onto a
506 core-genome phylogeny generated for all twenty-three *S. enterica* serovars, as
507 well as for all fourteen pathogens including all twenty-three *S. enterica* serovars
508 which jointly totaled fifteen pathogens (total of 37 genomes, one per species
509 including one per serovar of *S. enterica*, were used to construct the 16S rRNA-
510 based phylogeny for visualization purposes). The core-genome alignment for the
511 twenty-three *S. enterica* serovars was generated using Roary with this set of
512 parameters, "*roary -s -e --mafft -p 8 -cd 99 -i 95 ./prokka_output/*.gff -f*
513 *roary_output*" and the phylogenetic tree was produced using FastTree [76]. The
514 phylogenetic tree for the fifteen pathogens was generated using CLUSTALW
515 (https://www.genome.jp/tools-bin/clustalw) with the 16S rRNA sequences of the
516 selected 37 genomes generated by Prokka [78]. All phylogeny-based
517 visualizations were done using iTOL [77], and the recorded and overlaid statistics
518 were extracted with custom R scripts
519 (https://github.com/npavlovikj/MLST_stringMLST_analyses/blob/main/figures_c
520 ode/figures_code.Rmd).
521 In addition to calculating the percentage of ST miscalls for different kmer
522 lengths with stringMLST, for each dataset we calculated the percentage of

523 agreement (concordance) between mlst and stringMLST on ST calls ("good" or
524 "bad"), as previously described here. Of note, when the stringMLST and mlst
525 results were combined, the number of returned ST calls was not always 1,000 (the
526 original size of the used datasets). If 1,000 reads are used with stringMLST,
527 stringMLST generates ST calls for all 1,000 reads. On the other hand, when using
528 mlst, a set of steps are used before mlst, including filtering, and a fraction of
529 assemblies were disregarded due to poor quality. Thus, only genome sequences
530 that passed through the mlst program and yielded a "good" or "bad" call were
531 ultimately used to compare with stringMLST. The number of raw reads for each
532 dataset, as well as the number of final reads from mlst used for these analyses are
533 shown on S4 Table.
534

535 Statistical analyses

536 To compare the overall performance and accuracy of mlst vs. stringMLST on ST-
537 based classifications, the following statistics were used across all bacterial species
538 datasets: 1) ST richness; 2) Simpson's D index (1 – *D*) of diversity using ST
539 counts as input data; 3) Proportion of non-classified STs (missing values or blank
540 calls); and 4) Standard deviation of the proportion of non-classified STs. These
541 statistics were calculated to evaluate the algorithmic performance on ST-based
542 classification accuracy within and between bacterial species selected to be used in
543 the narrow scope analysis (*C. jejuni*, *S. aureus*, *L. monocytogenes*, and *S.*
544 *enterica*). ST richness was calculated by identifying the number of distinct STs
545 present in each species. The Simpson's D index of diversity (1- *D*) was used to

546 calculate the degree of genotypic diversity across species, using the diversity()
547 function available in the vegan (version 2.5-6) R library [29]. The proportion of
548 non-classified STs was calculated using the counts of isolates or genomes that
549 were not assigned a ST number after each run of either mlst or stringMLST. The
550 standard deviation of the proportion of non-classified STs was calculated using
551 the sd() function which is derived from an unbiased estimate of the sample
552 variance corrected by $n - 1$ (n for number of observations). The frequency of
553 genomes used for all analyses was calculated per batch and program across all
554 species, including across serovars for *S. enterica*. The relative frequency of the
555 most dominant ST lineages was also assessed across bacterial species.
556 PERMANOVA univariate or multivariate models were used to assess the
557 significance ($p < 0.05$) and the degree of association between the genome-intrinsic
558 and –extrinsic factors with the following dependent variables: ST richness,
559 Simpson's D index of diversity, or proportion of non-classified STs. Statistical
560 models were built for each of the dependent variables separately. Multivariate
561 models included either the combination of bacterial species and program, or
562 serovars in the case of *S. enterica* and program. These multivariate models were
563 stated to calculate the main and synergistic effects of the explanatory variables
564 (e.g., species*program or serovar*program). Univariate models were also
565 assessed for each of the dependent variables, using one of the following
566 independent/explanatory variables: 1) Genome-intrinsic variables: median number
567 of contigs, mean of the total count of nucleotides per genome, mean of the
568 average GC% content per genome, standard deviation of the number of contigs,

569 standard deviation of the total count of nucleotides per genome, and standard
570 deviation of the average GC% content per genome; 2) Genome-extrinsic
571 variables: species, serovar of *S. enterica*, program (mlst vs. stringMLST with
572 kmer lengths of 10, 20, 30, 35, 45, 55, 65, 70, 80, 90), mean of the total count of
573 unique STs per program, mean of the total count of unique alleles across all genes
574 per program, and the Simpson's D index of diversity per species. Statistical
575 significance and strength of association between the dependent and independent
576 variables were measured with *p*-values ($p < 0.05$) and *R*-squared, respectively. In
577 the case of contig size (median), total number of nucleotides per genome (mean),
578 and GC% content per genome, summary statistic values (median or mean) were
579 calculated grouped by species and batch (there was a total of three batches per
580 bacterial species or serovar). For the total count of STs and total number of alleles
581 in the database, summary statistic values (mean) were calculated grouped by
582 species, batch, and program. Lastly, the standard deviation of number of contigs,
583 total count of nucleotides per genome, or GC% content per genome were
584 calculated grouped by species. PERMANOVA models were run using the
585 adonis() function with 1,000 permutations using the vegan (version 2.5-6) R
586 library [29]. Principal component analysis (PCA) was used to analyze the
587 dinucleotide distribution across species and across serovars for *S. enterica* with
588 two dimensions using the prcomp() function. The PCA calculations and the
589 selection of the number of PCs were done using the factoextra (version 1.0.7)
590 library. Bar-plots, box-and-whiskers plots, and bivariate/trivariate scatter plots
591 were used to assess and visualize the distribution and associations within and

592 between dependent and independent/explanatory variables. The R software
593 (version 4.0.3) and R libraries such as Tidyverse (version 1.3.0) were used to
594 conduct all statistical analyses, and all R scripts are available here
595 (https://github.com/npavlovikj/MLST_stringMLST_analyses/tree/main/figures_c
596 ode). Data quality control was achieved with R base functions, in addition to the
597 following packages: skimr (version 2.1.3) and visdat (version 0.5.3). Graphical
598 visualizations were achieved using ggplot2 (version 3.3.2), GGally (version
599 2.1.2), and plotly (version 4.9.4.1). R code integrity was checked using the
600 assertive (version 0.3-6) package.
601

602 Results

603 The computational and analytical approaches used in this paper are shown on Fig
604 1. Our analytical approach was sub-divided into a narrow- and wide-scope
605 analysis aiming at accomplishing two goals: 1) Comparing the computational
606 performance and accuracy of MLST vs. stringMLST; 2) Optimizing the use of
607 stringMLST for a wide range of bacterial species; and 3) Implementing
608 stringMLST as part of the ProkEvo computational genomics platform. We
609 initially used publicly available raw Illumina paired-end sequence data from *C.*
610 *jejuni*, *L. monocytogenes*, *S. enterica* and *S. aureus*, to run stringMLST and mlst
611 independently in order to compare the accuracy in ST-based classifications and
612 assess the computational needs and performance in the overall analysis (narrow-
613 scope step). For this narrow-scope analysis, we performed a detailed comparative

614 analysis between these two programs including: i) analyses of computational
615 performance and resources needed (e.g., average runtime per genome and
616 maximum memory needed to analyze all genomes), and ii) statistical analyses to
617 determine the accuracy of classifications (e.g., ST richness, Simpson's D index of
618 ST-based diversity, proportion of miscalls, and percentage of agreement or
619 concordance between programs). For the wide-scope step of the analysis, we
620 systematically evaluated the accuracy and concordance between mlst and
621 stringMLST across a broader array of phylogenetic divergent pathogens with
622 direct implication for Public Health (*A. baumannii, C. difficile, E. faecium, E.*
623 *coli, H. influenzae, H. pylori, K. pneumoniae, M. tuberculosis, N. gonorrhoeae, P.*
624 *aeruginosa, S. pneumoniae, C. jejuni, L. monocytogenes, S. enterica* and *S.*
625 *aureus*). Combined with the intra-species analysis done across 23 serovars of *S.*
626 *enterica*, our assessment aimed at revealing the optimized kmer length to be used
627 with stringMLST in order to: i) minimize the percentage of ST miscalls, and ii)
628 maximize the use of computational resources by speeding up the analysis. Lastly,
629 we provided an implementation of stringMLST within ProkEvo - a freely
630 available and scalable computational platform that facilitates hierarchical
631 genotyping of bacterial populations including pan-genomic mapping [21].
632

633 # Computational performance

634 The computational performance between stringMLST and mlst was measured
635 using two metrics: 1) The average computational runtime per genome; and 2) The
636 maximum memory used per dataset. The average runtime in minutes per genome

637 per batch between mlst and stringMLST with different kmer lengths, for *C. jejuni*,
638 *L. monocytogenes*, *S. aureus*, and *S.* Typhimurium (major representative of *S.*
639 *enterica*), is shown on Fig 2. While the runtime of mlst varies between 20 and 80
640 minutes per genome depending on the dataset used, all stringMLST runs with
641 different kmers finished within a few minutes (ranging from ~1-16 minutes when
642 kmer 10 was included and ~1-5 minutes when kmer 10 was excluded). Apart from
643 stringMLST with kmer 10, all other kmer lengths showed a uniform runtime. The
644 longer runtime observed with kmer 10 can be partially explained by the higher
645 number of kmers that were generated and used for mapping (S1 Fig). The
646 obtained results show that ST-based classifications are accomplished considerably
647 more rapidly when carried out using stringMLST compared to the standard MLST
648 program.
649 Additionally, a comparison of maximum memory used when both stringMLST
650 and mlst were run for *C. jejuni*, *L. monocytogenes*, *S. aureus*, and *S.* Typhimurium
651 (major representative of *S. enterica*) is shown on S2 Fig. Across all species, the
652 range of maximum memory usage for mlst and stringMLST (across all kmers)
653 was ~2-16GBs and ~3-30GBs respectively. Although the memory used across
654 datasets is variable, none of the analyses we ran exceeded 30GBs of RAM. Since
655 most high-performance computers can consistently provide resources from 32GBs
656 to a few TBs of RAM, the memory available should not be considered a
657 bottleneck for running either program.
658

659 ## Factors that can influence ST-based classification

660 First, we describe the characteristics and composition of the data utilized for
661 comparison between programs regarding ST-based classification in the narrow-
662 scope approach (utilization of fewer phylogenetic diverse pathogen datasets). The
663 frequency of genomes utilized per species and across programs is shown on S3A-
664 D Fig. The frequency of *S. enterica* genomes was higher than other species
665 because an equal sample of ~600 genomes was taken from 20 representative
666 zoonotic serovars (S4A-D Fig). An assessment of the proportion of the most
667 dominant STs across species (proportion ≥ 2% - S5A-D Fig) or serovars of *S.*
668 *enterica* (proportion ≥ 15% - S4N Fig) initially revealed a similar ST-based
669 distribution across programs. Furthermore, genome-intrinsic and -extrinsic factors
670 that could potentially impact the mlst vs. stringMLST algorithmic comparison and
671 performance were *a priori* determined in the analysis. Among the genome-
672 intrinsic factors considered across species were the number of contigs per genome
673 (Fig 3A), the total number of nucleotides per genome (Fig 3B), GC% content per
674 genome (Fig 3C), and the distribution and composition of dinucleotides per
675 species (Fig 3D and S3E-F Fig). Similarly, the distribution of the genome-
676 intrinsic factors was analyzed across all twenty serovars of *S. enterica* (S4G-L
677 Fig). A correlogram (pairwise correlation analysis) was also used to assess the
678 bivariate correlation (Pearson's correlation coefficient) across genome-intrinsic
679 variables, for either all four bacterial species (S3G Fig) or serovars across *S.*
680 *enterica* (S4M Fig). At large, the differences observed in the distribution of

genomic-intrinsic variables were species driven, with a strong uniformity found across serovars of *S. enterica*.

As for the genome-extrinsic variables, the total count of unique STs (for species - Fig 3E) and unique number of alleles across all seven loci (for species - Fig 3F), and across all batches were selected as factors that could influence the comparative analysis between mlst and stringMLST. Similarly, the genome-extrinsic variables were analyzed across all twenty serovars of *S. enterica* (S4E-F Fig). Of note, database size differences (number of STs and alleles) may directly influence the number of miscalls since it is expected that the larger the database is, the more likely STs are to be classified, or to find a match, and not be miscalled [30]. Considering the differences in genome-intrinsic and -extrinsic variable distribution across species, such factors were further utilized for assessing their statistical contribution in the accuracy of ST-based classification between mlst and stringMLST.

Assessing the contribution of genome-intrinsic and –extrinsic variables

In order to assess the statistical association and contribution of each genomic-intrinsic and -extrinsic variable onto the accuracy of mlst vs. stringMLST on ST calls (narrow-scope analysis since it only included four bacterial species, *C. jejuni*, *S. aureus*, *L. monocytogenes*, and *S. enterica*), the following dependent variables (outcomes) were used in the PERMANOVA models: 1) ST richness (Fig 4A); 2) Simpson's D index of ST diversity (Fig 4B); and 3) Proportion of

704 non-classified STs (Fig 4C). Additionally, the standard deviation of the proportion
705 of non-classified STs was measured as an auxiliary metric for accuracy (Fig 4D).
706 At the species level, a multivariate model approach was used to examine the
707 interaction of species and program (mlst vs. stringMLST), whereas all remaining
708 analyses were done using univariate models containing each genome-intrinsic and
709 -extrinsic variable for all three outcomes (S6A-L Fig, S7A-K Fig, S8A-L Fig).
710 For each variable, the significance and strength of association were assessed by
711 jointly examining the *p*-value ($p < 0.05$) and *R*-squared, respectively. For both ST
712 richness (Fig 4A) and the Simpson's D index of diversity (Fig 4B), the difference
713 between species explained most of the variation with ~98.3% and ~99%,
714 respectively. As expected, based on the phylogenetic divergence of the four
715 chosen pathogens, differences across species could largely be explained by
716 genome-intrinsic variables associated with genome composition, such as: GC%
717 content (*p* ~0.0009, *R*-squared ~44%) for ST richness, and the number of contigs
718 per genome (*p* ~0.0009, *R*-squared ~39.5%) for the Simpson's D index of
719 diversity (Fig 4A-B). Notably, for both ST richness and the Simpson's D index of
720 diversity most of the differences between species could be explained by variation
721 in genome composition (Fig 4A-B). Not surprisingly, co-linearity was observed
722 between ST richness and the Simpson's D index of diversity across species (Fig
723 4A). In the case of the proportion of non-classified STs (ST miscalls) (Fig 4C),
724 most of the variation was explained by inter-species differences (*p* ~0.0009, *R*-
725 squared ~33%), with the number of contigs per genome being the most important
726 genome-intrinsic contributing factor (*p* ~0.0009, *R*- squared ~27%). As for the

727 kmer length parameter used by stringMLST, results for ST richness and the
728 Simpson's D index of diversity were uniform across all lengths (Fig 4A-B).
729 However, when examining the proportion of miscalls (Fig 4C) and the standard
730 deviation of that proportion (Fig 4D), the data pointed toward the optimal kmer
731 length being between 35 and 65 across all four species due to the intrinsic
732 variance within the *S. enterica* data (narrow scope analysis). Specifically, this
733 kmer length range was defined based on two criteria: i) minimization of the
734 proportion of miscalls; and ii) less variation (standard deviation) around the
735 average of ST-based miscalls. Of note, mlst demonstrated the highest proportion
736 of miscalls and standard deviation of that proportion for both *L. monocytogenes*
737 and *C. jejuni* (Fig 4C-D), and the kmer length 10 for stringMLST yielded very
738 low accuracy and null results for ST richness and Simpson's D index of diversity
739 (Fig 4A-D). Differences between species across ST richness, Simpson's D index
740 of diversity, and proportion of ST miscalls along with all genome-intrinsic and -
741 extrinsic variables across programs (mlst vs. stringMLST) were further examined
742 here (Fig 5A-D, S9A-O Fig). In general, differences in ST-based calls across
743 programs were largely influenced by the bacterial species dataset.
744 Given the complexity and diversity of the *S. enterica* population structure [12],
745 the stringMLST performance was analyzed across twenty zoonotic serovars
746 (S4O-R Fig), and resulted in a significant and predominant contribution of the
747 "serovar groupings" across all outcomes and PERMANOVA models (S10A-L
748 Fig, S11A-K Fig, S12A-L Fig): ST richness (p ~0.0009, R-squared ~75.4%),
749 Simpson's D index of diversity (p ~0.0009, R-squared ~88%), and proportion of

750 ST miscalls (p ~0.0009, R- squared ~35.4%). By assessing the distribution of the
751 model outcomes, along with PERMANOVA model results and bivariate
752 association between dependent and explanatory variables (S13A-R Fig), the
753 results recapitulated the species-level results with the optimal kmer length for
754 stringMLST being around 35 and 65, but also revealed the need to consider
755 difference across *S. enterica* serovars prior to implementation. Combined, these
756 accuracy-based results suggest that: i) stringMLST minimizes the ST miscalls
757 compared to mlst in a species-specific fashion, and by consequence the optimal
758 kmer length for stringMLST ranged from 35 to 65 overall; ii) the performance
759 and accuracy of stringMLST can vary across species and serovars of *S. enterica*
760 allowing for data-driven fine-tunning of the kmer length; and iii) the use of
761 sequence platform with longer reads, which would maximize the number of
762 contigs per genome, could directly alter both mlst and stringMLST accuracy in
763 ST calls across species.
764

765 # Concordance between programs

766 Concordance between programs was calculated as the percentage of cases in
767 which outputs from both mlst vs. stringmlst agreed in the call ("good" or "bad").
768 Results demonstrating the percentage agreement in ST calls between mlst and
769 stringMLST with different kmer lengths are shown on Fig 6. Apart from kmer 10,
770 across all species, the percentage of agreement between mlst and stringMLST
771 varies between ~81% and 97%. In the case of *L. monocytogenes*, *C. jejuni*, and *S.*
772 *aureus*, the kmer length of 35 appears to be the optimal value to reach the same

773 accuracy as mlst, which matches the original default and recommended parameter
774 value for stringMLST [17]. However, for *S. enterica*, a higher percentage of
775 agreement with MLST was achieved for kmer lengths of 55 and 65 (Fig 6). This
776 *S. enterica*-related observation recapitulated the initial findings of decreased
777 proportion of ST miscalls with higher kmer lengths (Fig 4C). Of note, our finding
778 collectively showed that the kmer length of 10 yielded low accuracy when
779 compared to mlst and other stringMLST kmer lengths. The most likely
780 explanation for lower accuracy generated by kmer 10 is that shorter kmers are
781 more likely to map unambiguously onto a genome when compared to other
782 lengths. That high frequency of kmer length 10 on a given dataset reflects their
783 higher likelihood of mapping to multiple regions of a genome (S1 Fig, S14A-B
784 Fig). Overall, stringMLST is a rapidly deployable and optimizable ST-based
785 genotyping algorithm that in this narrow-scope analysis proved to be applicable to
786 four phylogenetic distinct pathogens.
787

788 Optimization of stringMLST kmer length across
789 phylogenetic divergent species

790 Here, we systematically investigated what kmer length would give the fewest ST
791 miscalls (optimized length) with stringMLST across a broader array of
792 phylogenetic divergent pathogens. Given our previous results, we first fine-tunned
793 our investigation into the *S. enterica* population given the genetic and ecological
794 diversity across serovars. For that, we selected data from twenty-three *S. enterica*
795 zoonotic serovars and ran stringMLST with wide range of kmer lengths (20, 30,

796 35, 40, 45, 50, 55, 60, 65, 70, 80, 90). Fig 7A shows the core-genome phylogeny
797 mapping of the optimized kmer length across all twenty-three serovars along with
798 their corresponding percentage of ST miscalls. More detailed information on the
799 distribution of the percentage of ST miscalls for all used kmer lengths (20, 30, 35,
800 40, 45, 50, 55, 60, 65, 70, 80, 90) is shown on S15A Fig. As it can be seen on Fig
801 7A, many serovars (*S*. Anatum, *S*. Braenderup, *S*. Javiana, *S*. Mbandaka, *S*.
802 Montevideo, *S*. Oranienburg, *S*. Poona, *S*. Schwarzengrund, *S*. Senftenberg, *S*.
803 Typhimurium) have 0% of miscalls when the default kmer length 35 was used. *S*.
804 Infantis and *S*. Derby show the lowest percentage of ST miscalls (3% and 2%
805 respectively) with higher value of kmer, e.g., 90. Interestingly, *S*. Saintpaul
806 showed the highest percentage of ST miscalls when only considering the range of
807 kmer lengths used for the initial analyses (10-90). To investigate this further, we
808 ran stringMLST for *S*. Saintpaul with kmer lengths up to 240 (240 was chosen
809 because the maximum read length for the *S*. Saintpaul dataset is 250 base pairs or
810 nucleotides) (S15C-D Fig). As it can be seen on S15C Fig, the fewest ST miscalls
811 for *S*. Saintpaul were produced when kmer of length 140 was used (22%). When
812 comparing the percentage of ST miscalls between mlst and stringMLST, mlst
813 outperformed stringMLST for the used datasets and range of kmer lengths (the
814 percentage of miscalls for mlst ranged from ~0-1% across all serovars, while the
815 percentage of miscalls for stringMLST ranged from ~0-85% depending on the
816 serovar and kmer length used, with 2% being the median across all combinations
817 of serovar and kmer length for stringMLST). In addition to the percentage of ST
818 miscalls, we calculated the percentage of ST agreement between mlst and

819 stringMLST with the range of kmer lengths (S15B Fig). While for some serovars
820 this percentage is the highest when kmer with length 35 is used (e.g., *S*. Anatum,
821 *S*. Braenderup, *S*. Javiana, *S*. Mbandaka, *S*. Montevideo, *S*. Oranienburg, *S*.
822 Poona, *S*. Schwarzengrund, *S*. Senftenberg, *S*. Typhimurium), for other serovars
823 (e.g., *S*. Derby, *S*. Dublin, *S*. Enteritidis, *S*. Hadar, *S*. Heidelberg, *S*. Infantis, *S*.
824 Kentucky, *S*. Saintpaul) the percentage of ST agreement between the two
825 programs was higher with higher kmer lengths (the percentage of ST agreement
826 varied from ~14-100% depending on the serovar and kmer length used, with 97%
827 being the median across all combinations of serovar and kmer length).
828 In order to widen the scope of our phylogenetic-based analysis, we assessed the
829 percentage of ST miscalls across varying kmer lengths for divergent bacterial
830 pathogens with Public Health relevance. We selected 14 distinct organisms and
831 ran stringMLST with wide range of kmer lengths (20, 30, 35, 45, 55, 65, 70, 80,
832 90). Fig 7B depicts the 16S rRNA-based phylogeny overlaid with the optimal
833 kmer length that minimized the percentage of ST miscalls. Of note, the phylogeny
834 contained fourteen distinct pathogens and twenty-three genomes across each
835 serovar of *S. enterica.* The distribution of the percentage of ST miscalls for all
836 used kmer lengths (20, 30, 35, 40, 45, 50, 55, 60, 65, 70, 80, 90) is shown on
837 S15E Fig. While the percentage of ST miscalls varied between 0% and 22%
838 across the *S. enterica* serovars as shown in Fig 7A, the percentage of miscalls was
839 expectedly more variable for the fourteen bacterial pathogens, ranging from 1.2%
840 to 74.9%. The datasets for *A. baumannii*, *C. jejuni*, *H. influenzae*, *K. pneumoniae*,
841 *L. monocytogenes*, *N. gonorrhoeae*, *S. aureus* and *S. pneumoniae* showed the

842 lowest percentage of ST calls with the default kmer length of 35. *C. difficile* and
843 *M. tuberculosis* had minimized ST miscalls with kmer lengths of 20 and 30
844 respectively, while *P. aeruginosa* with kmer length of 65. Interestingly, for *E.*
845 *faecium* and *H. pylori*, the optimal kmer lengths were 35 and 20, even though the
846 percentage of miscalls was high (74.9% and 67.6%). To further investigate this,
847 we ran stringMLST for *E. faecium* and *H. pylori* with kmer lengths up to 140 (140
848 was chosen because the maximum read length for the two datasets is 150 base
849 pairs or nucleotides) (S15G-H Fig, S15K-L Fig). As can be seen on the Figures,
850 the percentage of miscalls was higher with higher kmer lengths, and the lower
851 kmer lengths yielded fewer miscalls, albeit this number was still considerably
852 high. Additionally, we ran stringMLST on another set of randomly selected 100
853 paired-end reads for *E. faecium* (S15M-N Fig), *H. pylori* (S15I-J Fig) and
854 *Enterococcus faecalis* (S15O-P Fig). These 100 reads were not part of the initial
855 datasets and were chosen to validate that the initial random data selection was not
856 completely biased. We also added *E. faecalis* here due to its close phylogenetic
857 association with *E. faecium*. For *E. faecium* and *H. pylori* we observed the same
858 pattern with 100 reads as with 1,000 reads. On the other hand, the pattern for *E.*
859 *faecalis* was quite opposite with lowest percentage of ST miscalls of 5.43% for
860 kmer 35. When comparing the percentage of ST miscalls between mlst and
861 stringMLST, for some datasets, such as *C. jejuni*, *H. pylori*, *L. monocytogenes*, *M.*
862 *tuberculosis*, *N. gonorrhoeae*, *S. aureus*, mlst performed worse than stringMLST
863 (with increase in percentage of miscalls by ~48%, 13%, 11%, 50%, 9%, 45%
864 respectively). In addition to the percentage of ST miscalls, we calculated the

865 percentage of ST agreement between mlst and stringMLST with the range of
866 kmer lengths (S15F Fig). Of note, in the case of stringMLST, when the optimal
867 kmer length was above the default parameter of 35, the ultimately selected kmer
868 length was picked based on our empirical evidence for longer kmers being
869 capable of speeding up the computational analysis.

870 In summary, while the default kmer length of 35 used by stringMLST performs
871 accurately across many organisms, our systems-based approach encompassed the
872 analysis of a broad array of phylogenetic divergent organisms revealed: i) intra-
873 and inter-species variation in the percentage of ST miscalls requires fine-tunning
874 of the kmer length parameter; ii) lack of association between taxonomy or
875 phylogenetic placement of organisms and the optimal kmer length; and iii) unique
876 species behave as outliers for which stringMLST cannot be directly applied with
877 the default settings.

878

879 Incorporating stringMLST in ProkEvo

880 ProkEvo was recently developed as an automated and scalable computational
881 platform for bacterial population genomics analyses that uses the Pegasus
882 Workflow Management System (WMS) [31]. In particular, ProkEvo facilitates
883 the use of a hierarchical approach for population stratification with different
884 layers of genotypic resolution. MLST-based classification of genomes into STs is
885 part of this hierarchical approach that has been proven to be predictive of
886 ecological traits such as AMR in *S. enterica* lineages [57]. However, ProkEvo
887 currently only uses the standard mlst algorithm for ST calls [21]. As part of this

888 paper, the stringMLST program was incorporated into ProkEvo without any
889 disruption in its workflow. The workflow design of ProkEvo with both mlst and
890 stringMLST is shown on S16 Fig.

891 In order to compare the performance of ProkEvo with mlst and stringMLST,
892 randomly shuffled subsets derived from the original datasets used for *C. jejuni*, *L.*
893 *monocytogenes*, *S.* Typhimurium, and *S. aureus* were used. One random subset
894 contained 100 genomes, while the second one had 1,000 genomes. ProkEvo was
895 run using either mlst or stringMLST on Crane, one of the high-performance
896 computing clusters at the Holland Computing Center [23]. For mlst, the pipeline
897 used was previously established and included a few required steps, such as quality
898 trimming and adapter clipping, *de novo* assembly and assembly discarding prior
899 to the ST mapping [21]. Based on the inter-species results shown here (Fig 6), the
900 default kmer length of 35 was used with stringMLST for this comparison. The
901 outcomes measured for this analysis were: i) total running time (Fig 8A); ii) the
902 percentage of non-classified STs (Fig 8B); and iii) the percentage of agreement
903 between programs (Fig 8C).

904 While the runtime of using ProkEvo with mlst varied from ~8 to 34 hours for
905 the subset containing 100 genomes, the runtime of ProkEvo with stringMLST
906 varied from ~25 minutes to 3 hours (Fig 8A). Similarly, for the larger datasets
907 containing 1,000 genomes, the runtime of ProkEvo with mlst varied from ~17 to
908 39 hours, while the runtime of ProkEvo with stringMLST varied from ~4 to 8
909 hours. Regardless of the pathogen species tested, stringMLST speeded up the
910 analyses ~4 times when utilizing 1,000 genomes across species.

911 In terms of accuracy in ST classifications, the use of stringMLST considerably
912 decreased the number of non-classified STs, regardless of the dataset size (100 or
913 1,000 genomes) and bacterial species (Fig 8B). In accordance, stringMLST
914 resulted in a higher frequency of genomes classified as novel STs (ST numbers
915 that were not classified by mlst) (S17 Fig). Additionally, the overall concordance
916 between mlst and stringMLST varies from 82% to 100% across all datasets. The
917 percentage of agreement is the lowest for *S.* Typhimurium, while it is the highest
918 for *S. aureus* (Fig 8C). The lower proportion of ST miscalls and high percentage
919 of agreement between programs for *S. aureus*, compared to other species, is
920 associated with its higher degree of genetic homogeneity (fewer dominant STs)
921 (S5 Fig). This difference in miscalls and concordance between programs may be
922 further explained by the variation in database sizes, since the PubMLST schemes
923 used for mlst have fewer alleles across all seven loci which results in fewer STs
924 compared to stringMLST as shown on S3 Table.
925 Previously, the scalability of ProkEvo was assessed by a comparative analysis
926 of its computational performance on Crane and OSG, using two datasets with
927 2,392 and 23,045 genomes each (10 X difference), and the standard mlst approach
928 for ST calling [21]. To further demonstrate the gain in computational runtime
929 obtained with the use of stringMLST within ProkEvo, the complete *S.*
930 Typhimurium dataset containing 23,045 genomes was run on OSG. While
931 ProkEvo with mlst finished all ST calls in 26 days and 6 hours when OSG was
932 used as a computational platform [21], ProkEvo with stringMLST completed the
933 task in 3 days and 6 hours. Altogether, stringMLST provides fine-tunable and

934 rapid alternative to mlst for scalable ST genotyping that is portable to be
935 implemented in any high-performance and high-throughput platform, with its use
936 being further facilitated by its implementation in ProkEvo.

937

938 # Discussion

939 The incorporation of WGS technology has advanced the study of bacterial
940 populations, enabling new strategies of inferring traits associated with populations
941 at different scales of resolution that may be contributing to important ecological
942 characteristics and/or epidemiological patterns [21][32][33][34][35][36][37][38].
943 In particular, the use of a hierarchical population structure analysis permits
944 inference into ancestral vs. recent evolutionary relationships and consequently,
945 the patterns of genomic and phenotypic diversification among genotypic units can
946 be studied within these evolutionary contexts [21][39][40][41][12][42][43].
947 ST-based classification is an integral part of the hierarchical genotyping
948 approach [21][27]. ST lineages are inferred from the patterns of allelic variation at
949 a relatively small (typically seven) set of conserved loci [8][9][44] and remain a
950 very useful context in which to study bacterial populations. However, as the field
951 transitions to WGS data, the ability to infer STs is bottlenecked by the
952 computationally intensive step of genome assembly. Thus, there is a great need
953 for algorithms that can efficiently infer STs from large WGS-based datasets.
954 [21][27][8][9][44]There are multiple tools available for MLST classification,
955 such as mlst [22], ARIBA **Error! Reference source not found.**, stringMLST

956 [17], MentaLiST **Error! Reference source not found.**, STing **Error! Reference**
957 **source not found.**. In general, the available tools can be categorized based on the
958 input data they use - some tools use raw Illumina paired-end sequence data, while
959 others use *de novo* assemblies [16]. Using raw sequence data for ST-based
960 classification has a tremendous advantage especially in pathogen surveillance,
961 since all the computationally demanding steps prior to the *de novo* assembly are
962 bypassed and the STs calls are made as the sequence reads are generated.
963 Specifically, mlst uses *de novo* genome assemblies as an input and performs
964 mapping in order to align sequences to pre-downloaded allelic files across all
965 target loci. ARIBA identifies AMR-associated genes, single nucleotide
966 polymorphisms, and ST calls using Illumina paired-end raw sequencing reads.
967 ARIBA clusters the raw reads by mapping them to genes, and then performs local
968 assembly within clusters to identify AMR genes and ST calls. On the other hand,
969 stringMLST and MentaLiST rely on kmer matching between raw sequence reads
970 and available ST schemes that allows for fast mapping and ST-based typing. In a
971 narrow setting with few bacterial species, both tools are shown to be accurate and
972 fast for standard MLST classification, while providing comparable accuracy with
973 MentaLiST albeit using less computational resources **Error! Reference source**
974 **not found.**. STing is the successor of stringMLST - it uses the same algorithmic
975 approach with additional computational applications for large MLST schemes
976 such as ribosomal MLST (rMLST) and core-genome MLST (cgMLST) **Error!**
977 **Reference source not found.**. All these tools have integrated ST schemes and/or
978 provide utilities for downloading the available PubMLST databases. There are a

979 few available comparisons of assembly-free tools for ST classification, and the
980 published studies are primarily focused on the computational resources used and
981 the percentage of correctly classified STs in relatively small data sets
982 [16][63][64]. When tools were tested with real outbreak datasets (*L.*
983 *monocytogenes*, *E. coli*, *C. jejuni*, *S. enterica*) comprising 85 samples,
984 stringMLST showed the fastest running time of 80.8 minutes and high accuracy in
985 ST calls (100%) [16]. While MentaLiST does not scale well when reads with high
986 coverage are used, it performs well on MLST schemes with up to a few thousand
987 genes and alleles, such as cgMLST (~3,000 genes) [63]. While most ST tools
988 perform satisfactorily, there are some relevant bottlenecks to be considered. For
989 example, some tools use out-of-date MLST databases that require manual
990 curation, and can directly affect the accuracy of ST calls, especially when mixed
991 and low coverage samples are used [16]. ST tools that are assembly and alignment
992 free, such as stringMLST, STing and MentaLiST, show quite a few advantages in
993 term of accuracy and efficiency that make them applicable for real-time molecular
994 epidemiology and surveillance. Thus, we chose stringMLST as a representative of
995 the kmer-based ST tools to perform a systems-based comparative analysis that
996 assess the computational and statistical efficacy of ST calls across divergent
997 pathogens in contrast to the legacy MLST approach.
998 As shown in our studies, the accuracy of stringMLST is affected by the species
999 being tested without any specific phylogenetic patterns. In particular, the choice
1000 of kmer length used directly impacts the proportion of ST miscalls across species,
1001 and in certain cases it may not be applied as designed even after parameter

1002 tunning. It is likely that the varying accuracy reflects the different levels of
1003 population structuring, patterns of genome diversification, degree of repetitive
1004 sequences, degree of horizontal gene transfer (HGT), and relative abundances of
1005 mobile elements such as prophages and insertion sequences, etc.
1006 [12][41][68][69][70][71][72][73]. A clear example in our study was *S. enterica*,
1007 for which the accuracy of stringMLST at different kmer lengths varied across
1008 ecologically distinct serovars that are known to have unique pan-genomic
1009 composition such as prophage content [12][71][74][75]. To the best of our
1010 knowledge, the currently available comparisons between ST tools have not
1011 considered any systematic approach for parameter tunning across phylogenetic
1012 divergent species known to vary in population structure [21][27][67].
1013 In evaluating genome-intrinsic and -extrinsic variables that could contribute to
1014 differences in accuracy between mlst and stringMLST, we found that species
1015 level variation in accuracy was mostly explained by the uniqueness of their
1016 genomic composition and number of contigs per genome. As genomic
1017 composition is an inheritable property of the bacterial species, subtype, and
1018 evolutionary history of specific populations, the association with algorithmic
1019 performance was somewhat expected [46][47]. However, the contribution of the
1020 number of contigs to performance implies that data from long-read sequencing
1021 platforms such as PacBio and Oxford Nanopore Technologies (ONT) may have a
1022 considerable effect on accuracy of programs such as stringMLST because these
1023 technologies produce reads with lower accuracy (~80-90%) that may inflate the
1024 number of false allelic calls at MLST loci, with the effect of artificially splitting

1025 major STs into multiple sub-populations [48]**Error! Reference source not**
1026 **found.Error! Reference source not found.**. Therefore, while more work is
1027 needed in this field, current studies using hybrid assembly approaches of both
1028 Illumina short reads and ONT long reads **Error! Reference source not found.**, as
1029 well as only polished ONT reads **Error! Reference source not found.** for
1030 performing ST-based classification may lead to promising, cost-effective results
1031 from these types of platforms. Hence, we expect that a combination of hybrid
1032 sequencing strategies with species and even subtype-specific tuning of programs
1033 such as stringMLST and STing, will facilitate real-time surveillance, prediction of
1034 STs and prediction of traits such as AMR that are found to associated with
1035 specific STs [14][49].

1036 While the kmer length of 35 is currently recommended as a default value of
1037 stringMLST, our systems-based approach demonstrated that for specific bacterial
1038 species it will result in increasing the frequency of ST miscalls which in turn may
1039 hinder epidemiological investigations. Across phylogenetic divergent pathogenic
1040 bacterial species, the optimal kmer length ranged from 20 to 140, regardless of
1041 their ancestral relationship or speciation pattern. The varying population structure,
1042 the pattern of genome diversification and architecture (e.g., impact of HGT), as
1043 well as sequence coverage may be some of the reasons underlying the observed
1044 statistics [12][41][68][69][70][71][72][73]. Although we hypothesize that longer
1045 sequence reads will help overcome this limitation, there is still a context-
1046 dependent consideration for parameter tunning and overall algorithmic
1047 implementation. Therefore, in the case of stringMLST, we suggest the following

1048 actionable strategies to maximize its utilization, including: i) developers to
1049 consider implementing a pre-step that heuristically searches for the optimal kmer
1050 length (minimizes ST miscalls) in dataset-dependent fashion (sampling from the
1051 testing data), perhaps even by comparing with the standard MLST program as
1052 positive controls; and/or ii) researchers to run wide range of kmer lengths on a
1053 subset of the dataset in order to select the optimal kmer length that minimizes the
1054 percentage of ST miscalls. Given the speed and scalability of stringMLST, using
1055 multiple kmer lengths is not likely to add much overhead to the analyses, and this
1056 provides an empirical statistical approach for kmer selection and optimization of
1057 ST classifications. With this data-driven fine-tunning of the kmer length,
1058 stringMLST is a powerful program that can be efficiently and effectively used in
1059 microbiological and epidemiological laboratories.

1060 We recently developed ProkEvo, a freely available scalable platform for
1061 performing hierarchical-based bacterial population genomics analyses [21].
1062 ProkEvo: 1) uses the Pegasus Workflow Management System to ensure
1063 reproducibility, scalability, and modularity; 2) uses high-performance and high-
1064 throughput computational platforms; 3) automates and scales multitude of
1065 computational analyses of a few to tens of thousands of bacterial genomes; 4) can
1066 run many thousands of analyses concurrently if the computational resources are
1067 available; 5) is easily modifiable and expandable platform that can incorporate
1068 additional algorithmic steps and custom scripts. The initial implementation of ST-
1069 based classifications through ProkEvo, as part of a hierarchical genotyping
1070 strategy to map and track populations, was done using the assembly-dependent

1071 MLST program [21][22]. Running mlst inside ProkEvo allows for parallelization
1072 of the genome assemblies (run per isolate or genome) which enhances scalability
1073 and facilitates the optimal use of computation resources. Theoretically, if there
1074 are *n* isolates and *n* cores available on the computational platform, ProkEvo can
1075 linearly utilize all resources and run all *n* independent tasks simultaneously.
1076 Typically, ST-based classifications are time consuming because the mapping
1077 process is run sequentially in a set of genomes instead of running them
1078 independently. Thus, using modular and distributed platforms such
1079 as ProkEvo for performing ST-based genotyping provides great benefit, especially
1080 if additional features such as other hierarchical genotypes and pan-genomic
1081 mapping tools are part of the same platform [21]. As part of this work, we
1082 modified ProkEvo to not only offer the standard assembly-dependent MLST
1083 mapping approach, but it now contains stringMLST, and our tests showed a
1084 significant speed-up in runtime for datasets ranging from a few hundreds to tens
1085 of thousands of genomes. Moreover, we additionally added STing to ProkEvo, as
1086 an efficient successor of stringMLST
1087 (https://github.com/npavlovikj/MLST_stringMLST_analyses/tree/main/Prokevo_
1088 STing). To use ProkEvo with stringMLST or STing, the researcher only needs to
1089 provide a list of SRA identifications and run the submit script without any
1090 advanced experience in high-performance or high-throughput computing.
1091 Depending on the configuration set, ProkEvo can use locally downloaded
1092 sequence data or download the data from NCBI directly. The Pegasus Workflow
1093 Managements System that is used by ProkEvo automatically handles the

1094 dependencies, as well as all the intermediate and final files. Thus, using platforms
1095 such as ProkEvo with fast tool for hierarchical genotyping, such as stringMLST,
1096 allows for robust and efficient population-based genomics analyses that facilitate:
1097 i) mapping and tracking of variants or lineages for epidemiological inquiries; ii)
1098 population structure analysis; and iii) ecological trait prediction using pan-
1099 genomic mapping to specific genotypes.
1100

1101 Conclusion

1102 In conclusion, stringMLST largely proved to be an accurate, rapid, and scalable
1103 tool for ST-based classifications that could be deployed in microbiological
1104 laboratories and epidemiological agencies. However, our work clearly illustrates
1105 the need to optimize stringMLST across phylogenetic divergent species and
1106 populations of bacterial pathogens. Based on our results, we propose that the kmer
1107 length should be optimized in two ways on a case-by-case basis: 1) intrinsically
1108 by implementing a pre-step inside the algorithm to sample from the target data
1109 and select the optimal kmer length; or 2) by the user through a heuristic data
1110 mining approach to select the optimal kmer length prior to finalizing the ST calls.
1111 Also, by assessing genome-intrinsic and -extrinsic factors that could affect the
1112 stringMLST performance, our work suggests that longer sequence reads have the
1113 potential to improve its accuracy for specific bacterial species. Furthermore, the
1114 integration of stringMLST into ProkEvo allows users to take advantage of other
1115 hierarchical genotyping strategies, including pan-genomic mapping, which
1116 reproducibly facilitates ecological and epidemiological inquiries at scale.

Ultimately, this work emphasizes the importance of developing robust algorithmic tools for mining WGS data that can have direct implications for mapping and tracking of bacterial populations.

Acknowledgements

This work was completed by utilizing the Holland Computing Center of the University of Nebraska, which receives support from the Nebraska Research Initiative, and using resources provided by the Open Science Grid, which is supported by the National Science Foundation and the U.S. Department of Energy's Office of Science. This research used the Pegasus Workflow Management Software funded by the National Science Foundation under grant #1664162. This publication made use of the PubMLST website (https://pubmlst.org/) developed by Keith Jolley (Jolley & Maiden 2010, BMC Bioinformatics, 11:595) and sited at the University of Oxford. The development of that website was funded by the Wellcome Trust. We would like to greatly thank Mats Rynge for his extensive assistance and valuable suggestions while setting up and running ProkEvo on the Open Science Grid. We also thank Dr. Derek Weitzel and Karan Vahi for their technical support.

1136 # References

1137 [1] Bedford J, Farrar J, Ihekweazu C, Kang G, Koopmans M, Nkengasong J.
1138 A new twenty-first century science for effective epidemic response.
1139 Nature. 2019 Nov;575(7781):130-6.
1140 [2] Lewnard JA, Reingold AL. Emerging challenges and opportunities in
1141 infectious disease epidemiology. American journal of epidemiology. 2019
1142 May 1;188(5):873-82.
1143 [3] Armstrong GL, MacCannell DR, Taylor J, Carleton HA, Neuhaus EB,
1144 Bradbury RS, Posey JE, Gwinn M. Pathogen genomics in public health.
1145 New England Journal of Medicine. 2019 Dec 26;381(26):2569-80.
1146 [4] Achtman M. How old are bacterial pathogens?. Proceedings of the Royal
1147 Society B: Biological Sciences. 2016 Aug 17;283(1836):20160990.
1148 [5] Selander RK, Musser JM, Caugant DA, Gilmour MN, Whittam TS.
1149 Population genetics of pathogenic bacteria. Microbial pathogenesis. 1987
1150 Jul 1;3(1):1-7.
1151 [6] Shapiro BJ. How clonal are bacteria over time?. Current opinion in
1152 microbiology. 2016 Jun 1;31:116-23.
1153 [7] Smith JM, Smith NH, O'Rourke M, Spratt BG. How clonal are bacteria?.
1154 Proceedings of the National Academy of Sciences. 1993 May
1155 15;90(10):4384-8.
1156 [8] Maiden MC, Bygraves JA, Feil E, Morelli G, Russell JE, Urwin R, Zhang
1157 Q, Zhou J, Zurth K, Caugant DA, Feavers IM. Multilocus sequence
1158 typing: a portable approach to the identification of clones within

1159 populations of pathogenic microorganisms. Proceedings of the National
1160 Academy of Sciences. 1998 Mar 17;95(6):3140-5.
1161 [9] Maiden MC. Multilocus sequence typing of bacteria. Annu. Rev.
1162 Microbiol.. 2006 Oct 13;60:561-88.
1163 [10] Feil EJ, Li BC, Aanensen DM, Hanage WP, Spratt BG. eBURST:
1164 inferring patterns of evolutionary descent among clusters of related
1165 bacterial genotypes from multilocus sequence typing data. Journal of
1166 bacteriology. 2004 Mar 1;186(5):1518-30.
1167 [11] Croucher NJ, Harris SR, Grad YH, Hanage WP. Bacterial genomes in
1168 epidemiology—present and future. Philosophical Transactions of the
1169 Royal Society B: Biological Sciences. 2013 Mar 19;368(1614):20120202.
1170 [12] Alikhan NF, Zhou Z, Sergeant MJ, Achtman M. A genomic overview of
1171 the population structure of Salmonella. PLoS genetics. 2018 Apr
1172 5;14(4):e1007261.
1173 [13] Achtman M, Wain J, Weill FX, Nair S, Zhou Z, Sangal V, Krauland MG,
1174 Hale JL, Harbottle H, Uesbeck A, Dougan G. Multilocus sequence typing
1175 as a replacement for serotyping in Salmonella enterica. PLoS Pathog. 2012
1176 Jun 21;8(6):e1002776.
1177 [14] Břinda K, Callendrello A, Ma KC, MacFadden DR, Charalampous T, Lee
1178 RS, Cowley L, Wadsworth CB, Grad YH, Kucherov G, O'Grady J. Rapid
1179 inference of antibiotic resistance and susceptibility by genomic neighbour
1180 typing. Nature microbiology. 2020 Mar;5(3):455-64.

1181 [15] MacFadden DR, Coburn B, Břinda K, Corbeil A, Daneman N, Fisman D,
1182 Lee RS, Lipsitch M, McGeer A, Melano RG, Mubareka S. Using genetic
1183 distance from archived samples for the prediction of antibiotic resistance
1184 in Escherichia coli. Antimicrobial agents and chemotherapy. 2020 Mar
1185 9;64(5):e02417-19.
1186 [16] Page AJ, Alikhan NF, Carleton HA, Seemann T, Keane JA, Katz LS.
1187 Comparison of classical multi-locus sequence typing software for next-
1188 generation sequencing data. Microbial genomics. 2017 Aug;3(8).
1189 [17] Gupta A, Jordan IK, Rishishwar L. stringMLST: a fast kmer based tool for
1190 multilocus sequence typing. Bioinformatics. 2017 Jan 1;33(1):119-21.
1191 [18] Abebe E, Gugsa G, Ahmed M. Review on major food-borne zoonotic
1192 bacterial pathogens. Journal of tropical medicine. 2020 Jun 29;2020.
1193 [19] Centers for Disease Control and Prevention. Foodborne germs and
1194 illnesses. Centers for Disease Control and Prevention. https://www. cdc.
1195 gov/foodsafety/foodborne-germs. html. 2016.
1196 [20] Centers for Disease Control and Prevention. Individual Salmonella
1197 serotypes reports. Centers for Disease Control and Prevention.
1198 https://www.cdc.gov/salmonella/reportspubs/salmonella-atlas/serotype-
1199 reports.html. 2020.
1200 [21] Pavlovikj N, Gomes-Neto JC, Deogun JS, Benson AK. ProkEvo: an
1201 automated, reproducible, and scalable framework for high-throughput
1202 bacterial population genomics analyses. PeerJ. 2021 May 21;9:e11376.
1203 [22] Seemann T, mlst Github https://github.com/tseemann/mlst.

1204 [23] HCC. 2008. Holland computing center | Nebraska.
1205 [24] Pordes R, Petravick D, Kramer B, Olson D, Livny M, Roy A, Avery P,
1206 Blackburn K, Wenaus T, Würthwein F+6 more. 2007. The open science
1207 grid. Journal of Physics: Conference Series 78:12057
1208 [25] Sfiligoi I, Bradley DC, Holzman B, Mhashilkar P, Padhi S, Wurthwein F.
1209 2009. The pilot way to grid resources using glideinWMS.
1210 [26] Anaconda. 2012. Anaconda | The World's Most Popular Data Science
1211 Platform.
1212 [27] Zhou Z, Alikhan NF, Mohamed K, Achtman M, Agama Study Group. The
1213 user's guide to comparative genomics with EnteroBase. Three case
1214 studies: micro-clades within Salmonella enterica serovar Agama, ancient
1215 and modern populations of Yersinia pestis, and core genomic diversity of
1216 all Escherichia. Biorxiv. 2019 Jan 1:613554.
1217 [28] Feil EJ, Cooper JE, Grundmann H, Robinson DA, Enright MC, Berendt T,
1218 Peacock SJ, Smith JM, Murphy M, Spratt BG, Moore CE. How clonal is
1219 Staphylococcus aureus?. Journal of bacteriology. 2003 Jun
1220 1;185(11):3307-16.
1221 [29] Oksanen J, Blanchet FG, Kindt R, Legendre P, Minchin PR, O'hara RB,
1222 Simpson GL, Solymos P, Stevens MH, Wagner H, Oksanen MJ. Package
1223 'vegan'. Community ecology package, version. 2013 Dec 12;2(9):1-295.
1224 [30] Jolley KA, Bray JE, Maiden MC. Open-access bacterial population
1225 genomics: BIGSdb software, the PubMLST. org website and their
1226 applications. Wellcome open research. 2018;3.

1227 [31] Deelman E, Singh G, Su MH, Blythe J, Gil Y, Kesselman C, Mehta G,
1228 Vahi K, Berriman GB, Good J, Laity A. Pegasus: A framework for
1229 mapping complex scientific workflows onto distributed systems. Scientific
1230 Programming. 2005 Jan 1;13(3):219-37.
1231 [32] Burnett E, Ishida M, De Janon S, Naushad S, Duceppe MO, Gao R,
1232 Jardim A, Chen JC, Tagg KA, Ogunremi D, Vinueza-Burgos C. Whole-
1233 genome sequencing reveals the presence of the blactx-m-65 gene in
1234 extended-spectrum β-lactamase-producing and multi-drug-resistant clones
1235 of salmonella serovar infantis isolated from broiler chicken environments
1236 in the Galapagos Islands. Antibiotics. 2021 Mar;10(3):267.
1237 [33] Cooper AL, Low AJ, Koziol AG, Thomas MC, Leclair D, Tamber S,
1238 Wong A, Blais BW, Carrillo CD. Systematic evaluation of whole genome
1239 sequence-based predictions of Salmonella serotype and antimicrobial
1240 resistance. Frontiers in microbiology. 2020 Apr 3;11:549.
1241 [34] Dallman TJ, Byrne L, Ashton PM, Cowley LA, Perry NT, Adak G,
1242 Petrovska L, Ellis RJ, Elson R, Underwood A, Green J. Whole-genome
1243 sequencing for national surveillance of Shiga toxin–producing Escherichia
1244 coli O157. Clinical Infectious Diseases. 2015 Aug 1;61(3):305-12.
1245 [35] Pightling AW, Pettengill JB, Luo Y, Baugher JD, Rand H, Strain E.
1246 Interpreting whole-genome sequence analyses of foodborne bacteria for
1247 regulatory applications and outbreak investigations. Frontiers in
1248 microbiology. 2018 Jul 10;9:1482.

1249 [36] Sheppard SK, Jolley KA, Maiden MC. A gene-by-gene approach to
1250 bacterial population genomics: whole genome MLST of Campylobacter.
1251 Genes. 2012 Jun;3(2):261-77.
1252 [37] Sheppard SK, Didelot X, Meric G, Torralbo A, Jolley KA, Kelly DJ,
1253 Bentley SD, Maiden MC, Parkhill J, Falush D. Genome-wide association
1254 study identifies vitamin B5 biosynthesis as a host specificity factor in
1255 Campylobacter. Proceedings of the national academy of sciences. 2013 Jul
1256 16;110(29):11923-7.
1257 [38] Alba P, Leekitcharoenphon P, Carfora V, Amoruso R, Cordaro G, Di
1258 Matteo P, Ianzano A, Iurescia M, Diaconu EL, Pedersen SK, Guerra B.
1259 Molecular epidemiology of Salmonella Infantis in Europe: insights into
1260 the success of the bacterial host and its parasitic pESI-like megaplasmid.
1261 Microbial genomics. 2020 May;6(5).
1262 [39] Cheng L, Connor TR, Sirén J, Aanensen DM, Corander J. Hierarchical
1263 and spatially explicit clustering of DNA sequences with BAPS software.
1264 Molecular biology and evolution. 2013 Feb 13;30(5):1224-8.
1265 [40] Tonkin-Hill G, Lees JA, Bentley SD, Frost SD, Corander J. Fast
1266 hierarchical Bayesian analysis of population structure. Nucleic Acids
1267 Research. 2019 Jun 20;47(11):5539-49.
1268 [41] Didelot X, Bowden R, Street T, Golubchik T, Spencer C, McVean G,
1269 Sangal V, Anjum MF, Achtman M, Falush D, Donnelly P. Recombination
1270 and population structure in Salmonella enterica. PLoS genetics. 2011 Jul
1271 28;7(7):e1002191.

1272 [42] Gymoese P, Kiil K, Torpdahl M, Østerlund MT, Sørensen G, Olsen JE,
1273 Nielsen EM, Litrup E. WGS based study of the population structure of
1274 Salmonella enterica serovar Infantis. BMC genomics. 2019 Dec;20(1):1-1.
1275 [43] Liao J, Orsi RH, Carroll LM, Wiedmann M. Comparative genomics
1276 reveals different population structures associated with host and geographic
1277 origin in antimicrobial-resistant Salmonella enterica. Environmental
1278 microbiology. 2020 Jul;22(7):2811-28.
1279 [44] Maiden MC, Van Rensburg MJ, Bray JE, Earle SG, Ford SA, Jolley KA,
1280 McCarthy ND. MLST revisited: the gene-by-gene approach to bacterial
1281 genomics. Nature Reviews Microbiology. 2013 Oct;11(10):728-36.
1282 [45] Spratt BG, Hanage WP, Li B, Aanensen DM, Feil EJ. Displaying the
1283 relatedness among isolates of bacterial species–the eBURST approach.
1284 FEMS microbiology letters. 2004 Dec 1;241(2):129-34.
1285 [46] Bobay LM, Ochman H. Impact of recombination on the base composition
1286 of bacteria and archaea. Molecular biology and evolution. 2017 Oct
1287 1;34(10):2627-36.
1288 [47] Raghavan R, Kelkar YD, Ochman H. A selective force favoring increased
1289 G+ C content in bacterial genes. Proceedings of the National Academy of
1290 Sciences. 2012 Sep 4;109(36):14504-7.
1291 [48] De Maio N, Shaw LP, Hubbard A, George S, Sanderson ND, Swann J,
1292 Wick R, AbuOun M, Stubberfield E, Hoosdally SJ, Crook DW.
1293 Comparison of long-read sequencing technologies in the hybrid assembly
1294 of complex bacterial genomes. Microbial genomics. 2019 Sep;5(9).

1295 [49] Page AJ, Keane JA. Rapid multi-locus sequence typing direct from
1296 uncorrected long reads using Krocus. PeerJ. 2018 Jul 31;6:e5233.
1297 [50] Bolger AM, Lohse M, Usadel B. Trimmomatic: a flexible trimmer for
1298 Illumina sequence data. Bioinformatics. 2014 Aug 1;30(15):2114-20.
1299 [51] Andrews S. FASTQC: a quality control tool for high throughput sequence
1300 data. Available at
1301 http://www.bioinformatics.babraham.ac.uk/projects/fastqc/. 2010.
1302 [52] Bankevich A, Nurk S, Antipov D, Gurevich AA, Dvorkin M, Kulikov AS,
1303 Lesin VM, Nikolenko SI, Pham S, Prjibelski AD, Pyshkin AV. SPAdes: a
1304 new genome assembly algorithm and its applications to single-cell
1305 sequencing. Journal of computational biology. 2012 May 1;19(5):455-77.
1306 [53] Gurevich A, Saveliev V, Vyahhi N, Tesler G. QUAST: quality assessment
1307 tool for genome assemblies. Bioinformatics. 2013 Apr 15;29(8):1072-5.
1308 [54] Treangen TJ, Sommer DD, Angly FE, Koren S, Pop M. Next generation
1309 sequence assembly with AMOS. Current Protocols in Bioinformatics.
1310 2011 Mar;33(1):11-8.
1311 [55] Rice P, Longden I, Bleasby A. EMBOSS: the European molecular biology
1312 open software suite. Trends in genetics. 2000 Jun 1;16(6):276-7.
1313 [56] Rizk G, Lavenier D, Chikhi R. DSK: k-mer counting with very low
1314 memory usage. Bioinformatics. 2013 Mar 1;29(5):652-3.
1315 [57] Gomes-Neto JC, Pavlovikj N, Cano C, Abdalhamid B, Al-Ghalith GA,
1316 Loy JD, Knights D, Iwen PC, Chaves BD, Benson AK. Heuristic and
1317 hierarchical-based population mining of Salmonella enterica lineage I pan-

1318 genomes as a platform to enhance food safety. Front. Sustain. Food Syst.
1319 5:725791. doi: 10.3389/fsufs.2021.725791.
1320 [58] Valieris R. Parallel-fastq-dump. GitHub. Available at
1321 https://github.com/rvalieris/parallelfastq-dump. 2020.
1322 [59] Rang FJ, Kloosterman WP, de Ridder J. From squiggle to basepair:
1323 computational approaches for improving nanopore sequencing read
1324 accuracy. Genome biology. 2018 Dec;19(1):1-1.
1325 [60] Rhoads A, Au KF. PacBio sequencing and its applications. Genomics,
1326 proteomics & bioinformatics. 2015 Oct 1;13(5):278-89.
1327 [61] Chen Z, Erickson DL, Meng J. Benchmarking hybrid assembly
1328 approaches for genomic analyses of bacterial pathogens using Illumina
1329 and Oxford Nanopore sequencing. BMC genomics. 2020 Dec;21(1):1-21.
1330 [62] Liou CH, Wu HC, Liao YC, Lauderdale TL, Huang IW, Chen FJ.
1331 nanoMLST: accurate multilocus sequence typing using Oxford Nanopore
1332 Technologies MinION with a dual-barcode approach to multiplex large
1333 numbers of samples. Microbial genomics. 2020 Mar;6(3).
1334 [63] Feijao P, Yao HT, Fornika D, Gardy J, Hsiao W, Chauve C, Chindelevitch
1335 L. MentaLiST–A fast MLST caller for large MLST schemes. Microbial
1336 genomics. 2018 Feb;4(2).
1337 [64] Espitia-Navarro HF, Chande AT, Nagar SD, Smith H, Jordan IK,
1338 Rishishwar L. STing: accurate and ultrafast genomic profiling with exact
1339 sequence matches. Nucleic acids research. 2020 Aug 20;48(14):7681-9.

1340 [65] Hunt M, Mather AE, Sánchez-Busó L, Page AJ, Parkhill J, Keane JA,
1341 Harris SR. ARIBA: rapid antimicrobial resistance genotyping directly
1342 from sequencing reads. Microbial genomics. 2017 Oct;3(10).
1343 [66] Page AJ, Cummins CA, Hunt M, Wong VK, Reuter S, Holden MT,
1344 Fookes M, Falush D, Keane JA, Parkhill J. Roary: rapid large-scale
1345 prokaryote pan genome analysis. Bioinformatics. 2015 Nov
1346 15;31(22):3691-3.
1347 [67] Zhou Z, Alikhan NF, Mohamed K, Fan Y, Achtman M, Brown D,
1348 Chattaway M, Dallman T, Delahay R, Kornschober C, Pietzka A. The
1349 EnteroBase user's guide, with case studies on Salmonella transmissions,
1350 Yersinia pestis phylogeny, and Escherichia core genomic diversity.
1351 Genome research. 2020 Jan 1;30(1):138-52.
1352 [68] Sheppard SK, Colles FM, McCARTHY ND, Strachan NJ, Ogden ID,
1353 Forbes KJ, Dallas JF, Maiden MC. Niche segregation and genetic
1354 structure of Campylobacter jejuni populations from wild and agricultural
1355 host species. Molecular ecology. 2011 Aug;20(16):3484-90.
1356 [69] Sheppard SK, Cheng L, Méric G, De Haan CP, Llarena AK, Marttinen P,
1357 Vidal A, Ridley A, Clifton-Hadley F, Connor TR, Strachan NJ. Cryptic
1358 ecology among host generalist Campylobacter jejuni in domestic animals.
1359 Molecular ecology. 2014 May;23(10):2442-51.
1360 [70] Yahara K, Didelot X, Ansari MA, Sheppard SK, Falush D. Efficient
1361 inference of recombination hot regions in bacterial genomes. Molecular
1362 biology and evolution. 2014 Jun 1;31(6):1593-605.

1363 [71] Mottawea W, Duceppe MO, Dupras AA, Usongo V, Jeukens J, Freschi L,
1364 Emond-Rheault JG, Hamel J, Kukavica-Ibrulj I, Boyle B, Gill A.
1365 Salmonella enterica prophage sequence profiles reflect genome diversity
1366 and can be used for high discrimination subtyping. Frontiers in
1367 microbiology. 2018 May 4;9:836.
1368 [72] den Bakker HC, Desjardins CA, Griggs AD, Peters JE, Zeng Q, Young
1369 SK, Kodira CD, Yandava C, Hepburn TA, Haas BJ, Birren BW.
1370 Evolutionary dynamics of the accessory genome of Listeria
1371 monocytogenes. PLoS One. 2013 Jun 25;8(6):e67511.
1372 [73] Castillo-Ramírez S, Corander J, Marttinen P, Aldeljawi M, Hanage WP,
1373 Westh H, Boye K, Gulay Z, Bentley SD, Parkhill J, Holden MT.
1374 Phylogeographic variation in recombination rates within a global clone of
1375 methicillin-resistant Staphylococcus aureus. Genome biology. 2012
1376 Dec;13(12):1-3.
1377 [74] Laing CR, Whiteside MD, Gannon VP. Pan-genome analyses of the
1378 species Salmonella enterica, and identification of genomic markers
1379 predictive for species, subspecies, and serovar. Frontiers in microbiology.
1380 2017 Jul 31;8:1345.
1381 [75] Ferrari RG, Rosario DK, Cunha-Neto A, Mano SB, Figueiredo EE, Conte-
1382 Junior CA. Worldwide epidemiology of Salmonella serovars in animal-
1383 based foods: a meta-analysis. Applied and environmental microbiology.
1384 2019 Jul 1;85(14):e00591-19.

1385 [76] Price MN, Dehal PS, Arkin AP. FastTree: computing large minimum
1386 evolution trees with profiles instead of a distance matrix. Molecular
1387 biology and evolution. 2009 Jul 1;26(7):1641-50.
1388 [77] Letunic I, Bork P. Interactive Tree Of Life (iTOL) v5: an online tool for
1389 phylogenetic tree display and annotation. Nucleic acids research. 2021 Jul
1390 2;49(W1):W293-6.
1391 [78] Seemann T. Prokka: rapid prokaryotic genome annotation. Bioinformatics.
1392 2014 Jul 15;30(14):2068-9.
1393 [79] Selander RK, Caugant DA, Ochman H, Musser JM, Gilmour MN,
1394 Whittam TS. Methods of multilocus enzyme electrophoresis for bacterial
1395 population genetics and systematics. Applied and environmental
1396 microbiology. 1986 May;51(5):873-84.
1397 [80] Maslow JN, Whittam TS, Gilks CF, Wilson RA, Mulligan ME, Adams
1398 KS, Arbeit RD. Clonal relationships among bloodstream isolates of
1399 Escherichia coli. Infection and Immunity. 1995 Jul;63(7):2409-17.
1400 [81] Turner KM, Hanage WP, Fraser C, Connor TR, Spratt BG. Assessing the
1401 reliability of eBURST using simulated populations with known ancestry.
1402 BMC microbiology. 2007 Dec;7(1):1-4.
1403 [82] Chang Q, Abuelaish I, Biber A, Jaber H, Callendrello A, Andam CP,
1404 Regev-Yochay G, Hanage WP, PICR Study Group. Genomic
1405 epidemiology of meticillin-resistant Staphylococcus aureus ST22
1406 widespread in communities of the Gaza Strip, 2009. Eurosurveillance.
1407 2018 Aug 23;23(34):1700592.

1408 [83] Chang Q, Stevenson AE, Croucher NJ, Lee GM, Pelton SI, Lipsitch M,
1409 Finkelstein JA, Hanage WP. Stability of the pneumococcal population
1410 structure in Massachusetts as PCV13 was introduced. BMC infectious
1411 diseases. 2015 Dec;15(1):1-7.

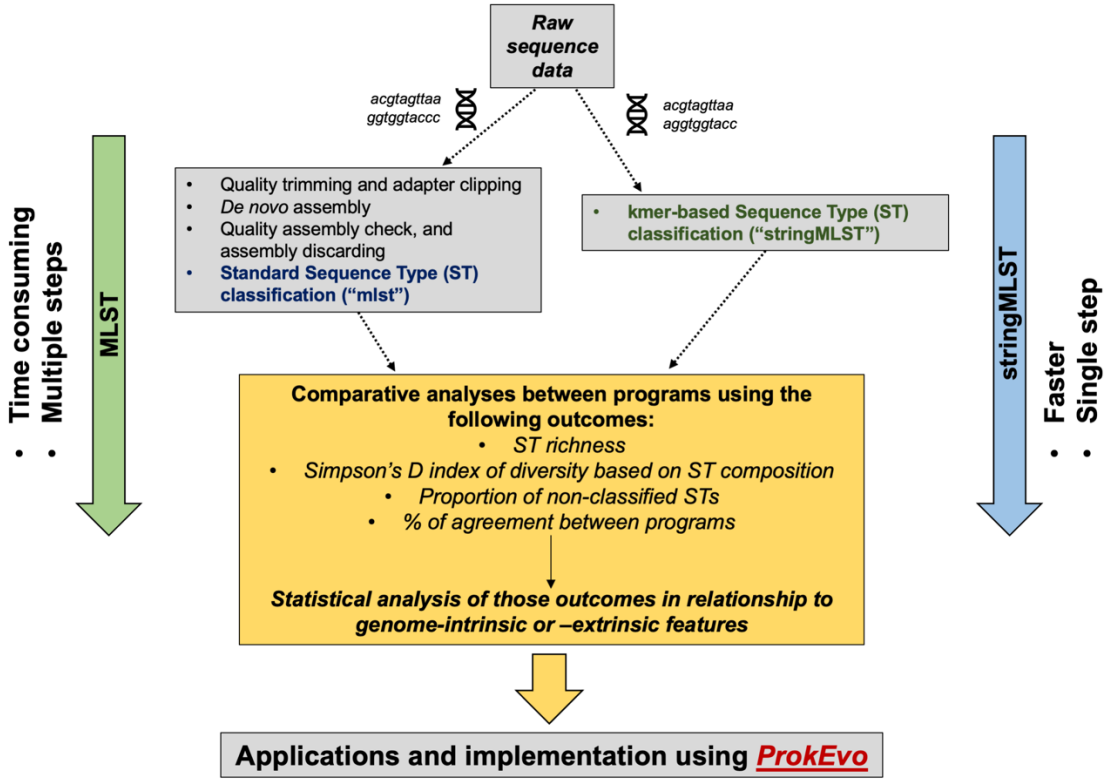

Fig 1. Computational workflow describing the analytical steps for a comparative analysis of two algorithms used for ST-based classification.
From top-down, the first step (narrow-scope) of the analytical approach entailed the acquisition and processing of Illumina paired-end raw reads from four distinct pathogens (*C. jejuni*, *L. monocytogenes*, *S. enterica* and *S. aureus*), through an assembly-dependent (mlst) or assembly-free (stringMLST) approach for ST-based classification. Next, a set of comparative analyses encompassing measuring the computational performance, statistical metrics, and modeling were used to assess the accuracy and efficiency of mlst vs. stringMLST. Additionally, the contribution of genome-intrinsic and –extrinsic variables were used to identify explanatory factors that could impact the algorithmic efficiency across phylogenetic divergent species. Upon identification of inter-species differences in the performance of stringMLST, a wide-scope analysis was done to assess its accuracy across an array of fifteen phylogenetic divergent pathogenic species of bacteria with Public Health relevance. Ultimately, stringMLST was added to the computational platform ProkEvo to facilitate ST-based classification at scale, as part of a hierarchical-based approach for population genomic analyses.

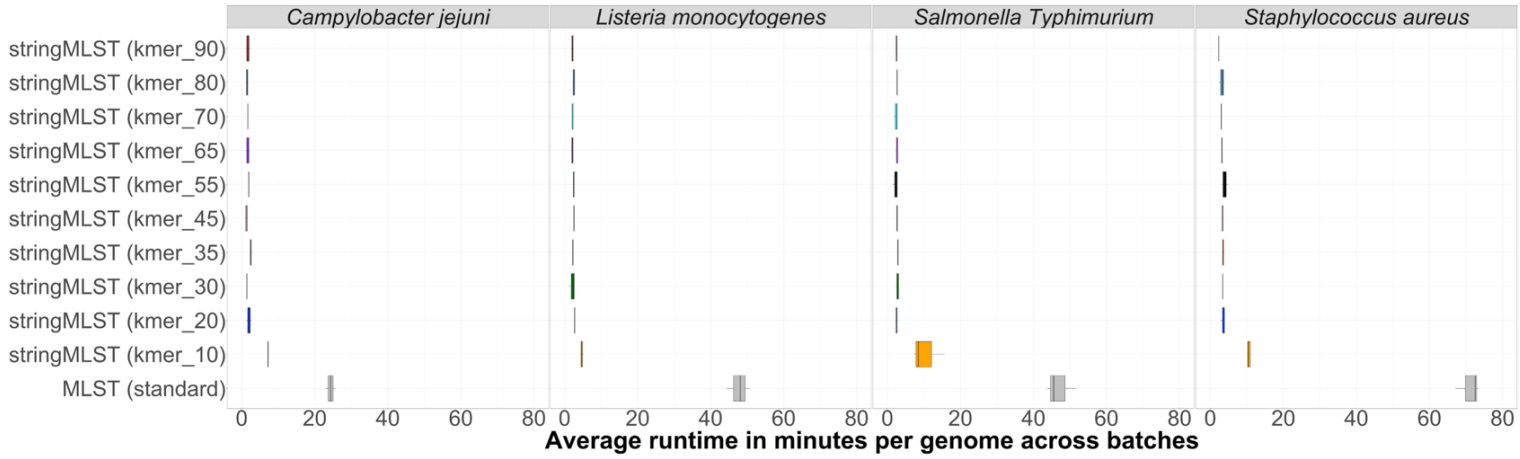

Fig 2. Box-and-whiskers plot showing the comparison of the average runtime per genome per batch (in minutes) needed by mlst and stringMLST for ST classification of genomes across four distinct bacterial species.
In order to compare the average runtime used by mlst and stringMLST with different kmer lenghts, we chose four different datasets, including four phylogenetic divergent bacterial pathogenic species: *C. jejuni*, *L. monocytogenes*, one major serovar of *S. enterica* (*S.* Typhimurium) and *S. aureus* - using 600 randomly selected genomes for each species. These 600 genomes were randomly split into three batches with 200 genomes each. We then ran mlst with all required steps, such as quality trimming and adapter clipping, *de novo* assembly and assembly discarding, on each batch and dataset. Separately, we ran stringMLST with a range of 10 different kmer values (10, 20, 30, 35, 45, 55, 65, 70, 80, 90) on each dataset, including the default length of 35 (y-axis). For each organism, the runtime was calculated as an average of 200 genomes per batch - since there were three batches, three datapoints were used to depict the distribution of runtime in minutes (x-axis).

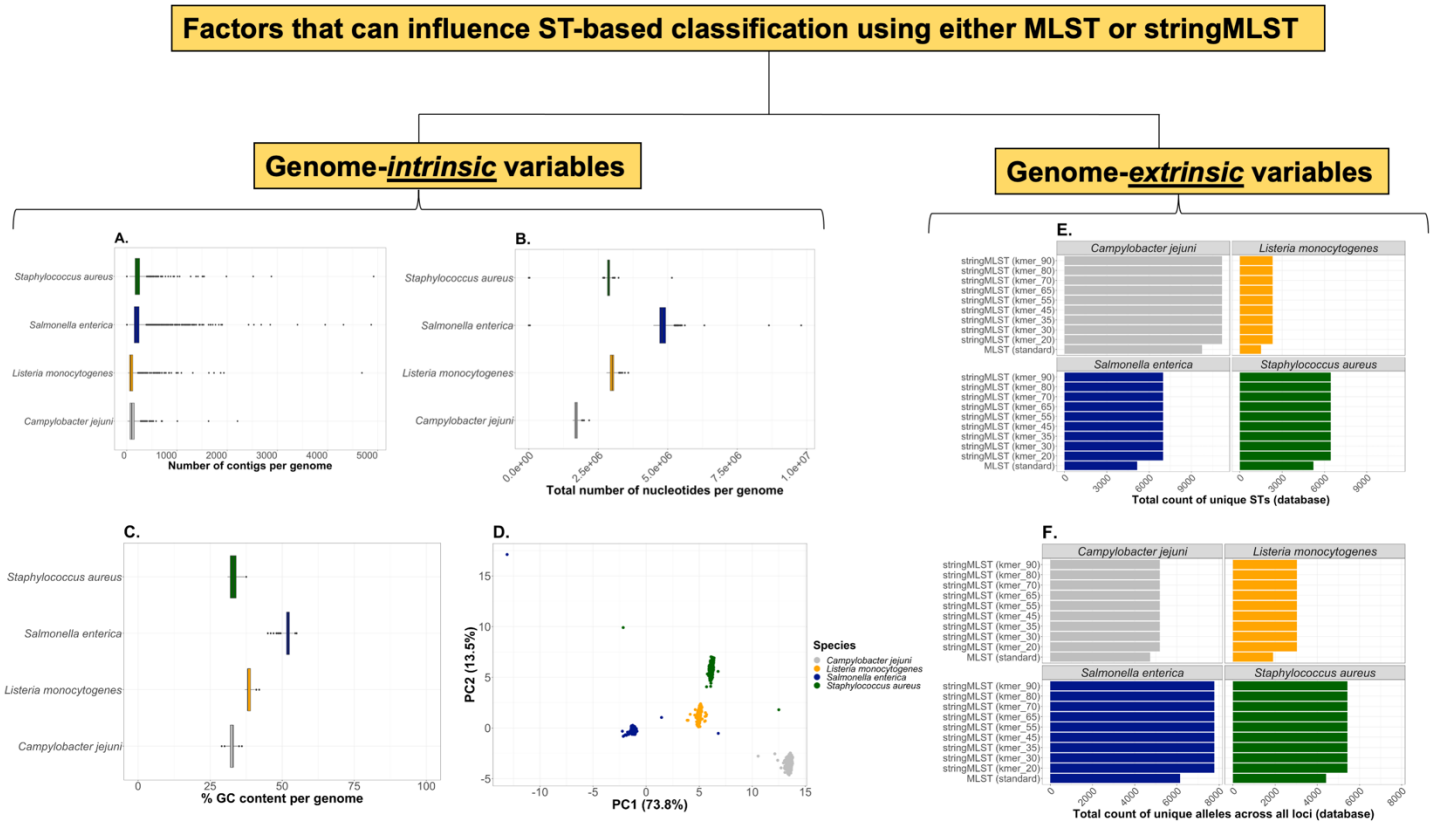

Fig 3. Genome-intrinsic and –extrinsic variables that can impact the accuracy of ST-based classification using either mlst (MLST-based genotyping) or stringMLST algorithmic approaches.
Box-and-whiskers plot showing genome-intrinsic variables, varying in distribution according to the bacterial species (A-C as y-axis), that may affect ST-based classification, include: (A) Number of contigs per genome (x-axis); (B) Total number of nucleotides per genome (x-axis); (C) GC% content per genome (x-axis); and (D) Dinucleotide composition of genomes. (D) Inter-species PCA using the relative frequency of all pairs of dinucleotides (16 pairs) present in the genome as input data. Only two PCs are shown, and the percentage of variance explained by either PC is depicted in parenthesis. Bar plots showing genome-extrinsic variables that may influence the performance of mlst vs. stringMLST across species include but are not limited to: (E) Total count of unique STs per database (ST richness in the database used for mapping of raw reads or assemblies) (x-axis); and (F) Total count of unique alleles across all seven loci used for ST classification (x-axis). Specifically, the differences in ST richness and allelic composition in the databases reflect difference between mlst vs. stringMLST, and were not impacted by the kmer length (E-F, y-axis).

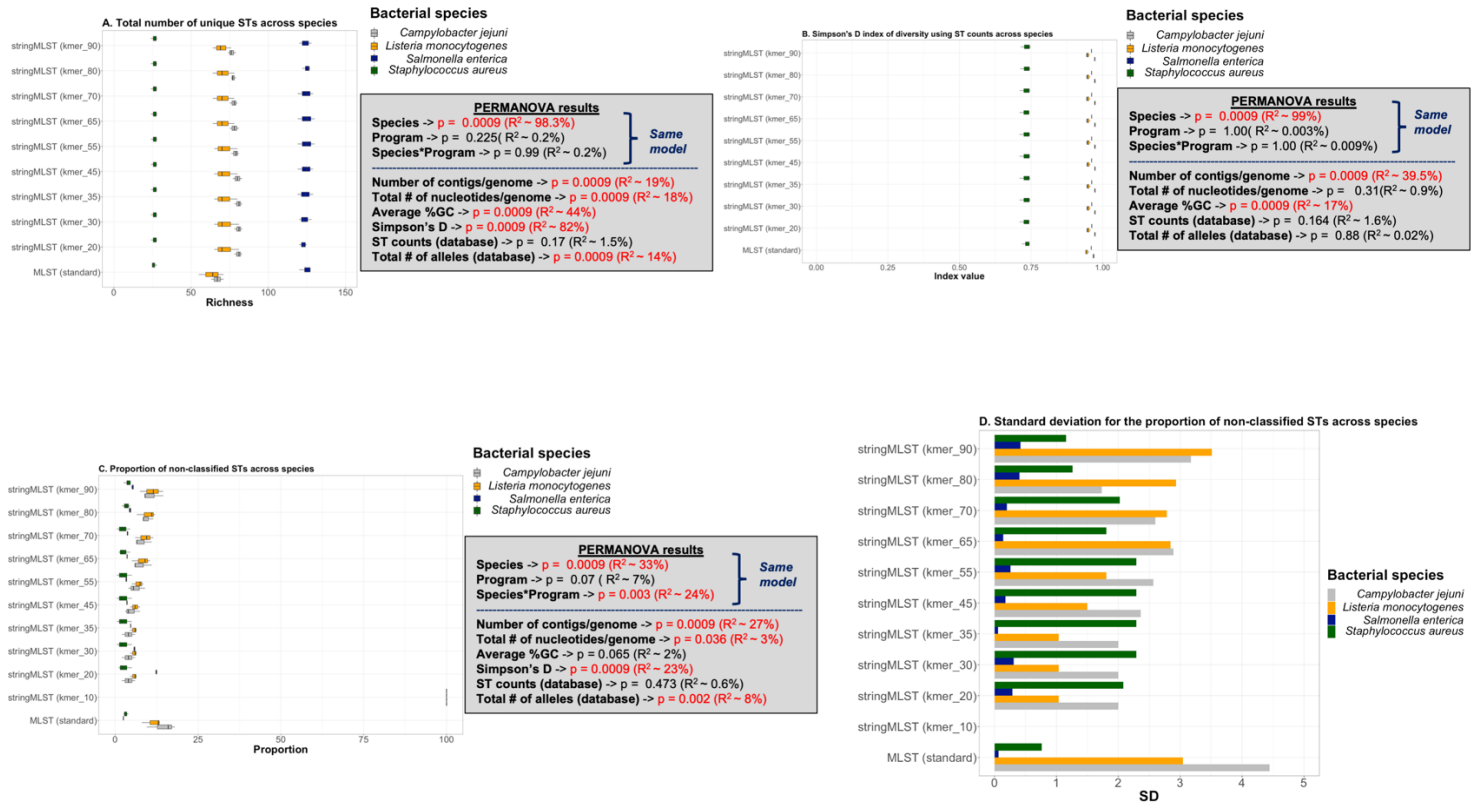

Fig 4. Statistical analysis of ST-based classification outcomes for comparison between mlst and stringMLST performance across bacterial species.
(A-C) Box-and-whiskers plots A-C demonstrate the relationship between ST richness (x-axis), Simpson's index of diversity (1 – *D*) based on ST composition (x-axis), or the proportion of non-classified STs (x-axis) across bacterial species (color-coded differently) and programs (y-axis), respectively. Along with plots A-C are depicted all PERMANOVA results including *p*-values ($p < 0.05$) and the univariate or synergistic contribution of factors measured by *R*-squared. PERMANOVA modeling was done in two specific ways: 1) A model including species, program, and their interaction, considering that those were the main variables of interest; and 2) All other results were calculated using univariate models and included modeling using genome-intrinsic (number of contigs per genome, total number of nucleotides per genome, and average GC% content) and –extrinsic (Simpson's D index of diversity, ST and allelic counts per database) variables. (D) Bar plot depicting the distribution of the standard deviation (SD, y-axis) for the proportion of non-classified STs based on species (color-coded differently) and programs (y-axis).

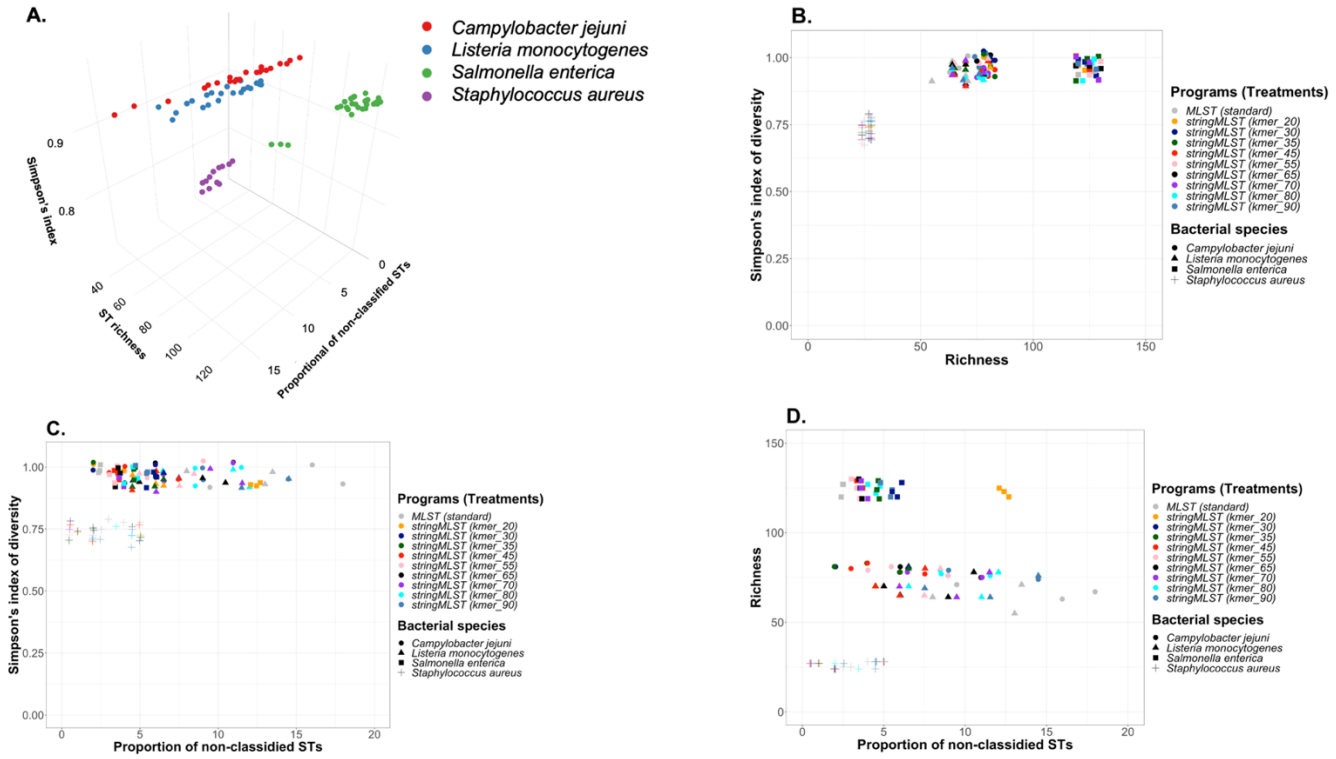

Fig 5. Multi-dimensional analysis of ST-based classification outcomes across different species using mlst vs. stringMLST.
(A) Tri-dimensional scatter plot demonstrating species grouping based on the outcomes calculated using the ST classification across programs, including: 1) Simpson's index of diversity (1 – *D,* Simpson's index); 2) ST richness; and 3) proportion of non-classified STs. (B-D) Biplots demonstrating groupings formed across species and programs based on the same outcomes. Scatter plot B depicts groupings produced based on the relationship between Simpson's index of diversity vs. ST richness (Richness); whereas scatter plot C shows the relationship between the Simpson's index of diversity and the proportion of non-classified STs; and lastly, scatter plot D depicts the relationship between ST richness (Richness) and the proportion of non-classified STs.

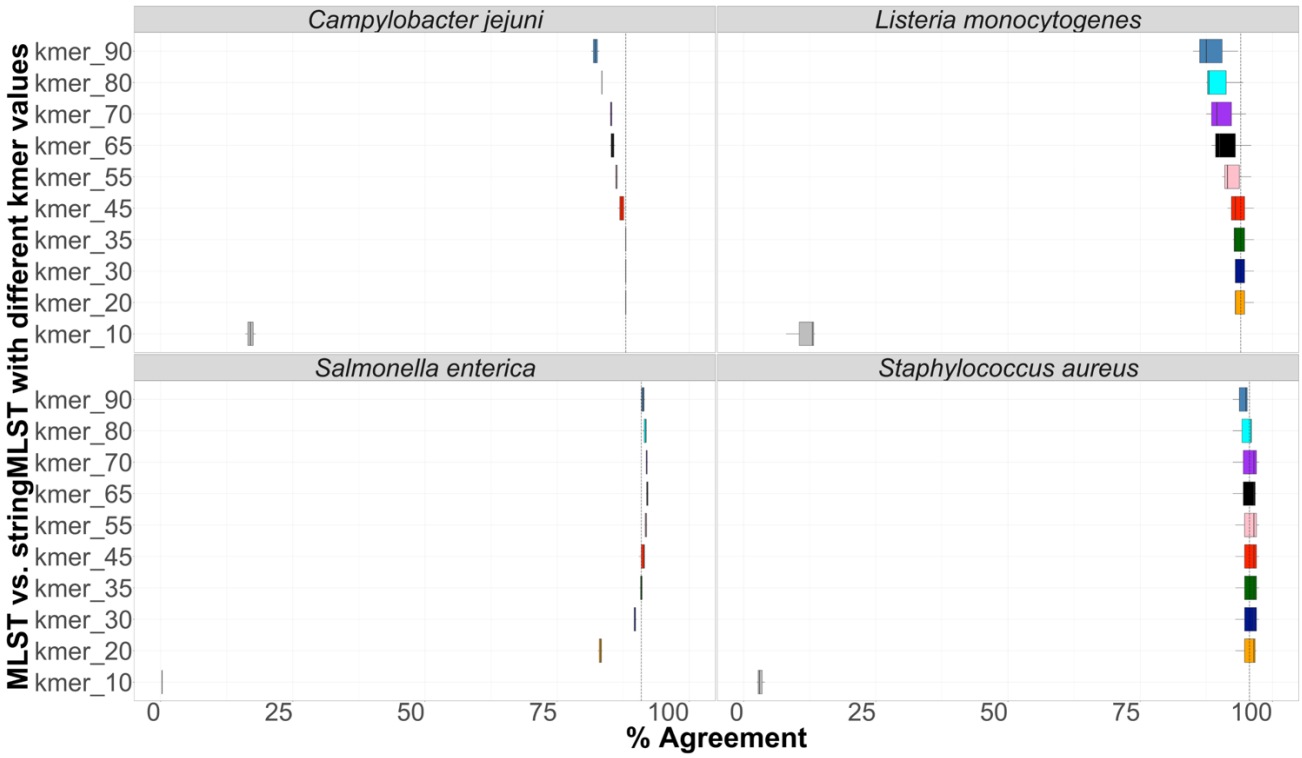

Fig 6. Box-and-whiskers plot depicting the concordance between mlst and stringMLST in ST calls.
Four different datasets belonging to four phylogenetic distinct bacterial pathogens, including *C. jejuni* (600 genomes), *L. monocytogenes* (600 genomes), *S. enterica* (11,787 genomes from 20 different serovars) and *S. aureus* (600 genomes) were run with mlst and stringMLST for ST-based classification. In the case of stringMLST, kmer lengths varied from 10 to 90 to identify the optimal value (highest percentage of agreement with the standard MLST approach), across all four species (y-axis). If both programs outputted identical ST calls (either number of missing/blank value), the call was defined as a match; otherwise, it was identified as a mismatch, and the percentage of agreement (x-axis, concordance) was calculated accordingly. The dashed line on the x-axis represents the percentage agreement for the kmer value of 35 which is used as a default parameter by stringMLST.

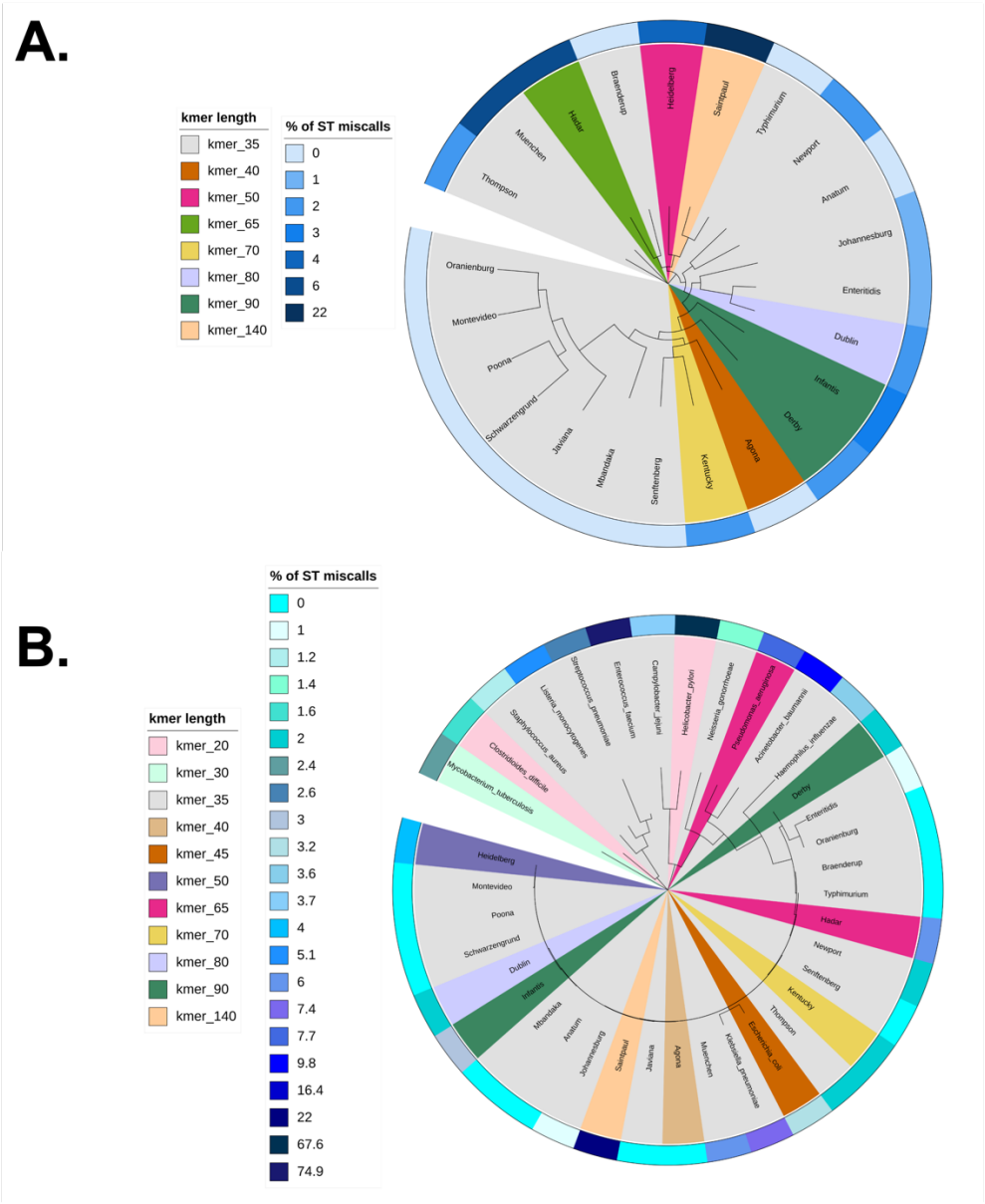

Fig 7. Phylogeny-guided display of optimal kmer length and algorithmic performance when using stringMLST for ST mapping across bacterial species. (A) Core-genome based phylogenetic display of stringMLST results across the twenty-three zoonotic serovars of *Salmonella enterica* subsp. *enterica* lineage I (*S. enterica*). The branches are colored based on the optimal kmer length which gives the lowest percentage of ST miscalls (ST calls that returned missing/blank values for stringMLST). The outer ring present in the phylogeny is colored based on the corresponding ST miscall percentages associated with each optimal kmer length. The dataset used to identify the optimal kmer length and percentage of ST miscalls

was composed of 2,300 genomes (100 genomes per serovar) and the phylogenetic tree was generated using twenty-three core-genome sequences (one of each serovar to facilitate data visualization); (B) Single locus-based (i.e., 16S rRNA gene) phylogenetic display of stringMLST results across fourteen divergent bacterial pathogens, including twenty-three representative genomes across each zoonotic serovar of the *S. enterica* species. The tree branches are colored based on the optimal kmer length which minimizes the percentage of ST miscalls (ST calls that returned missing/blank values for stringMLST). The outer ring present in the phylogeny corresponds to ST miscall percentage associated with each optimal kmer length. The dataset used to identify the optimal kmer length and percentage of ST miscalls was composed of 14,000 genomes (1,000 genomes for each bacterial pathogen) and 2,300 *Salmonella* genomes (100 genomes per serovar). The phylogeny (B) was generated using the 16S rRNA representative sequences across 37 bacterial species (one for each species was used to facilitate visualization). All phylogeny-based visualization were generated using iTOL version 6.4.

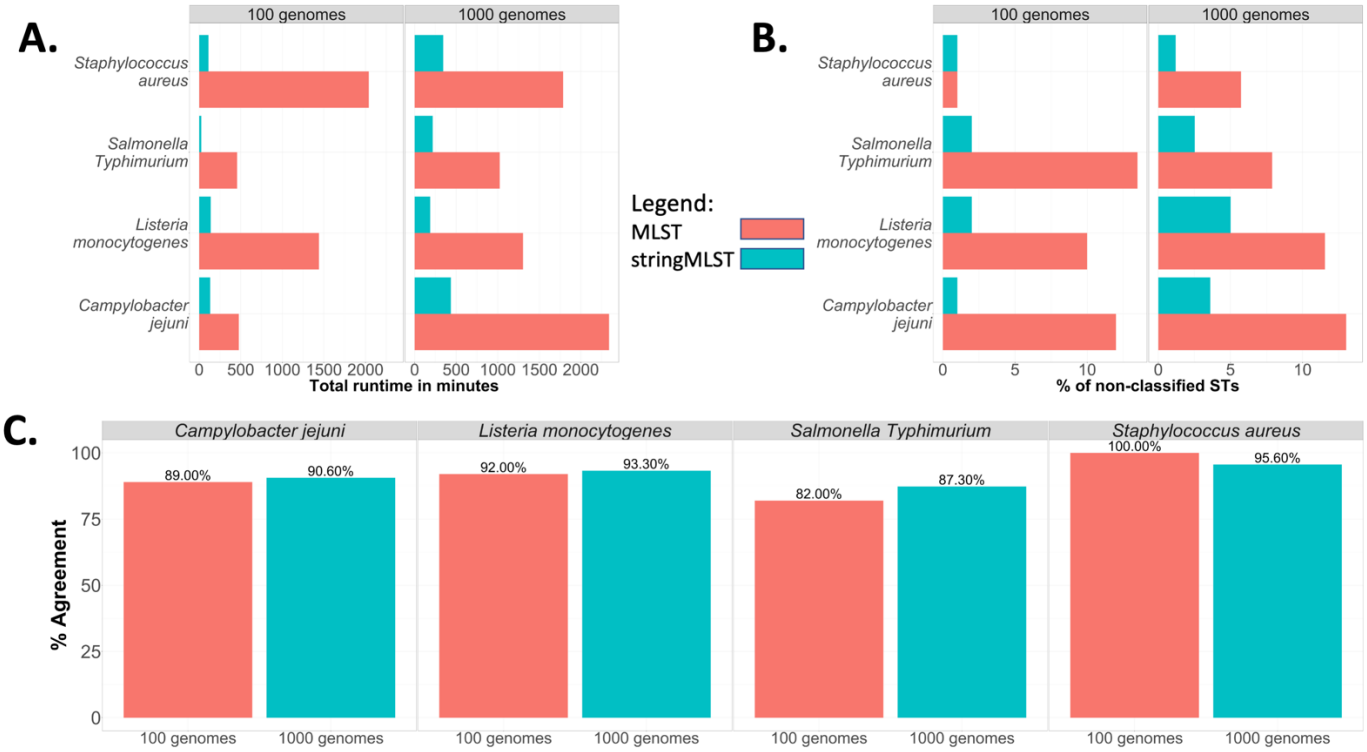

Fig 8. Comparison between the computational and statistical performance of mlst and stringMLST when using ProkEvo to run both programs.
Two subsets, one with 100 and the second one with 1,000 randomly chosen genomes, were selected from *C. jejuni*, *L. monocytogenes*, one major serovar of *S. enterica* (*S.* Typhimurium) and *S. aureus* to compare the performance of running mlst or stringMLST through ProkEvo. The performance and statistical metrics used for comparison were: (A) Total runtime of individual workflow in minutes; (B) Percentage of non-classified STs (ST calls that returned missing/blank values); and (C) Percentage of agreement (concordance) between programs ("good" or "bad" ST calls that matched between mlst and stringMLST).