CAMBer: An Approach to Support Comparative Analysis of Multiple Bacterial Strains

Michal Wozniak*‡, Limsoon Wong†, Jerzy Tiuryn*

*Faculty of Mathematics, Informatics and Mechanics, University of Warsaw, Poland.

†School of Computing, National University of Singapore, Singapore.

‡Corresponding author: Michal Wozniak, m.wozniak@mimuw.edu.pl.

Abstract

There is a large amount of inconsistency in gene structure annotations of bacterial strains. This inconsistency is a frustrating impedance to effective comparative genomic analysis of bacterial strains in promising applications such as gaining insights into bacterial drug resistance. Here, we propose CAMBer as an approach to support comparative analysis of multiple bacterial strains. CAMBer produces what we called multigene families. Each multigene family reveals genes that are in one-to-one correspondence in the bacterial strains, thereby permitting their annotations to be integrated. As a result, more accurate and more comprehensive annotations of the bacterial strains can be produced.

1 Introduction

Large amounts of genomic information are currently being generated, including whole-genome sequences of multiple strains of many bacterial species. The availability of these sequences provides exciting opportunities and applications for comparative genomic analysis of multiple bacterial strains. For example, comparative genomic analysis of the avirulent H37Ra and virulent H37Rv strains of *M. tuberculosis* provides insights into the virulence and pathogenesis of *M. tuberculosis* [14]. As another example, comparative genomic analysis of three linezolid-resistant *S. pneumoniae* strains identified three mutations and the associated genes involved in antibiotic resistance [5]. As a last example, an ingenious comparative genomic analysis of susceptible and resistant strains of *M. tuberculosis* and *M. smegmatis* and found that the only gene commonly affected in all three resistant strains encodes atpE, thereby uncovering the mode of action of the novel class of compound Diarylquinoline to be the inhibition of the proton pump of *M. tuberculosis* ATP synthase [1].

These impressive results were achieved by integrating and connecting information generated during the sequencing of multiple distinct strains of the bacteria species mentioned. In order to repeat these past successes, there is a need for a general annotation consensus, as the physical and functional annotations of the strains of the same bacteria differ significantly in some cases. As an extreme case of the problem, the strains of *E. coli* reportedly have only 20% of their genes in common [8]. One cause for the inconsistency of gene annotations is sequencing errors. For example, surprised by the higher similarity between H37Ra and CDC1551 *M. tuberculosis* strains than that between H37Ra and H37Rv, Zheng et al. [14] re-sequenced the relevant loci in H37Rv and discovered a mere 6 out of 85 of the variations were genuine and the rest were sequencing errors [14]. A second cause for gene annotation inconsistency is gene structure prediction errors. For example, when Wakaguri et al. determined the entire sequences of 732 cDNAs in *T. gondii* to evaluate earlier annotated gene models of *T. gondii* at the complete full-length transcript level, they found that 41% of the gene models contained at least one inconsistency [12]. Also, a persistent weakness of gene structure prediction methods is the accuracy of start codon assignment [2], giving rise to a significant amount of gene annotation inconsistency from the resulting gene size variations. Another cause for

the inconsistency of gene annotations is the inability to put genes from different strains into correct gene families. For example, the extreme case of *E. coli* is probably due to the simple-minded BLAST reciprocal pairwise comparison that was used in [8] to identify genes belonging to the same gene family. This strategy may identify as few as 15% of genes that are known to have evolutionary relationship; a more sophisticated strategy based on linking by intermediate sequences—a strategy that we also adopt—may increase the ability to recognize genes evolutionary relationship by 70% [9].

This is a frustrating state of affairs for both biologists and bioinformaticians. Therefore, we require structured, exhaustive, comparative databases. While broad-based, web-technology-enabled community annotation has been proposed as a solution to the problem [10], it is feasible only for species having a large interested research community. Unfortunately, this may not be the case for many bacterial strains such as *M. Tuberculosis* due to, for example, insufficient profit opportunity [11]. Therefore, we should explore the development of approaches and technologies that integrate, connect, and produce consensus gene annotations to support comparative analysis of multiple bacterial strains.

We have designed CAMBer to support comparative analysis of multiple bacterial strains. CAMBer approaches the problem as follows. First, we use intermediate sequences—a tactic originally proposed for enhancing FASTA's ability to detect evolutionary relationship [9]—to link multiple annotations on a gene. We call the resulting structure a *multigene*. Next, multigenes are linked by BLAST edges between their elements into a *consolidation graph*. Multigenes in the same connected component of the graph are proposed to form a family. Finally, we use genomic context information—a tactic originally proposed for enhancing gene function prediction [13]—to refine the consolidation graph. This way we obtain more multigene families where the multigenes in each family are in one-to-one relationship in the bacterial strains considered. These resulting multigene families can be used to support more detailed comparative analysis of multiple bacterial strains for detecting sequencing error, identifying mutations for drug resistance, and other purposes.

In the remainder of this paper, we present the details of CAMBer and our results on *M. tuberculosis*.

2 Methods

We present here the details of our approach. We assume that we have a set of bacterial strains whose genomes have been sequenced and annotated. The goal is to arrive at revised annotations of the strains which arise from projecting an annotation of one strain onto the annotations of another. Furthermore, we focus on Translation Initiation Site (TIS) annotations. In this operation, we do not remove the original TIS in the second strain, but rather add new TISs suggested by the annotations of the first one. In particular, we may arrive at new annotated genes in the second strain. In this way, we naturally arrive at the concept of a *multigene* which is just a gene with possibly several TISs.

More precisely: Given an annotation A in strain S_1 and let x be an ORF which according to A encodes a gene in S_1. We run BLAST with query x against the sequence of a genome of a strain S_2. Let y' be a hit in S_2 returned by BLAST to the query x and let y be the sequence obtained from y' by extending it to the nearest stop codon (in the 3' direction on the same strand as y'). We call y an *acceptable hit* with respect to x if the following four conditions are satisfied:

- y starts with one of the appropriate start codons: ATG, GTG, TTG.
- The BLAST hit y' has aligned beginning of the query x with the beginning of y'.
- The e-value score of the BLAST hit from x to y' is below a given threshold e_t (typically it is set to 10^{-10} or 10^{-20}).
- The ratio of the length of y to x is less than $1+p_t$ and greater than $1-p_t$, where p_t is a given threshold (typically 0.2 or 0.3). This condition is imposed in order to keep similar lengths of related sequences.
- The percent of identity of the hit is above the threshold i_t (typically set to 50% or 70%).

So, assuming that we use BLAST with default parameters our method has three specific parameters: e-value threshold e_t, length tolerance threshold p_t, and percent of identity threshold i_t.

It follows from the definition above that an acceptable hit y may overlap a gene annotated in S_2 in the same frame, sharing the same stop codon,

but having a different TIS. As mentioned above this gives rise to the notion of a multigene. Different TISs in a multigene g give rise to different putative genes. We call them elements of g. Obviously genes can be viewed as multigenes with just one element.

2.1 Consolidation graph

We compute iteratively a closure of annotations which is based on the above described operation of taking acceptable hits. Initially, as step zero, we take original annotations which are furnished with the genomic sequences. Assume that at step $i \geq 0$ we have an annotation $A_S^{(i)}$ associated with each strain S. Annotation $A_S^{(i+1)}$ in the step $i+1$ is obtained by taking all acceptable hits in S for the queries ranging over all genes annotated in $A_T^{(i)}$, for every other strain T. This process stops when no new acceptable hit is obtained. This process generalizes a proven strategy for identifying more homologs by linking intermediate sequences [9].

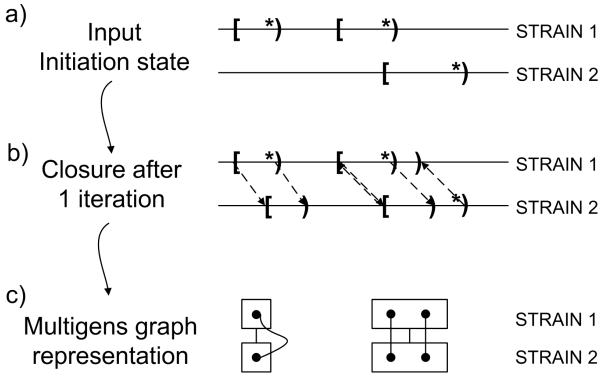

Figure 1: Schema of our method to represent the structure of multigenes. For clarity of presentation only one step of the procedure is shown. Square brackets correspond to stop codons of annotated genes, while round brackets with a star correspond to start codons of annotated genes. Round brackets without a star correspond to putative genes indicated by our method (new elements of the multigene). a) Input annotations for strains indicate the initial state of the procedure. b) Dashed arrows indicate acceptable hits. The reader should notice a birth of a second element, rendering a multigene with two elements. c) Two examples of edges in the consolidation graph. Dots represent different elements of a multigene which is represented here as a rectangle. Edges connecting dots represent acceptable hits (we ignore directions here). Edges between rectangles represent edges of the consolidation graph.

Having computed the closure we can construct now a *consolidation graph* G. Its nodes are all multigenes obtained during the process of computing the closure. There is an edge from a multigene g to a multigene g' if one of the elements of g' is obtained as an acceptable hit with respect to one of the elements of g. Figure 1 illustrates the process of computing the closure, as well as a construction of the consolidation graph.

2.2 Refinement of the consolidation graph

Connected components of the consolidation graph G represent multigene families with a common ancestor. Our next goal is to refine the multigene homology relation represented by edges in G to obtain as many one-to-one homology classes as possible, i.e. having at most one multigene per strain in such a class. We call a connected component of G an *anchor* if it includes at most one multigene for every strain. One-element anchors are called *orphans*. *Non-anchors* are the components which fail to be anchors. Obviously the above definitions of anchors, orphans, and non-anchors apply to any graph with nodes being multigenes from various strains.

Multigenes in the same anchor are potentially orthologous to each other. In contrast, a non-anchor contains at least two multigenes that are potentially non-orthologous. Genomic context information has been successfully used to clarify gene relationships and improve gene function prediction [13]. So, we propose exploiting genomic context information to analyse and decompose non-anchors into smaller connected subgraphs that can emerge as anchors in the resulting refined consolidation graph.

Our construction of the *refinement* proceeds in stages. At each stage we carry a graph which is a subgraph of the graph from the previous stage. At stage 0, the original consolidation graph G is used as the initial input graph $G^{(0)}$.

Suppose we have at stage i a graph $G^{(i)}$. We restrict this graph by performing the following test on each pair (g, g') of multigenes originating from strains S_1 and S_2, connected by an edge in $G^{(i)}$ which belongs to a non-anchor component of $G^{(i)}$. Let a be the nearest left neighbor multigene of g in S_1 which belongs to an anchor of $G^{(i)}$ containing a multigene from S_2. Similarly let b be the nearest right neighbor multigene of g in S_1 which belongs to an anchor of $G^{(i)}$ containing a multigene from S_2.

In similar way define left (a') and right (b') neighbors of g' in S_2. Assuming that all four multigenes a, a', b, b' exist, we keep the edge connecting g and g' in $G^{(i+1)}$ if either (a, a') and (b, b') (see Figure 2 (a)), or (a, b') and (b, a') (see Figure 2 (b)) are edges in $G^{(i)}$, i.e. the corresponding pairs are in the same anchors of $G^{(i)}$. If at least one of the multigenes a, a', b, b' does not exist, the edge connecting g and g' in $G^{(i+1)}$ is not copied from $G^{(i)}$. The procedure stops when no edge is removed from the current graph. We call the resulting graph a *refinement* of G. Figure 2 (c) shows a situation when we have to retain two edges, leading to a cluster with unresolved one-to-one relationship. These cases may get resolved later when more anchors are obtained.

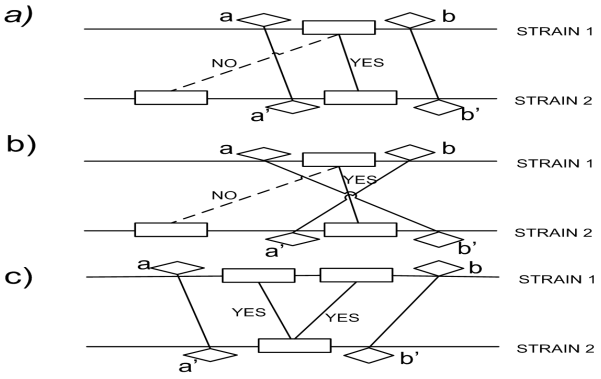

Figure 2: One step of the refinement procedure. Rectangles denote multigenes which belong to non-anchors in the current stage. Rhombus denotes a multigene which is already in an anchor at this stage. Edges connecting rectangles (dashed and solid) are edges of the graph of the current stage. Edges connecting rhombuses are the anchor edges. 'YES' means that the edge is keep for the next stage, while 'NO' means it is omitted. Parts a) and b) illustrate two situations when we can select one of the edges and leaving out the other. Part c) illustrates the situation when we cannot make such a decision, leading to an unresolved cluster. Both edges are kept in the graph of the next stage. Such a cluster may be resolved at a later stage. Other cases which lead to omitting the edges are possible too.

2.3 Time complexity

The most time consuming operation in the closure procedure is running BLAST. We denote by $blast()$ the BLAST running time. Let k be the number of all considered strains and let n be the maximal number of annotated genes in the genomes under consideration. For each strain during computing the closure operation we use every identified or annotated ORF only once. Assuming that the number of newly discovered multigenes does not grow fast, we can estimate the total time of the procedure as $k^2 * n * blast()$.

Now, we estimate time complexity of one iteration in the refinement procedure. Again, let k be the number of all considered strains and let n be the maximal number of identified multigenes among all strains. Denote by m the number of non-anchors in the consolidation graph. Additionally, let p denote the maximal number of multigenes for one strain among all non-anchor components. In order to find the nearest left and right neighbors of a multigene in linear time we first sort all of them. This takes time $k * n * logn$. Since we have at most $p^2 * \binom{k}{2}$ edges to check for support of the neighboring anchors (checking for support may take time at most n), for each of the m non-anchors, it follows that the estimated total time to resolve all of the m non-anchors is $k * n * logn + m * p^2 * \binom{k}{2} * n$.

3 Case study - MTB strains

We applied our approach, called CAMBer, to nine *M. tuberculosis* (MTB) strains. Tuberculosis is still a major cause of deaths worldwide, in particular due to still poorly-understood mechanisms of drug resistance. The first fully sequenced MTB strain was H37Rv and since then several new strains have been sequenced [3, 4, 7, 14].

Table 1 gives details of the strain data. We notice that a substantial variance (left box plot in Figure 3) in the number of originally annotated genes. To a large extent this is probably due to different gene finding tools and methodologies being applied by different labs, rather than the real genomic composition.

We have run our method with the following parameters: $e_t = 10^{-10}$, $p_t = 0.3$ and $i_t = 50\%$. It is quite remarkable that variance in the number of predicted multigenes after the closure is much smaller (right box plot in Figure 3). The reader may also compare the corresponding data presented in Tables 1 and 2. Table 2 shows for each strain the distribution of multigenes with respect to the number of elements. By far the largest group in each strain are one element multigenes.

The consolidation graph contains 4176 connected

strain ID	source	resist.	# of genes	lab.
H37Rv	NCBI ID: NC_000962	DS	3988(26)	S
H37Ra	NCBI ID: NC_009525	DS	4034(22)	C
F11	NCBI ID: NC_009565	DS	3941(5)	T
KZN 4207(T)	PLoS One. 2009 [7]	DS	3902(47)	T
KZN 4207(B)	Broad Institute	DS	3996(4)	B
KZN 1435	Broad Institute	MDR	4059(10)	B
KZN V2475	PLoS One. 2009 [7]	MDR	3893(3792)	T
KZN 605	Broad Institute	XDR	4024(26)	B
KZN R506	PLoS One. 2009 [7]	XDR	3902(46)	T

Table 1: Details for input strains for the case study. The first number in column called '*# of genes*' corresponds to the number of annotated genes, the second (in brackets) corresponds to the number of genes excluded in the study due to unusual start or stop codons or sequence length not divisible by three. In order to avoid ambiguity in naming the same strain sequenced by two labs we introduce an additional suffix (T or B). Characters in last column, called '*lab.*', describe the sequencing laboratories: B - The Broad Institute, T - Texas A&M University, C - Chinese National Human Genome Center at Shanghai, S - Sanger Institute.

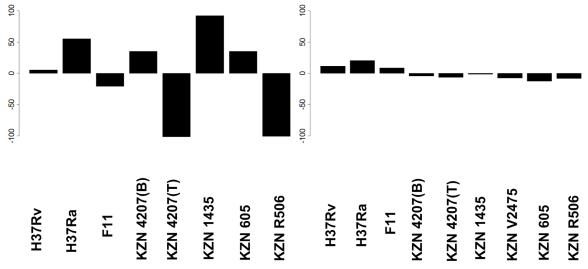

Figure 3: Left: deviation from mean (=3957) in numbers of annotated protein coding genes (KZN V2475 is omitted, due to very high difference). Right: deviation from mean (=4146) in numbers of multigenes after unification by the closure procedure. The same scale is used for both charts. Level 0 in the Y axis corresponds to the mean value.

components, with only 43 components (about 1%) being non-anchors and 48 being orphans. After the refinement procedure we obtained slightly more connected components (4288), but the number of non-anchors substantially dropped to 21 (Table 3).

It took about 10 hours (on a computer with 16 cores, 3000 MHz, 64 GB RAM) to compute the consolidation graph and only several minutes to perform the refinement procedure.

4 Discussion

As the number of sequenced genomes of closely related bacterial strains grows, as shown by the examples given in this paper, there is a need to join and consolidate different annotations of genomes. It turns out that annotations of related strains are often inconsistent in declaring Translation Initiation Sites (TIS) for the corresponding homologous genes. They also sometimes miss a gene in a segment which sequence-wise is very similar to a segment in the genome of another species which is declared as a gene. We propose in this paper a methodology which consists in collecting all possible different TISs, as well as genes which are present sequence-wise in a strain but whose annotation is missing. We believe this is the right approach toward correcting annotations.

	# of multigenes with					
	5 elt.	4 elt.	3 elt.	2 elt.	1 elt.	total
H37Rv	1	6	66	601	3484	4158
H37Ra	1	5	66	606	3489	4167
F11	1	6	68	605	3475	4155
KZN 4207(T)	1	6	70	600	3463	4140
KZN 4207(B)	1	5	69	601	3466	4142
KZN 1435	1	6	69	596	3473	4145
KZN V2475	1	6	70	601	3461	4139
KZN 605	1	6	68	601	3458	4134
KZN R506	1	6	70	602	3459	4138

Table 2: Multigene start sites statistics after the closure procedure.

	# of connected components before refinement	# of connected components after refinement
all connected components	4176	4288
non-anchors	43	21
anchors	4133	4267
orphans	48	68
anchors in all strains	3943	4013

Table 3: Statistics of the connected components before and after refinement.

To achieve this goal we constructed a *consolidation graph* which is based on the concept called here a *multigene*. Multigene is an entity which combines all different TISs derived from sequence comparisons with genes annotated in other strains, or genes which were already established as multigenes. The transitive closure of this operation on all genomes of interest gives the space of multigenes. Multigenes serve as nodes of the consolidation graph. Each TIS in a multigene gives rise to a gene which we called an *element* of the multigene. All elements of a given multigene share the same stop codon. Each multigene with more than one element can be viewed as a task of deciding on the right TIS. Such a decision may have to involve some wet lab experiments or consideration of ESTs or 5' cDNAs [12]. This issue is not discussed in the present paper. So conceptually a multigene corresponds to a gene in which a TIS is yet to be determined (hopefully by selecting one of the listed start sites).

Why does genome alignment not give similar re-

sults as the consolidation graph? The main reason is that in genome alignment one works with sequences which are fragments of genomes without paying any attention to functional genetic elements. In this way one discovers genomic areas of high similarity. Even though postprocessing is often performed, by considering functional genomic elements and the homology relationship between genes or revised genes, gene annotation is not always correctly reconstructed. Moreover, pairwise genome alignment approaches may also miss homologous fragments that can only be linked by intermediate sequences [9]. In contrast, in the consolidation graph we start with annotated genes and close up iteratively with the sequences which come out as significant BLAST hits to the queries already obtained in this analysis.

Connected components of the consolidation graph naturally define sets of multigenes which might be called *multigene families*. This concept of a multigene family is rather new, since in the multigene family construction we did not rely exclusively on given annotations. It turns out that these multigene families can be used to reconstruct a *one-to-one homology* relation for most of the genes. This procedure we call *refinement*. For this we start off with families which consist of at most one multigene from each strain. These we called *anchors*. Then we extend the one-to-one homology relation by considering a genome position of genes, which were not yet related by the one-to-one relationship, with respect to the anchors. This method leaves unresolved only very few small families which presumably should be further curated manually. The one-to-one relationship can be used, among other things, in deciding which multiple alignments should be considered for detection of possible mutations, or even detection of possible sequencing errors.

The methodology above was illustrated with the case study of 9 *Mycobacterium tuberculosis* strains. It is evident from the results presented in this paper that genome annotations done in different labs were not congruent to each other. After performing the consolidation, variance in the total gene count (around 4150 multigenes for all strains) is much smaller than before, suggesting that the revised annotations could lead to a more coherent view of functional elements in various MTB strains. This method can also be applied to completely unannotated genomes, yielding an initial annotation of a newly sequenced genome. The careful reader may have noticed that the same strain (KZN 4207) sequenced in two labs has quite different numbers of annotated genes (3902 vs. 3996); but after consolidation we have for these two genomes almost the same number of multigenes (4140 vs. 4142).

After refinement of the consolidation graph, the number of connected components rose from 4176 to 4288, but size of the largest component dropped from 127 (there are two such components in the consolidation graph) to 15 (only one such component after refinement). Also the maximal number of multigenes in one species and in one non-anchor dropped from 17 in the consolidation graph to 3.

It is interesting to compare the two largest components of the consolidation graph. As mentioned above they have in total 127 multigenes, each strain having between 12 and 17 multigenes in these non-anchors. What is remarkable here is that H37Rv, having 16 multigenes in each of the two components, has all of these 32 genes annotated in the *Tuberculist* database (http://tuberculist.epfl.ch/) as transposons which belong to the same insertion element (IS6110). Even though these two non-anchors were not successfully resolved by the refinement procedure, the resulting non-anchors (four obtained from each of the original two large non-anchors in the consolidation graph) are pretty small: at most two multigenes per strain. More precisely, each of the original non-anchors was split by the refinement procedure into 34 subclusters (4 non-anchors, and 30 anchors with 9 orphans).

The above statistics for the computational experiment on 9 *M. tuberculosis* strains suggest that CAMBer may be a useful utility in comparing and revising annotations of closely related bacterial genomes. With this approach we were also able to discover five cases of gene fusion/fission in the investigated genomes which seems pretty unusual for such closely related species. We leave the analysis of this phenomenon for further study.

As explained in the paper this approach is quite scalable. We plan to test its scalability in future by checking the 61 sequenced *Escherichia coli* and *Schigella spp.* genomes [8] to see whether indeed only 20% of all genes of any strain goes into a *core* genome, i.e. are shared by all strains. This is related to the so-called *distributed-genome hypothesis* (DGH) [6, 8] which states that pathogenic bacteria possess a supragenome that is much larger than the

genome of any single bacterium. Based on 17 sequenced genomes of *Streptococcus pneumoniae* the core genome was estimated [6] as 54% of all the genes. In the case of the nine strains considered in this paper, after the closure procedure we ended up with 3894 multigenes shared by all strains which is 93% of all predicted multigenes.

Input data, software used in the paper (written in Python), and detailed xls files with results of the case study experiment are available at `http://bioputer.mimuw.edu.pl/camber`.

5 Acknowledgments

This work is partially supported by Polish Ministry of Science and Higher Education grant no. N N301 065236 and Singapore Ministry of Education Tier-2 grant MOE2009-T2-2-004.

References

[1] K. Andries, P. Verhasselt, J. Guillemont et al. A diarylquinoline drug active on the ATP synthase of *Mycobacterium tuberculosis*. *Science*, 307(5707):223–227, January 2005.

[2] I. G. Boneca, H. de Reuse, J.-C. Epinat et al. A revised annotation and comparative analysis of *Helicobacter pylori* genomes. *Nucleic Acids Research*, 31(6):1704–1714, March 2003.

[3] R. D. Fleischmann, D. Alland, J. A. Eisen et al. Whole-genome comparison of *Mycobacterium tuberculosis* clinical and laboratory strains. *Journal of Bacteriology*, 184(19):5479–5490, October 2002.

[4] S.T. Cole, R. Brosch, J. Parkhill et al. Deciphering the biology of *Mycobacterium tuberculosis* from the complete genome sequence. *Nature*, 393(6685):537–544, June 1998.

[5] J. Feng, A. Lupien, H. Gingras et al. Genome sequencing of linezolid-resistant *Streptococcus pneumoniae* mutants reveals novel mechanisms of resistance. *Genome Res*, 19(7):1214–1223, July 2009.

[6] N. L. Hiller, B. Janto, J. S. Hogg et al. Comparative genomic analyses of seventeen *Streptococcus pneumoniae* strains: Insights into the pneumococcal supragenome. *Journal of Bacteriology*, 189(22):8186–8195, November 2007.

[7] T. R. Ioerger, S. Koo, E.-G. No et al. Genome analysis of multi- and extensively-drug-resistant tuberculosis from KwaZulu-Natal, South Africa. *PLoS ONE*, 4(11):e7778, November 2009.

[8] O. Lukjancenko, T. M. Wassenaar, and D. W. Ussery. Comparison of 61 sequenced *Escherichia coli* genomes. *Microb Ecol*, in press, July 2010.

[9] J. Park, S. A. Teichmann, T. Hubbard, and C. Chothia. Intermediate sequences increase the detection of homology between sequences. *Journal of Molecular Biology*, 273(1):349–354, October 1997.

[10] S. D. Schlueter, M. D. Wilkerson, E. Huala et al. Community-based gene structure annotation. *Trends Plant Sci*, 10(1):9–14, January 2005.

[11] J. van den Boogaard, G. S Kibiki, E. R Kisanga et al. New drugs against tuberculosis: Problems, progress, and evaluation of agents in clinical development. *Antimicrob Agents Chemother*, 53(3):849–862, March 2009.

[12] H. Wakaguri, Y. Suzuki, M. Sasaki et al. Inconsistencies of genome annotations in apicomplexan parasites revealed by 5'-end-one-pass and full-length sequences of oligo-capped cDNAs. *BMC Genomics*, 10:312, July 2009.

[13] Y. I. Wolf, I. B. Rogozin, A. S. Kondrashov, and E. V. Koonin. Genome alignment, evolution of prokaryotic genome organization and prediction of gene function using genomic context. *Genome Research*, 11(3):356–372, March 2001.

[14] H. Zheng, L. Lu, B. Wang et al. Genetic basis of virulence attenuation revealed by comparative genomic analysis of *Mycobacterium tuberculosis* strain H37Ra versus H37Rv. *PLoS One*, 3(6):e2375, June 2008.