sigB Database Management SOP

FILE NAME: sigB_Database_Management_SOP

Authored by: Steven Warchocki
Last Modified on: 04/10/2014
Approved by: Martin Wiedmann

EFFECTIVE DATE:

APPROVED BY:

_____ _____
Dr. Martin Wiedmann (date)

AUTHORED BY:

_____Steven Warchocki_____ _____09/18/2014_____
(Name) (date)

TABLE OF CONTENTS

SECTION 1 INTRODUCTION

1.1 Purpose

a. To identify novel sigB sequence allelic types (AT) of bacteria isolated in the Food Safety Laboratory (FSL) and Milk Quality Improvement Program (MQIP).

b. To provide detailed instructions for assigning new allelic types, managing, and updating the sigB database.

1.2 Scope

This protocol applies to the Food Safety Laboratory and Milk Quality Improvement Program at Cornell University

1.3 Definitions

AT: allelic type; defined as one specific DNA sequence within a gene, in this case a 660 nucleotide region of the sigB gene in *Listeria*

bp: base pair

BLAST: Basic Local Alignment Sequence Tool

Consensus: a single sequence derived from a set of overlapping DNA segments originating from one genetic source

FMT: Food Microbe Tracker; WWW-based tool for information exchange on bacterial subtypes and strains, containing a large amount of bacterial gene information

PCR: Polymerase Chain Reaction, used to amplify a specific region within a DNA sequence.

Percentage sequence identity: proportion of identical nucleotides between two sequences multiplied by 100.

Phylogeny – The evolutionary history of taxonomic groups.

16S rRNA: 16S ribosomal ribonucleic acid, a RNA component of the 30S small subunit of prokaryotic ribosomes

sigB: sigma factor that coordinates the stress response of *Listeria* species

SECTION 2 MATERIALS

- **Computer with multicore processor**

- **Partial sigB gene sequences:** From PCR products of isolates; sequences are obtained after PCR products are sent to the BRC facility (Biotechnology Research Center). Raw data files are in .ab1 format and edited consensus sequences are saved as .fas files

- **Internet Access:** For accessing Food Microbe Tracker and Ribosomal Database Project

- **Mesquite:** Evolutionary biology software that can be used to make alignments. Can be downloaded at: http://mesquiteproject.org/mesquite/download/download.html. Mesquite is used in Section 3.2 for use in database management and multiple sequence alignments.

- **Muscle:** Comparison algorithm used by Mesquite. It can be downloaded at: http://www.drive5.com/muscle/downloads.htm. It must be imputed into Mesquite during your first use of the program. Muscle is used with Mesquite in Section 3.2 for database management.

- **RaxML:** Computer software used to create phylogenetic trees based on maximum likelihood and bootstrapping. RaxML GUI is the graphical user interface version of RaxML: https://sites.google.com/site/raxmlgui/ or http://sourceforge.net/projects/raxmlgui/

- **sigB NEXUS file:** File that is kept up to date with sigB allelic typing entries, which is subsequently used to create phylogenetic trees, only to be updated by database manager.

- **sigB Access database:** Comprehensive database containing representative ATs, FSL number, genus/species, date added, initials of person who added them, and any relevant notes. This will be managed and updated by Steven Warchocki as of 11/11/2014.

- **FigTree:** Program (http://tree.bio.ed.ac.uk/software/figtree/) used to view phylogenetic tree files, used in conjunction with RaxML

- **BioEdit:** Biological sequence alignment editor that can be used for blasting. http://www.mbio.ncsu.edu/bioedit/bioedit.html

- **USEARCH:** Program used for sequence analysis including blasting. http://www.drive5.com/usearch/

SECTION 3 PROCEDURES

3.1 Confirming New sigB Allelic Types Using Food Microbe Tracker, BioEdit, or USEARCH

Monthly or bi-weekly, collect sigB sequences that have been sent by individuals in the lab working on sigB gene sequencing. [They will have sent an email to the sigB database manager and should have added raw and consensus (final) sequence data to BoorWiedmannLab→LAB STUFF→sigB allelic types→Potential New sigB Allelic Types.].

Confirm the new sigB ATs using one of the three methods below. Users who submitted the potentially new sigB allelic type should have checked to make sure the Single Nucleotide Polymorphism(s) are legitimate in Sequencher. As the database administrator, you have the option to check this as outlined in "New allelic types" in section 3.1.3 below.

3.1.1 Introduction

BioEdit, Food Microbe Tracker, and USEARCH can be used to blast sequences against a given database. Both have their advantages and disadvantages. BioEdit and Food Microbe Tracker are great for blasting a few sequences or checking where SNPs are located in a new sequence type. However, using BioEdit and Food Microbe Tracker is a slower process than USEARCH and results are not exported into an excel file. USEARCH is ideal for blasting a large number of sequences quickly. USEARCH also generates the results in an excel file. Therefore, result information can be copied into a batch upload form for Food Microbe Tracker. Unfortunately, the process of finding new allelic types in USEARCH is more difficult than desired. Additionally, it is not possible to see where SNPs occur because USEARCH does not produce alignments.

3.1.2 Using Food Microbe Tracker:

1. Obtain edited (final) sigB sequence data (*CreatingConsensusDNA.doc*). These sequences should be minimally 660bp in length, but are often longer.
2. Open website for Food Microbe Tracker: http://www.foodmicrobetracker.com
    a. Log-in, or request account for Log-in.
3. In Food Microbe Tracker (FMT), on the left-hand side of the main page, under "Search By", click on "DNA Sequence".
    b. Once on this page use the pull-down menus to adjust your search parameters:
        i. "Number of Results", *default=10*, you may wish to increase/decrease this.
        ii. "Genus", *default=Unspecified*, leave this.
        iii. "Species", *default=Unspecified*, leave this.

iv. "Sequence Type", *default=Unspecified*, this must be changed to "sigB allelic typing" in the pull-down menu.

4. Open the sigB consensus (final) sequence file (.fas) you wish to find an allelic type for. This can be done in either Notepad or Sequencher.
5. Cut and paste your sigB sequence into the space labeled "Enter DNA sequence".
6. Click Submit.
   a. Once your results page ("Search Results from DNA Sequence Search") has loaded choose on the first Alignment file by clicking on "See Report" in red.
7. When new page/tab has appeared click to view it and review some key details:
   b. "Identities", this should read 660/660 (100%) for a 100% allelic type match, and less than 100% for unique AT sequences.
   c. "Sbjct", which is the sigB allelic typing sequence your entered "Query" is being compared to, should start at 1 and end at 660.
   d. Less than a 100% match (out of 660bp) for "Identities" indicates a potential new sigB allelic type.

3.1.3 Using BioEdit

Uploading a local nucleotide database file: The Food Safety Laboratory and Milk Quality Improvement Program have local sigB allelic type databases. Every time one of these databases is updated, it also needs to be updated in Bioedit before blasting any sequences. To update database in BioEdit, go to Select Accessory Application→BLAST→Create a local nucleotide database file→find the most recent sigB database file (make sure it is in fasta format) → select the most current sigBhaplotypes file (e.g. sigBhaplotypes08062010.fas) →Open.

Blasting sigB sequences that are in FASTA – concatenated format: Select *File→Open…*→find your FASTA – concatenated file containing your final sequences →*Open→Edit→Select All Sequences→Accessory Application→BLAST→Local BLAST*→select *Yes* when prompted to do a batch job→select the most current sigB file from the *Nulceotide Database* drop-down menu→*1* for *Max number of hits to report→1* for *Max number of alignments to show→Do Search*. The window that appears contains all your sequences in the appropriate order. The relevant information under each isolate or query is as follows: the *Database* that was blasted against (should be the most current sigB file), the best allelic type match with the corresponding genus/species/lineage (e.g. *AT_60_LmonocytogenesI*), *Identities* (if your isolate is a perfect match with an existing allelic type, this will read *660/660*), an alignment between your isolate and the best allelic type match (e.g. *AT_60_LmonocytogenesI*). Record the allelic type and genus/species/lineage of your isolate. Scroll down to view the next isolate. If *Identities* are not out of 660 (e.g. 660/660), there was an editing error (e.g. too much was trimmed off one end of the sequence) or the sequencing reaction was not sufficient to produce clean sequence of

adequate length. If the sequence cannot be re-edited to produce a sequence out of 660, it will have to be re-sequenced.

Blasting sequences that are in FASTA form: This is identical to Blasting sequences that are in FASTA – concatenated format with a few exceptions. If you try to open multiple FASTA files at once within Bioedit, they will all open in their own window which is one reason to use FASTA – concatenated files for blasting multiple sequences at once. Additionally, since there is only one isolate in the FASTA file unlike the multiple isolates within the FASTA – concatenated file, the BLAST will only return one search within the BLAST window. Also, you will not be prompted to do a batch job.

New allelic types: If your isolate is not a perfect match with the closest sigB allelic type match, *Identities* will read 659/660, 656/660, etc. This may indicate a new sigB allelic type. To legitimize a new sigB allelic type, view the alignment between your isolate (*Query*) and the best match (*Sbjct*). Where there is a single nucleotide polymorphism (SNP), a line will be missing between the *Query* and *Sbjct* in the alignment. Copy the *Query* sequence around the SNP (copy roughly 25 bases) and remember where the SNP resides within the copied sequence. Open the contig file and chromatogram for that isolate using Sequencher. After opening select *Select*→*Find Bases…*→paste the copied sequence in the text field→*Exact Matches*→*Find*→observe the chromatogram at the location of the SNP to make sure the peaks are clean and that the SNP is not a result of an editing error. Repeat for every other SNP if there is more than one. If SNP(s) are legitimate, export the new allelic type from Sequencher as its own .fas file. Add to alignment in Mesquite (see below).

Do not save any BLAST searches upon closing Bioedit.

3.1.4 Using USEARCH

USEARCH, as with BioEdit, is a free sequence analysis program. It does not have a graphical user interface. Therefore, you will need to use the command prompt. Sufficient instructions are provided here to use USEARCH with the command prompt.

1. Download USEARCH and move the program file to a permanent location.

2. Open the command prompt by typing *cmd* in the start menu search field

3. In the command prompt, the current directory will be showing (e.g. C:\Users\skw59>). To change directories type *cd* followed by a space and then the directory you want to go to (e.g. C:\Users\skw59>cd Desktop). Hit enter and you will be taken to that directory (e.g. C:\Users\skw59\Desktop>). Keep on going until you find the directory USEARCH is in.

4. The following is what you might enter next: *"c:\Program Files (x86)\usearch.exe" -search_global LauraNewSigB09092014forusearch.fas -id 1 -db sigBhaplotypes02152011.fas -strand both -maxaccepts 1 -blast6out sigB.out* (don't add period at end of this). The program you are running and its location is in between the quotation marks. *–search_global* is the command which blasts a file against a database. The file you are blasting, *LauraNewSigB09092014forusearch.fas*, follows the *–search_global* command. *–id* refers to the identity threshold. It is set to 1 so that only 100% matches are shown in the output file. *-db* specifies the database being blasted against. *–strand both* will check both the forward and its reverse complemented strand. *-maxaccepts* is the number of hits each sequence will receive. This is set to 1 so only the closest match is seen. *-blast6out* indicates the type of output file and *sigB.out* is the name of the output file. It is also important to note that the query file (e.g. *LauraNewSigB09092014forusearch.fas)* and database file (e.g. *sigBhaplotypes02152011.fas)* need to be in the same folder (e.g. Desktop). The pathway all together is as follows:

   C:\Users\skw59\Desktop>*"c:\Program Files (x86)\usearch.exe" -search_global LauraNewSigB09092014forusearch.fas -id 1 -db sigBhaplotypes02152011.fas -strand both -maxaccepts 1 -blast6out sigB.out*

   The information for the blasting criteria described above can be found on the USEARCH website.

5. *-blast6out* produces a tab-separated text file that can be opened in excel. All the output fields for *-blast6out* are below but can also be found on the USEARCH website:

Field	Description
1	Query label.
2	Target (database sequence or cluster centroid) label.
3	Percent identity.
4	Alignment length.
5	Number of mismatches.
6	Number of gap opens.
7	1-based position of start in query. For translated searches (nucleotide queries, protein targets), query start<end for +ve frame and start>end for -ve frame.
8	1-based position of end in query.
9	1-based position of start in target. For untranslated nucleotide searches, target start<end for plus strand, start>end for minus strand.
10	1-based position of end in target.
11	E-value calculated using Karlin-Altschul statistics.
12	Bit score calculated using Karlin-Altschul statistics.

6. To determine if there are any new allelic types, run the program with *–id* .97 (you can use .98 or .96). Give the output file a different name. This will report all queries that are a 97% match or better to an allelic type already in the database. Take the Queries reported from the .97 file and compare them to the original file with only 100% matches. Use sorting and formatting options to determine which queries are missing from the 100% file that are in the 97%. These are the new allelic types.

7. Use BioEdit or Food Microbe Tracker to confirm the new ATs and determine if the SNPs are legitimate (see 3.1.3 above).

3.2 Using Mesquite, RAxMLGUI, and Fig Tree to Further Confirm Novelty of and Determine Clade Clustering of New sigB Allelic Types

After new sigB allelic types have been confirmed using Food Microbe Tracker, BioEdit, or USEARCH, further analysis is needed to confirm their novelty, determine clade clustering, and genus/species.

3.2.1 – Assembling Alignments Using Mesquite

Mesquite is free software for PCs, Macs, and Linux systems. Use *Mesquite* to make an alignment of old and new sigB sequences. If downloading Mesquite 3.0, use the 2 GB version. On some computers, an error prevents the user from running Mesquite 2GB. If this occurs, run the 1GB version. After downloading *Mesquite,* download the algorithm Muscle. Remember where the muscle file lives. *Muscle* contains the set of commands that dictates how the alignment is assembled.

A. After downloading *Mesquite*, open the program and then open the existing sigB database file from within the program. The default file format for *Mesquite* is nexus. However, you can just as easily import other file types, including fasta, and the program will prompt you to save in nexus format upon import.

B. Select *Show Matrix*. Taxa will be at the left of the *Character Matrix* and accompanying sequence to the right. Additional sigB files can be added to the *Character Matrix* by drag and drop or by selecting *File Incorporation* → *Merge Taxa & Matrices* → select file type → *Fuse with Selected Taxa Block* → *Fuse with Selected Matrix*. If using drag and drop, drop new sigB file(s) at the bottom of the Character Matrix. This keeps any new Allelic Types (ATs) in sequential order even though taxa can be moved around.

C. To create alignment select *Edit* → *Select all* → *Matrix* → *Align Multiple Sequences* → *Muscle Align* → do not run on separate thread → locate muscle file → *ok* . The process isn't instantaneous so there will be some waiting for the alignment to finish.

D. Scroll through the alignment and check new sequences for deletions or insertions. If any of these are present, they will need to be confirmed in the chromatogram (can be viewed in *Sequencher*) before moving forward. Trim off excess sequence from each end of the alignment by holding down shift and clicking the outer blocks of the area to be deleted (Numbers at the top of the alignment can be selected to trim multiple sequences at one time). Once selected, select *Edit → Cut* to delete sequence.

E. Save new alignment with appropriate name (e.g. sigBhaplotypes05072014) and add the .nex extension to the file name. This alignment (nexus format) can be used for tree construction in *Paup*. For use in *RaxML*, export alignment in Phylip format.

3.2.2 Constructing Phylogenetic Trees Using RAxMLGUI

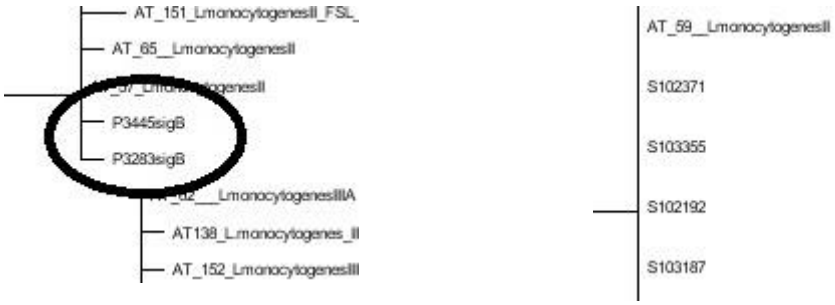
A. Download *raxmlGUI* to construct a phylogenetic tree. For users more familiar with the command prompt, *RAxML* can be installed and used instead. *raxmlGUI* also requires that *Python 2.5-2.7* be installed. *raxmlGUI* is not compatible with Python 3.0 (as of September 2014). Open *raxmlGUI* and *load alignment* that was exported from Mesquite in Phylip format. RaxMLGUI will display a message 'RAxML found at least 1 sequence that is exactly identical to other sequences and/or gap-only characters in the alignment. Do you want to exclude it/them from the analysis?', select 'No'. Make sure *ML + rapid bootstrap* is selected and set *reps.* to a minimum of 100. All other parameters can be left at the default. Select *Run RAxML*. The run can take a while so it is best to work on something else during this time.

B. *RaxmlGUI* will generate multiple output files in the folder/directory the alignment was loaded from: (1) the best-scoring ML tree 'RAxML_bestTree.YOUR_FILE_NAME.tre, (2) Best-scoring ML tree with bootstrap support values 'RAxML_bipartitions.YOUR_FILE_NAME.tre, (3) Best-scoring ML tree with bootstrap support values as branch labels 'RAxML_bipartitionsBranchLabels. YOUR_FILE_NAME.tre', (4) Program execution info 'RAxML_info. YOUR_FILE_NAME.tre', (5) All 100 bootstrapped trees 'RAxML_bootstrap. YOUR_FILE_NAME.tre, (6) a phylip formatted file with all unique sequences 'YOUR_FILE_NAME.reduced', and (7) a file listing which sequences are identical to other sequences in the original file 'RAxML_info". The last two files are important for further analyses. The ".reduced" file will contain only unique sigB sequences. Therefore, this file will only contain existing representative ATs and any representative new ATs. The "reduced" and "info" files can be used to identify a representative new allelic type and determine identical sequences.

C. To view the tree created by *RAxML*, open the file named *RAxML_bipartitions.filename* in *Fig Tree* or another tree viewing program. Type *bootstrap* in the text field when prompted to select a name for the node/branches. If *L. grayi* was kept in the tree, an

optional step is to select the *L. grayi* branch and reroot the tree. *L. grayi* is used as the outgroup here. Important: Unless one of the new ATs is a potential *L. grayi*, it is usually best to keep *L. grayi* out of the tree because it can create tree abnormalities.

D. Zoom in to find new isolates in the tree and observe what species they cluster with. It can be determined whether any of your isolates are duplicates as in the "RAxML_info" file. These isolates will be next to one another on the same branch. The two isolates in the left picture below are right next to each other but are different ATs because each isolate has its own branch. The picture on the right shows an example of multiple identical isolates. Be careful when viewing the tree. If a group of isolates share the same vertical branch as depicted in the right picture, but are separated by nodes, they are still identical to each other.



3.3 Updating Database Files/Records

1. After new sigB allelic types have been confirmed in *Fig Tree* and their genus/species determined, taxa need to be updated in the master Nexus file using *Mesquite*. This will include renaming taxa with AT #, genus/species, and FSL #. Double click on each "Taxa" name, and add "ATXXX_FSL_ID", adding new AT numbers in consecutive order, FSL ID should remain. New sigB ATs numbers will be assigned chronologically (by date of discovery). New duplicate ATs should also be removed from the alignment by highlighting taxon/character → *Edit* → *Cut*.

2. The new Nexus database file (e.g. sigBhaplotypes05072014.nex) should be saved and put in BoorWiedmannLab → LAB STUFF → sigB allelic types. The other file in the folder (which should now be older) can then be moved into the folder "old files – do not use".

3. Add new AT sequence(s) to Food Microbe Tracker (FMT) → open FMT → go to the given isolate's page → *Add* a *DNA Sequence* → select *sigB allelic typing* from drop down → paste in trimmed sequence or upload fasta file of final trimmed sequence → *submit*. *Add* AT # under *Additional Characteristics*.

4. Add new ATs to sigB Access Database (maintained by database manager). Information to add includes the date, manager's net i.d., AT range added, and specific ATs with associated species.

SECTION 4 TROUBLESHOOTING

4.1 If "Identities" in your results read anything but out of 660 (e.g. 658/658 or 655/659), the BLAST algorithm has somehow trimmed your sequence. First check the second best match, if that one isn't out of 660 either, then it is best to pull out sigB sequences from each isolate's FMT page and align them (ClustalW or Mesquite can do this) and see if the complete length matches for 660bp.

SECTION 5 REFERENCES

Edgar, R.C. (2004) MUSCLE: multiple sequence alignment with high accuracy and high throughput *Nucleic Acids Res.* 32(5):1792-1797

A. Stamatakis, F. Blagojevic, C.D. Antonopoulos, D.S. Nikolopoulos: (2007) Exploring new Search Algorithms and Hardware for Phylogenetics: RAxML meets the IBM Cell *Journal ofVLSI Signal Processing Systems* 48(3):271-286