A peer-reviewed version of this preprint was published in PeerJ on 1 April 2014.

View the peer-reviewed version (peerj.com/articles/332), which is the preferred citable publication unless you specifically need to cite this preprint.

Sahl JW, Caporaso JG, Rasko DA, Keim P. 2014. The large-scale blast score ratio (LS-BSR) pipeline: a method to rapidly compare genetic content between bacterial genomes. PeerJ 2:e332 https://doi.org/10.7717/peerj.332

Title: The large-scale blast score ratio (LS-BSR) pipeline: a method to rapidly compare genetic content between bacterial genomes

Running header: Comparative genomics with LS-BSR

Authors:

Jason W. Sahl*
Division of Pathogen Genomics
Translational Genomics Research Institute
3051 W. Shamrell Blvd, Suite 106
Flagstaff, AZ, 86001, USA
Email: jsahl@tgen.org

J. Gregory Caporaso
Department of Biological Sciences
Northern Arizona University
1298 S. Knoles Drive, PO Box 4073
Flagstaff, AZ, 86011, USA
Email: gregcaporaso@gmail.com

David A. Rasko
Department of Microbiology and Immunology
University of Maryland School of Medicine
Baltimore, MD, 21202, USA
Email: drasko@som.umaryland.edu

Paul Keim
Center for Microbial Genetics and Genomics
Northern Arizona University
1298 S. Knoles Drive, PO Box 4073
Flagstaff, AZ, 86011, USA
Email: paul.keim@nau.edu

*To whom correspondence should be addressed

PeerJ PrePrints | http://dx.doi.org/10.7287/peerj.preprints.220v1 | CC-BY 4.0 Open Access | received: 24 Jan 2014, published: 24 Jan 2014

Abstract

Background. As whole genome sequence data from bacterial isolates becomes cheaper to generate, computational methods are needed to correlate sequence data with biological observations. Here we present the large-scale BLAST score ratio (LS-BSR) pipeline, which rapidly compares the genetic content of hundreds to thousands of bacterial genomes, and returns a matrix that describes the relatedness of all coding sequences (CDSs) in all genomes surveyed. This matrix can be easily parsed in order to identify genetic relationships between bacterial genomes. Although pipelines have been published that group peptides by sequence similarity, no other software performs the large-scale, flexible, full-genome comparative analyses carried out by LS-BSR.

Results. To demonstrate the utility of the method, the LS-BSR pipeline was tested on 96 *Escherichia coli* and *Shigella* genomes; the pipeline ran in 163 minutes using 16 processors, which is a greater than 7-fold speedup compared to using a single processor. The BSR values for each CDS, which indicate a relative level of relatedness, were then mapped to each genome on an independent core genome single nucleotide polymorphism (SNP) based phylogeny. Comparisons were then used to identify clade specific CDS markers and validate the LS-BSR pipeline based on molecular markers that delineate between classical *E. coli* pathogenic variant (pathovar) designations. Scalability tests demonstrated that the LS-BSR pipeline can process 1,000 *E. coli* genomes in ~60h using 16 processors.

Conclusions. LS-BSR is an open-source, parallel implementation of the BSR algorithm, enabling rapid comparison of the genetic content of large numbers of genomes. The results of the pipeline can be used to identify specific markers between user-defined phylogenetic groups, and to identify the loss and/or acquisition of genetic information between bacterial isolates. Taxa-specific genetic markers can then be translated into clinical diagnostics, or can be used to identify broadly conserved putative therapeutic candidates.

PeerJ PrePrints | http://dx.doi.org/10.7287/peerj.preprints.220v1 | CC-BY 4.0 Open Access | received: 24 Jan 2014, published: 24 Jan 2014

INTRODUCTION

Whole genome sequence (WGS) data has changed our view of bacterial relatedness and evolution. Computational analyses available for WGS data include, but are not limited to, single nucleotide polymorphism (SNP) discovery (DePristo et al. 2011), core genome phylogenetics (Sahl et al. 2011), and gene based comparative methods (Hazen et al. 2013; Sahl et al. 2013). In 2005, a BLAST score ratio (BSR) method was introduced in order to compare peptide identity from a limited number of bacterial genomes (Rasko et al. 2005). However, the "all vs. all" implementation of this method scales poorly with a larger number of sequenced genomes. Here we present the Large Scale BSR method (LS-BSR) that can rapidly compare gene content of a large number of bacterial genomes. Comparable methods have been published in order to group genes into gene families, including OrthoMCL (Li et al. 2003), TribeMCL (Enright et al. 2002), and GETHOGs (Altenhoff et al. 2013). Although grouping peptides into gene families is not the primary focus of LS-BSR, the output can be parsed to identify the pan-genome (Tettelin et al. 2008) structure of a species; scripts are included with LS-BSR that classify coding sequences (CDSs) into pan-genome categories based on user-defined identity thresholds. Pipelines have also been established to perform comprehensive pan-genome analyses, including PGAP (Zhao et al. 2012), which requires gene annotation, and complicates the analysis of large numbers of novel genomes. GET_HOMOLOGUES (Contreras-Moreira & Vinuesa 2013) is a recently published tool that can be used for pan-genome analyses, including the generation of dendrograms based on the presence/absence of homologous genes; by only using presence/absence based on gene homology, more distantly related gene relatedness cannot be fully investigated. No previously published method carries out the large-scale, flexible, gene-based comparative methods currently performed by LS-BSR.

MATERIALS AND METHODS

The LS-BSR method can either use a defined set of genes, or can use Prodigal (Hyatt et al. 2010) to predict CDSs from a set of query genomes. When

PeerJ PrePrints | http://dx.doi.org/10.7287/peerj.preprints.220v1 | CC-BY 4.0 Open Access | received: 24 Jan 2014, published: 24 Jan 2014

using Prodigal, all CDSs are concatenated and then de-replicated using USEARCH (Edgar 2010) at a pairwise identity of 0.9 (identity threshold can be modified by the user). Each unique CDS is then translated with BioPython (www.biopython.org) and aligned against its nucleotide sequence with TBLASTN (Altschul et al. 1997) to calculate the reference bit score. Each query peptide is then aligned against each genome with TBLASTN and the query bit score is tabulated. The BSR value is calculated by dividing the query bit score by the reference bit score, resulting in a BSR value between 0.0 and 1.0 (values slightly higher than 1.0 have been observed due to variable bit score values obtained by TBLASTN). The results of the LS-BSR pipeline include a matrix that contains each unique CDS name, and the BSR value in each genome surveyed. CDSs that have more than one significant BSR values in at least one genome are also identified in the output. A separate file is generated for CDSs where one duplicate is significantly different than the other in at least one genome; these regions could represent paralogs and may require further detailed investigation. Once the LS-BSR matrix is generated, the results can easily be visualized as a heatmap or cluster with the Multiple Experiment Viewer (MeV) (Saeed et al. 2006); the heatmap represents a visual depiction of the relatedness of all peptides in the pan-genome across all genomes. A script is included with LS-BSR (compare_BSR.py) to rapidly compare CDSs between user-defined sub-groups, using a range of BSR thresholds set for CDS presence/absence. Annotation of identified CDSs can then be applied using tools including RAST (Aziz et al. 2008). LS-BSR source code and unit tests can be freely obtained at https://github.com/jasonsahl/LS-BSR under a GNU GPL v3 license.

RESULTS AND DISCUSSION

LS-BSR algorithm speed and scalability. To determine the scalability of the LS-BSR method, 1,000 *Escherichia coli* and *Shigella* genomes were downloaded from Genbank (Benson et al. 2012); *E. coli* was used as a test case due to the large number of genomes deposited in Genbank. Genomes were sub-sampled

PeerJ PrePrints | http://dx.doi.org/10.7287/peerj.preprints.220v1 | CC-BY 4.0 Open Access | received: 24 Jan 2014, published: 24 Jan 2014

at different depths (100 through 1000, sampling every 100) with a python script (https://gist.github.com/jasonsahl/115d22bfa35ac932d452) and processed with LS-BSR using 16 processors. A plot of wall time and the number of genomes processed demonstrates the scalability of the method (Figure 1A). To demonstrate the parallel nature of the algorithm, 100 *E. coli* genomes were processed with different numbers of processors. The results demonstrate decreased runtime of LS-BSR with an increase in the number of processors used (Figure 1B).

Improvements on a previous BSR implementation. The LS-BSR method is an improvement on a previous BSR implementation (http://bsr.igs.umaryland.edu/) in terms of speed and ease of use. The former BSR algorithm (Rasko, et al., 2005) requires peptide sequences and genomic coordinates of CDSs to run. LS-BSR only requires genome assemblies in FASTA format, which is the standard output of most genome assemblers. To test the speed differences between methods, 10 *E. coli* genomes (Supplemental Table 1) were processed with both methods. Using the same number of processors (n=2) on the same server, the original BSR method took ~14 hours (wall time) to complete, while the LS-BSR method took ~25 minutes to complete (wall time). Because the original BSR method is an "all vs. all" comparison and the LS-BSR method is a "one vs. all" comparison, this difference is expected to be more pronounced as the number of genomes analyzed increases.

Test case: analysis of 96 *E. coli* and *Shigella* genomes. To demonstrate the utility of the LS-BSR pipeline, a set of 96 *E. coli* and *Shigella* genomes were processed (Supplemental Table 1); these genomes are in various stages of assembly completeness and have been generated with various sequencing technologies from Sanger to Illumina. The BSR matrix was generated in 2h34m from a set of ~20,000 unique CDSs using 16 processors. In addition to the LS-BSR analysis, a core genome single nucleotide polymorphism (SNP) phylogeny was inferred on 96 genomes using methods published previously (Sahl et al.

PeerJ PrePrints | http://dx.doi.org/10.7287/peerj.preprints.220v1 | CC-BY 4.0 Open Access | received: 24 Jan 2014, published: 24 Jan 2014

2011); the SNP phylogeny with labels is shown in Supplemental Figure 1. Briefly, all genomes were aligned with Mugsy (Angiuoli & Salzberg 2010) and the core genome was extracted from the whole genome alignment; the alignment file was then converted into a multiple sequence alignment in FASTA format. Gaps in the alignment were removed with Mothur (Schloss et al. 2009) and a phylogeny was inferred on the reduced alignment with FastTree2 (Price et al. 2010).

The compare_BSR.py script included with LS-BSR was used to identity CDS markers that are unique to specific phylogenetic clades (Figure 2). Identified CDSs had a BSR value ≥0.8 in targeted genomes and a BSR value <0.4 in non-targeted genomes; the gene annotation of all marker CDSs is detailed in Supplemental Table 2. The conservation and distribution of all clade-specific markers was visualized by correlating the phylogeny with a heatmap of BSR values (Figure 2). This presentation provides an easy way for the user to highlight features conserved in one or more phylogenomic clades.

E. coli and *Shigella* pathogenic variants (pathovars) are delineated by the presence of genetic markers primarily present on mobile genetic elements (Rasko et al. 2008). The conservation of these markers was used as a validation of the LS-BSR method. A representative sequence from each pathovar-specific marker (Supplemental Table 2) was screened against the 96-genome test set and the BSR values (Supplemental Table 3) were visualized as a heatmap (Figure 2). The BSR matrix demonstrates that pathovar-specific genes were accurately identified in each targeted genome (Supplemental Table 3, Figure 2). For example, the *ipaH3* marker was positively identified in all *Shigella* genomes and the Shiga toxin gene (*stx2a*) was conserved in the clade including O157:H7 *E. coli* (Figure 2). A sub-set of these 96 *E. coli* genomes is included with the repository as test data to characterize the conservation and distribution of pathovar specific genes.

Finally, the BSR values were used to cluster all 96 genomes with an average linkage algorithm implemented in MeV and the structure of the resulting dendrogram was compared to the core SNP phylogeny. The BSR based

PeerJ PrePrints | http://dx.doi.org/10.7287/peerj.preprints.220v1 | CC-BY 4.0 Open Access | received: 24 Jan 2014, published: 24 Jan 2014

clustering method incorporates both the core and accessory genome, while the SNP phylogeny relies on core genomic regions alone. A comparison of the tree structures demonstrates that while *Shigella* genomes share a diverse evolutionary history (Figure 3A), they all cluster together based on gene presence and conservation (Figure 3B). This result was also observed using a k-mer frequency method (Sims & Kim 2011), which uses all possible k-mer values to infer a phylogeny and validates the findings of the LS-BSR pipeline. The dendrogram also differed from the core SNP phylogeny in other genomes, which could represent either assembly problems, or more likely the acquisition of accessory genomic regions that are not a product of direct descent.

LS-BSR was compared to a recently released software package, GET_HOMOLOGUES (Contreras-Moreira & Vinuesa 2013), which performs several pan-genome based analyses. A set of 100 *E. coli* / *Shigella* genomes was chosen for the comparative analysis. For LS-BSR, the genome assemblies were used, while for GET_HOMOLOGUES, CDSs were identified with Prodigal and the resulting peptides were used as input. LS-BSR finished in 2h39m, while the clustering step in GET_HOMOLOGUES took 29h20m to finish using the same number of allocated processors. Based on this result, LS-BSR offers a significant speedup compared to comparable methods for large-scale genetic comparisons.

CONCLUSIONS

The LS-BSR method can rapidly compare the gene content of a relatively large number of bacterial genomes in either draft or complete form, though with more fragmented assemblies LS-BSR is likely to perform sub-optimally. As sequence read lengths improve, assembly fragmentation should become less problematic due to more contiguous assemblies. LS-BSR can also be used to rapidly screen a collection of genomes for the conservation of known virulence factors or genetic features. By using a range of peptide relatedness, instead of a defined threshold, homologs and paralogs can also be identified for further characterization.

PeerJ PrePrints | http://dx.doi.org/10.7287/peerj.preprints.220v1 | CC-BY 4.0 Open Access | received: 24 Jan 2014, published: 24 Jan 2014

LS-BSR is written in python, with many steps conducted in parallel. This allows the script to scale well from hundreds to thousands of genomes. The LS-BSR method is a major improvement on a previous BSR implementation in terms of speed, ease of use, and utility. As more WGS data from bacterial genomes becomes available, methods will be required to quickly compare their genetic content and perform pan-genome analyses. LS-BSR is an open-source software package to rapidly perform these comparative genomic workflows.

ACKNOWLEDGEMENTS

This work was funded by the NAU Technology and Research Initiative Fund (TRIF). Thanks to Darrin Lemmer for his critical review of the LS-BSR code.

REFERENCES

Altenhoff AM, Gil M, Gonnet GH, and Dessimoz C. 2013. Inferring hierarchical orthologous groups from orthologous gene pairs. *PLoS ONE* 8:e53786.

Altschul SF, Madden TL, Schaffer AA, Zhang J, Zhang Z, Miller W, and Lipman DJ. 1997. Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. *Nucleic Acids Res* 25:3389-3402.

Angiuoli SV, and Salzberg SL. 2010. Mugsy: Fast multiple alignment of closely related whole genomes. *Bioinformatics*.

Aziz RK, Bartels D, Best AA, DeJongh M, Disz T, Edwards RA, Formsma K, Gerdes S, Glass EM, Kubal M et al. . 2008. The RAST Server: rapid annotations using subsystems technology. *BMC genomics* 9:75.

Benson DA, Karsch-Mizrachi I, Clark K, Lipman DJ, Ostell J, and Sayers EW. 2012. GenBank. *Nucleic Acids Res* 40:D48-53.

Contreras-Moreira B, and Vinuesa P. 2013. GET_HOMOLOGUES, a versatile software package for scalable and robust microbial pan-genome analysis. *Applied and Environmental Microbiology*.

DePristo MA, Banks E, Poplin R, Garimella KV, Maguire JR, Hartl C, Philippakis AA, del Angel G, Rivas MA, Hanna M et al. . 2011. A framework for variation discovery and genotyping using next-generation DNA sequencing data. *Nature genetics* 43:491-498.

Edgar RC. 2010. Search and clustering orders of magnitude faster than BLAST. *Bioinformatics* 26:2460-2461.

Enright AJ, Van Dongen S, and Ouzounis CA. 2002. An efficient algorithm for large-scale detection of protein families. *Nucleic acids research* 30:1575-1584.

Hazen TH, Sahl JW, Fraser CM, Donnenberg MS, Scheutz F, and Rasko DA. 2013. Refining the pathovar paradigm via phylogenomics of the attaching and

PeerJ PrePrints | http://dx.doi.org/10.7287/peerj.preprints.220v1 | CC-BY 4.0 Open Access | received: 24 Jan 2014, published: 24 Jan 2014

effacing *Escherichia coli. Proceedings of the National Academy of Sciences of the United States of America*.

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

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

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

Rasko DA, Myers GS, and Ravel J. 2005. Visualization of comparative genomic analyses by BLAST score ratio. *BMC Bioinformatics* 6:2.

Rasko DA, Rosovitz MJ, Myers GS, Mongodin EF, Fricke WF, Gajer P, Crabtree J, Sebaihia M, Thomson NR, Chaudhuri R et al. . 2008. The pangenome structure of *Escherichia coli*: comparative genomic analysis of *E. coli* commensal and pathogenic isolates. *J Bacteriol* 190:6881-6893.

Saeed AI, Bhagabati NK, Braisted JC, Liang W, Sharov V, Howe EA, Li J, Thiagarajan M, White JA, and Quackenbush J. 2006. TM4 microarray software suite. *Methods Enzymol* 411:134-193.

Sahl JW, Gillece JD, Schupp JM, Waddell VG, Driebe EM, Engelthaler DM, and Keim P. 2013. Evolution of a pathogen: a comparative genomics analysis identifies a genetic pathway to pathogenesis in *Acinetobacter*. *PLoS ONE* 8:e54287.

Sahl JW, Steinsland H, Redman JC, Angiuoli SV, Nataro JP, Sommerfelt H, and Rasko DA. 2011. A comparative genomic analysis of diverse clonal types of enterotoxigenic *Escherichia coli* reveals pathovar-specific conservation. *Infect Immun* 79:950-960.

Schloss PD, Westcott SL, Ryabin T, Hall JR, Hartmann M, Hollister EB, Lesniewski RA, Oakley BB, Parks DH, Robinson CJ et al. . 2009. Introducing mothur: Open-Source, Platform-Independent, Community-Supported Software for Describing and Comparing Microbial Communities. *Appl Environ Microbiol* 75:7537-7541.

Sims GE, and Kim SH. 2011. Whole-genome phylogeny of *Escherichia coli*/*Shigella* group by feature frequency profiles (FFPs). *Proc Natl Acad Sci U S A*.

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

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

PeerJ PrePrints | http://dx.doi.org/10.7287/peerj.preprints.220v1 | CC-BY 4.0 Open Access | received: 24 Jan 2014, published: 24 Jan 2014

Figure Legends:

Figure 1. Time performance of the LS-BSR pipeline. **Panel A)** 1000 *Escherichia coli* and *Shigella* genomes were randomly sub-sampled and analyzed using default LS-BSR parameters and 16 processors. Wall time was plotted against the number of genomes analyzed. The results demonstrate that the LS-BSR pipeline scales well with increasing numbers of genomes. **Panel B)** The same set of 100 *E. coli* genomes was processed with different numbers of processors and the wall time was plotted. The results demonstrate that using additional processors decreases the overall run time of LS-BSR.

Figure 2. The distribution of virulence factors and phylogenomic markers associated with a core single nucleotide polymorphism (SNP) phylogeny. The core SNP phylogeny was inferred from a whole genome alignment produced by Mugsy (Angiuoli & Salzberg 2010). Known virulence genes (Supplemental Table 2) were screened against 96 *Escherichia coli* and *Shigella* genomes using BLASTN within LS-BSR. Clade specific markers were identified at defined nodes in the phylogeny (A through Q). Gene annotations for these markers are detailed in Supplemental Table 2.

Figure 3. A comparison of 96 *Escherichia coli* / *Shigella* genomes between (Panel A) a core single nucleotide polymorphism (SNP) phylogeny or (Panel B) a cluster generated with the Multiple Experiment Viewer (Saeed et al. 2006) from BLAST Score Ratio (BSR) values that include the entire pan-genome. Colors applied to each classical *E. coli* phylogroup were applied to the SNP phylogeny and transferred to the BSR cladogram. *Shigella* genomes are marked with a red circle.

Supplemental Figure 1. A core genome SNP phylogeny of 96 *Escherichia coli* and *Shigella* genomes. The core genome was extracted from the output of

PeerJ PrePrints | http://dx.doi.org/10.7287/peerj.preprints.220v1 | CC-BY 4.0 Open Access | received: 24 Jan 2014, published: 24 Jan 2014

Mugsy (Angiuoli & Salzberg 2010) and the phylogeny was inferred with FastTree2 (Price et al. 2010). This phylogeny contains labels that can be used to identify specific genomes in Figures 2 and 3.

PeerJ PrePrints | http://dx.doi.org/10.7287/peerj.preprints.220v1 | CC-BY 4.0 Open Access | received: 24 Jan 2014, published: 24 Jan 2014

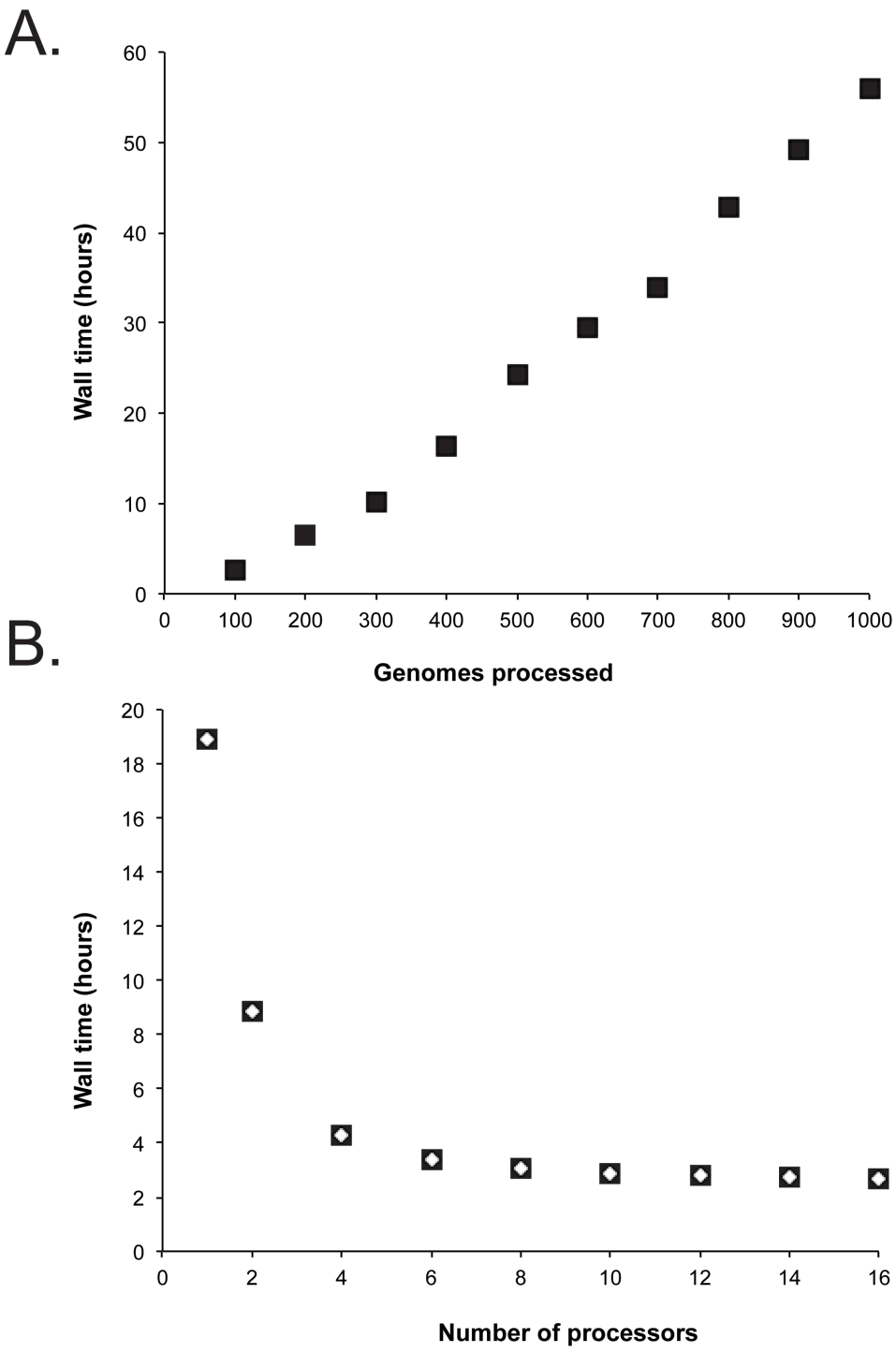

Figure 1

PeerJ PrePrints | http://dx.doi.org/10.7287/peerj.preprints.220v1 | CC-BY 4.0 Open Access | received: 24 Jan 2014, published: 24 Jan 2014

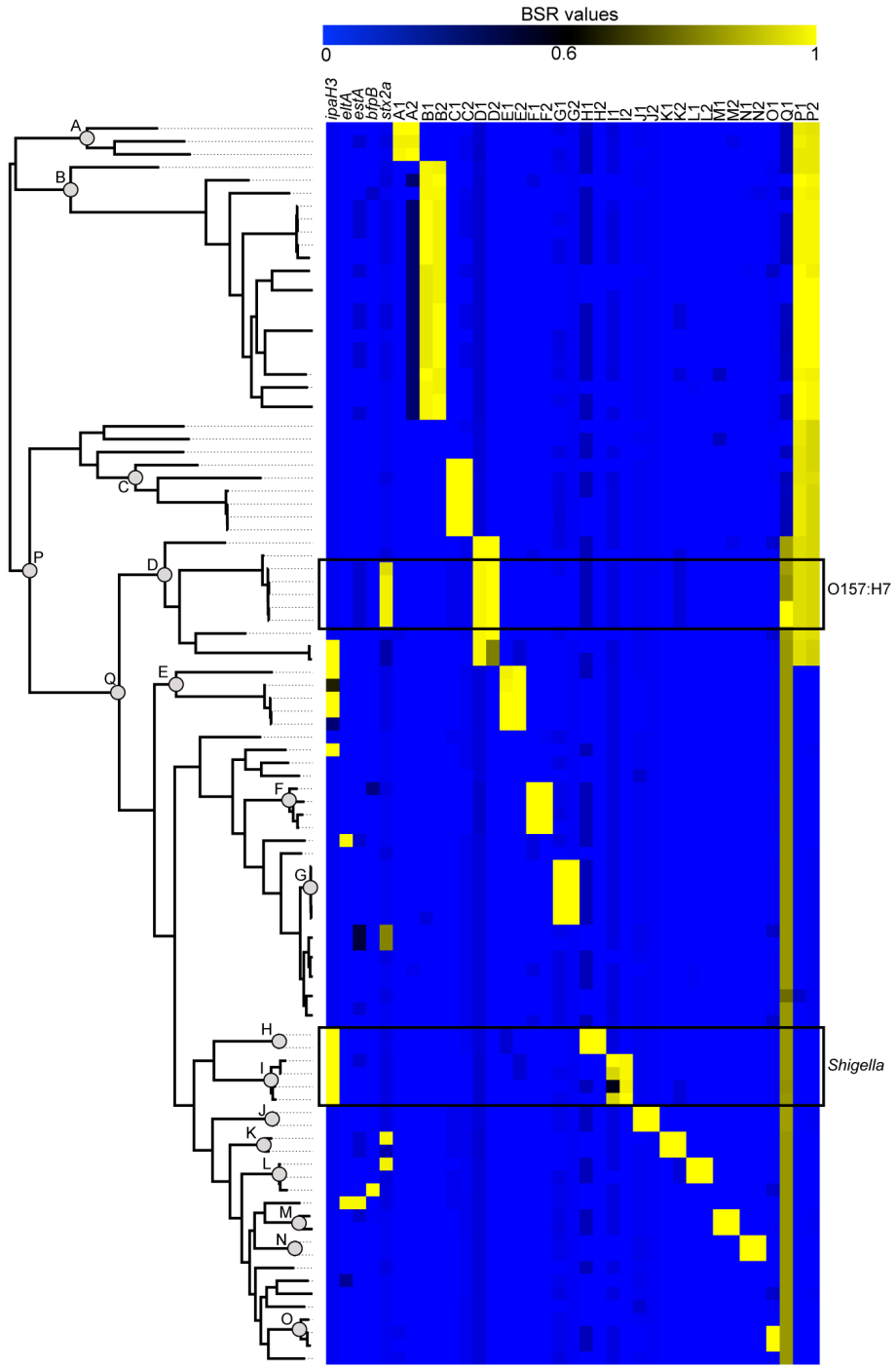

Figure 2

PeerJ PrePrints | http://dx.doi.org/10.7287/peerj.preprints.220v1 | CC-BY 4.0 Open Access | received: 24 Jan 2014, published: 24 Jan 2014

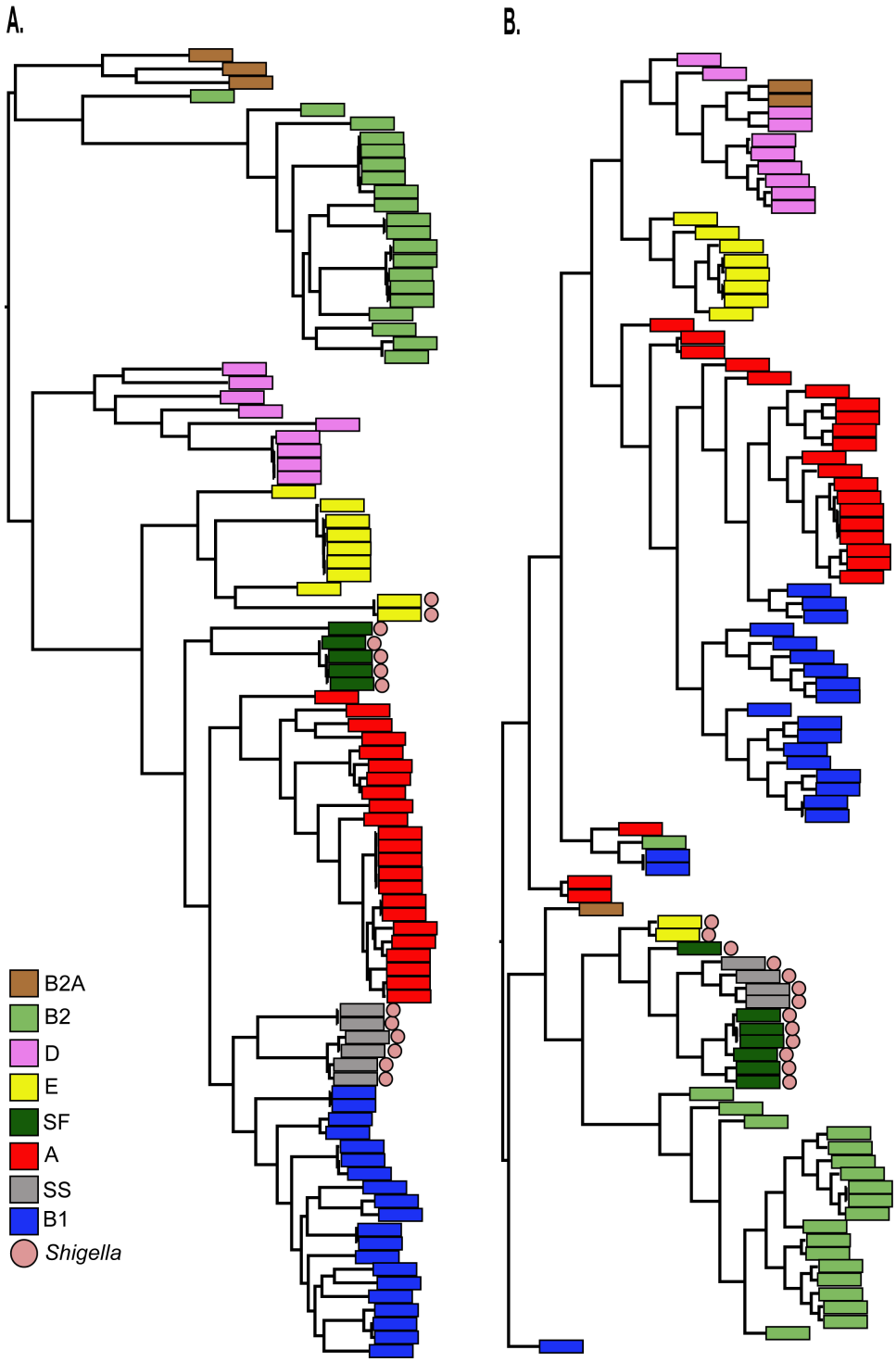

Figure 3

PeerJ PrePrints | http://dx.doi.org/10.7287/peerj.preprints.220v1 | CC-BY 4.0 Open Access | received: 24 Jan 2014, published: 24 Jan 2014