10

Analyzing Patterns of Microbial Evolution Using the Mauve Genome Alignment System

Aaron E. Darling, Todd J. Treangen, Xavier Messeguer, and Nicole T. Perna

Summary

During the course of evolution, genomes can undergo large-scale mutation events such as rearrangement and lateral transfer. Such mutations can result in significant variations in gene order and gene content among otherwise closely related organisms. The Mauve genome alignment system can successfully identify such rearrangement and lateral transfer events in comparisons of multiple microbial genomes even under high levels of recombination. This chapter outlines the main features of Mauve and provides examples that describe how to use Mauve to conduct a rigorous multiple genome comparison and study evolutionary patterns.

Key Words: Microbial evolution; sequence alignment; comparative genomics; genome alignment; genome rearrangement; lateral transfer; *Yersinia pestis*.

1. Introduction

As genomes evolve, mutational forces and selective pressures introduce rearrangements via inversion, transposition, and duplication/loss processes. Lateral transfer can introduce novel gene content, or in the case of homologous recombination, introduce more subtle changes such as allelic substitution. Through genome comparison we hope to first identify differences among organisms at the genome level and then infer the biological significance of differences and similarities among related organisms.

From: *Methods in Molecular Biology, vol. 396: Comparative Genomics, Volume 2*
Edited by: N. H. Bergman © Humana Press Inc., Totowa, NJ

Traditional sequence alignment methods were developed to accurately align individual gene sequences where rearrangement rarely occurs ***(1–6)***. When aligning two or more genomes, shuffled regions of orthologous sequence may be interspersed with paralogous and novel sequence regions, forcing a genome alignment method to map segmental homology among genomes ***(7–14)***. Once a segmental homology map exists, the alignment task can be approached using traditional alignment methods based on dynamic programming and pair-hidden Markov model heuristics ***(15)***.

In genome alignment, it is important to go beyond simple classification of sequence regions as either homologous or unrelated. Local alignment programs such as Basic Local Alignment Search Tool provide such functionality. We define a global genome alignment to be a complete catalog of orthologous, xenologous ***(16)***, paralogous, and unrelated sites among a group of genome sequences. All sites must be assigned to one of the given categories, and in the cases of orthology, paralogy, and xenology, the related sites must be identified. Furthermore, homologous sites should be grouped into regions of maximal collinearity such that the leftmost and rightmost site in any group defines a breakpoint of rearrangement. We refer to such maximal collinear sets of homologous sites as locally collinear blocks (LCBs) because they cover a "block" of sequence without any internal genome rearrangement. Such a categorization of sites implicitly defines breakpoints of rearrangement, recombination, duplication, and insertion and deletion processes. A genome alignment lends itself to downstream evolutionary inferences such as rearrangement history ***(17–21)***, phylogeny ***(22)***, ancestral state prediction, and detection of selective pressure in coding sequence ***(23)*** and in noncoding sequence ***(24)***.

Most current genome alignment systems, including Mauve ***(25)***, construct an incomplete form of genome alignment as defined above. Shuffle-LAGAN ***(26)*** aligns both single-copy and repetitive regions in pairs of genomes, but does not classify them as either orthologous or paralogous. Mauve aligns orthologous and xenologous regions, but does not distinguish between the two cases. Mauve also aligns orthologous repeats, but does not align paralogous repeats. Mulan ***(27)*** and M-GCAT ***(28)*** both construct alignments similar to Mauve. Recent versions of MUMmer ***(29)*** identify and align orthologous and paralogous sequence, but among pairs of genomes only. Some earlier genome alignment systems have treated segmental homology mapping as a separate step and thus assume sequence collinearity ***(30–33)***.

Mauve performs five basic steps when constructing a genome alignment:

1. Search for local multiple alignments (approximate multi-MUMs).
2. Calculate a phylogenetic guide tree using the local alignments.

3. Select a subset of the local alignments to use as anchors.
4. Conduct recursive anchoring to identify additional alignment anchors.
5. Perform a progressive alignment of each LCB.

The METHODS section describes each of these five steps in greater detail.

Before performing an analysis with Mauve, a researcher should first ask whether genome alignment is the right analysis. If the answer is "yes," the next question should be "Is Mauve the right tool for the job?" Mauve performs best when aligning a relatively small number of closely related genomes. It can align genomes that have undergone rearrangement and lateral transfer. However, as mentioned previously, the current version of Mauve constructs only an incomplete form of genome alignment. Mauve genome alignments alone do not provide a suitable basis for inferences on paralogous gene families. Mauve is also limited in its ability to align rearranged and large (e.g., > 10 Kbp) collinear segments that exist in only a subset of the genomes under study. And, in general, the level of nucleotide similarity among all taxa should be greater than 60%. **Subheading 3.** explains the reasons for these limitations in more detail and gives hints on choosing alignment parameters so as to mitigate any potential problems.

2. Materials

1. Windows, Mac OS X 10.3+, or Linux Operating System.
2. Mauve Multiple Genome Alignment software (http://gel.ahabs.wisc.edu/mauve/download.php).
3. Four *Yersinia* genomes in GenBank format:
   a. *Yersinia pestis* KIM, accession number: **AE009952** (http://www.ncbi.nlm.nih.gov/entrez/viewer.fcgi?db=nucleotide&val=AE009952).
   b. *Y. pestis* CO92, accession number: **AL590842** (http://www.ncbi.nlm.nih.gov/entrez/viewer.fcgi?db=nucleotide&val=AL590842).
   c. *Y. pestis* 91001, accession number: **AE017042** (http://www.ncbi.nlm.nih.gov/entrez/viewer.fcgi??db=nucleotide&val=AE017042).
   d. *Yersinia pseudotuberculosis* IP32953, accession number: **BX936398** (http://www.ncbi.nlm.nih.gov/entrez/viewer.fcgi?db=nucleotide&val=BX936398).
4. Two drosophila genomes:
   a. *Drosophila melanogaster* assembly 4.0 in GenBank format: (http://gel.ahabs.wisc.edu/mauve/chapter/dmel.gbk).
   b. *Drosophila yakuba* assembly 2.1 in FastA format: (http://gel.ahabs.wisc.edu/mauve/chapter/dyak.fas).
5. Example output files located at http://gel.ahabs.wisc.edu/mauve/chapter/.

3. Methods

The following sections provide a deeper look into how the five main steps in Mauve's alignment algorithm contribute to the global genome alignment process.

3.1. Search for Local Multiple Alignments (Approximate Multi-MUMs)

Mauve uses a seed-and-extend hashing method to simultaneously identify highly similar unique regions among all genomes. The method used in current Mauve releases remains similar to that described in ***(14,25)*** but has been extended to approximate matching using spaced seeds ***(34)***. In addition to finding matching regions that exist in all genomes, the algorithm identifies matches that exist only among a subset of the genomes being aligned. The local multiple alignment method is very efficient in practice and typically requires less than 1 min per bacterial-size genome to find local alignments, and around 3 h for each mammalian genome on a standard workstation computer.

3.2. Calculate a Phylogenetic Guide Tree Using Local Alignments

As part of the alignment process, Mauve calculates a guide tree via genome distance phylogeny ***(35)***. Rather than using pairwise BLAST hits to estimate distance, Mauve uses the local multiple alignments identified in the previous step. The average amount of novel sequence among each pair of genomes is calculated using the local multiple alignments. This value is used as a pairwise distance measure to construct a phylogenetic tree via Neighbor Joining ***(36)***.

3.3. Selecting a Set of Anchors

In addition to local multiple alignments that are part of truly homologous regions, the set of approximate multi-MUMs may contain spurious matches arising as a result of random sequence similarity. This step attempts to filter out such spurious matches while determining the boundaries of LCBs. Local alignments are clustered together into LCBs and each LCB is required to meet a minimum weight criteria, calculated as the sum of lengths of its constituent local alignments. Local alignments contained in LCBs that do not meet the weight criteria are deleted.

3.4. Recursive Anchoring and Gapped Alignment

The initial anchoring step may not be sensitive enough to detect the full region of homology within and surrounding the LCBs. Using the existing

anchors as a guide, two types of recursive anchoring are performed repeatedly. First, regions outside of LCBs are searched to extend the boundaries of existing LCBs and identify new LCBs. Second, unanchored regions within LCBs are searched for additional alignment anchors.

3.5. Progressive Alignment

After gathering a complete set of alignment anchors among all genomes, Mauve then performs either a MUSCLE or a CLUSTAL W progressive alignment. CLUSTAL alignments use the previously calculated genome guide tree. The progressive alignment algorithm is executed once for each pair of adjacent anchors in every LCB, calculating a global alignment over each LCB. Tandem repeats <10 Kbp in total length are aligned during this phase. Regions >10 Kbp without an anchor are ignored. For additional details and a more in-depth algorithmic analysis refer to **ref**. ***(25)***.

3.6. Understanding the Alignment Parameters

To accurately align a set of genomes using the aforementioned steps, it can be helpful to carefully tailor the provided Mauve alignment parameters to better suit the characteristics of the genomes being compared. The following sections provide a detailed explanation of each configurable alignment parameter.

3.6.1. Match Seed Size

The seed size parameter sets the minimum length of local multiple alignments used during the first pass of anchoring the alignment. When aligning divergent genomes or aligning more genomes simultaneously, lower seed sizes may provide better sensitivity. However, because Mauve also requires the matching seeds to be unique in each genome, setting this value too low will reduce sensitivity (small *k*-mers are less likely to be unique).

3.6.2. Default Seed Size

Setting this option will allow Mauve to automatically select an initial match seed size that is appropriate for the length of sequences being aligned. The default seed size for 1 MB genomes is typically around 11, around 15 for 5 MB genomes, and continues to grow with the size of the genomes being aligned up to 21 for mammalian genomes. The defaults may be conservative (too large), especially when aligning more divergent genomes (*see* **Note 1** for suggestions).

3.6.3. LCB Weight

The LCB weight sets the minimum number of matching nucleotides identified in a collinear region for that region to be considered true homology vs random similarity. Mauve uses a greedy breakpoint elimination algorithm to compute a set of LCBs that have the given minimum weight. By default an LCB weight of three times the seed size will be used. For many genome comparisons the default LCB weight is too low, and a higher value will be desired. The ideal LCB weight can be determined interactively in the alignment display using the LCB weight slider.

3.6.4. Determine LCBs

If this option is disabled Mauve will identify local multiple alignments, but will not cluster them into LCBs. See the description of match generation in the command-line interface chapter.

3.6.5. Assume Collinear Genomes

Select this option if it is certain that there are no rearrangements among the genomes to be aligned. Using this option when aligning collinear genomes can result in improved alignment accuracy.

3.6.6. Island and Backbone Sizes

An island is a region of the alignment where one genome has a sequence element that one or more others lack. This parameter sets the alignment gap sizes used to calculate islands and backbone segments.

3.6.7. Full Alignment

Selecting the "Full alignment" option causes Mauve to perform a recursive anchor search and a full gapped alignment of the genome sequences using either the MUSCLE or the ClustalW progressive alignment method. Disabling this option will allow Mauve to rapidly generate a homology map without the time consuming gapped-alignment process (*see* **Note 2** for performance tips for full alignments).

3.7. Examples

3.7.1. A Detailed Example Using Four Yersinia Genomes

The *Yersinia* genus is responsible for diseases as common as gastroenteritis and as infamous as the plague. Of the 11 known *Yersinia* species, *Y. pestis*

became the most notorious when identified as the cause of the bubonic and pneumonic plague *(37)*. *Y. pestis* can be further classified into three main biovars according to three historically recognized pandemics: Antiqua (~1000 AD), Medievalis (1300–1800), and Orientalis (1900–Present). Additionally, it is believed that *Y. pestis* is a clone that has evolved from *Y. pseudotuberculosis* as recent as 1500 yr ago. The recent and rapid evolution of *Yersinia* species provides an excellent example for study with Mauve. In this example we will align and analyze the four currently finished genomes of *Yersinia*: *Y. pseudotuberculosis* IP32953, *Y. pestis* KIM, *Y. pestis* 910001, and *Y. pestis* CO92.

3.7.2. Running Mauve

Under Windows, Mauve can be launched directly from the Start Menu. On Mac OS X, Mauve is distributed as a stand-alone application and can be run from any location. On Linux and other Unix variants, simply run. `/Mauve` from within the Mauve directory to start the Mauve Java GUI (*see* **Note 3** for further tips on configuring Mauve to avoid Java heap space limitations).

3.7.3. Locating the Input Data

To align these four *Yersinia* genomes, we first need to download the four input genome sequence files (as listed in **Subheading 2.**) and load them into Mauve. Mauve accepts the following genome sequence file formats: FastA, Multi-FastA, GenBank flat file, and raw format. FastA and GenBank format files with the genome of your organism can usually be downloaded from NCBI at ftp://ftp.ncbi.nih.gov/genomes/. The .fna files are in FastA format and the .gbk files are in GenBank format; for this example we will use the .gbk files.

3.7.4. Loading the Input Data

Once Mauve has started up, simply select "Align..." from the "File" menu to access the Alignment dialog box, as shown in **Fig. 1**:

The top entry area lists the sequence file(s) containing the genomes that will be aligned. To add a sequence file, click the "Add sequence..." button and select the file to add. The Windows and Mac OS X versions of Mauve support drag-and-drop, allowing sequence files to be added by dragging them in from the Windows Explorer or Mac OS Finder.

3.7.5. Configuring the Output Data Location

The location where Mauve stores its alignment results can be set using the "Output file:" text entry field. By default Mauve creates output files in the system's

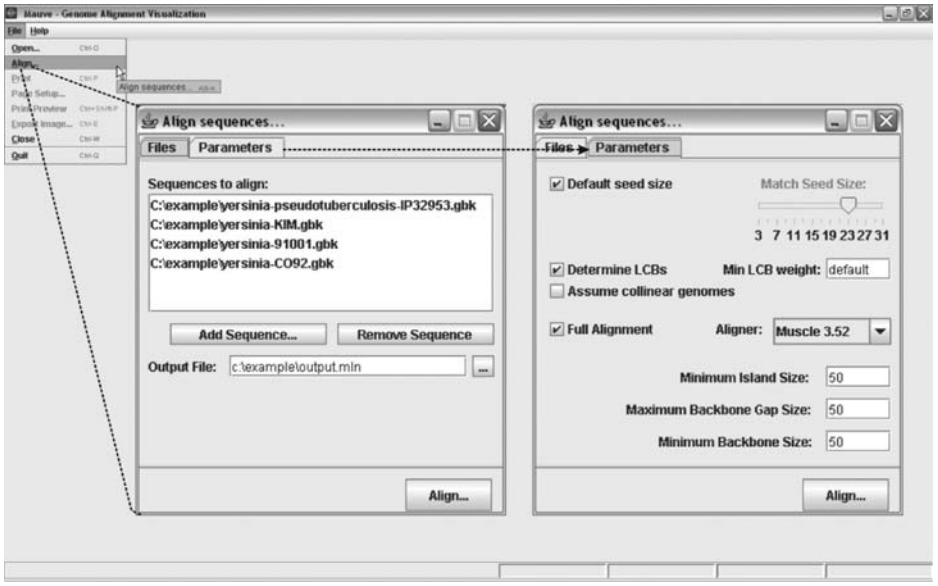

Fig. 1. The sequence alignment dialog. Selecting the "Align…" option from the "File…" menu will display the "Align sequences" dialog. The "Files" panel allows for selection of files containing genome sequences to be aligned, and when switching to the "Parameters" panel, configurable alignment parameters are displayed.

temporary storage directory. Under Windows this is usually `C:\Documents and Settings\<username>\Local Settings\Temp\` or `/tmp` under Mac OS X and Unix. Because the location of the system's temporary storage is rather obscure, we heartily encourage users to select a different output location. In this example, we specify `C:\example\output.mln` to store the Mauve output.

3.7.6. Configuring the Alignment Parameters

By default, Mauve configures the alignment parameters so they are appropriate for aligning closely related genomes with moderate to high amounts of genome rearrangement. However, some alignment parameters can be modified to better suit alignments involving more distantly related genomes, or with a lower amount of rearrangement. For example, when aligning more divergent genomes, the seed size can be reduced to find additional alignment anchors and achieve greater alignment coverage over the genomes. Another option

disables the full alignment process, allowing Mauve to quickly generate a simple comparative picture of genome organization. For this example, we start with the default parameters, and later use the interactive LCB weight slider (*see* **Fig. 2**) to determine an LCB weight that excludes most spurious rearrangements (1500). We then recomputed the alignment with the LCB weight set to 1564.

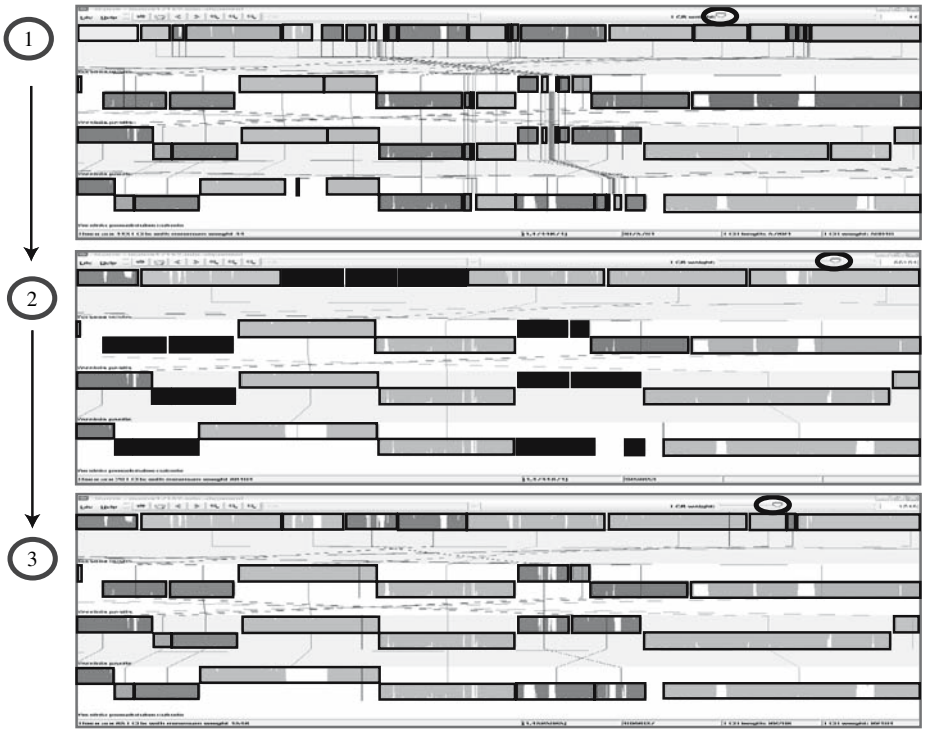

Fig. 2. Interactively determining the ideal locally collinear block (LCB) weight. Starting with an alignment based on the default LCB weight parameter **(1)**, we move the interactive LCB weight slider to the right **(2)** to exclude spurious rearrangements. The LCB weight of 66,104 in **2** is too large as it deletes many large segmental homologies that appear to be valid—shown as dark gray blocks in panel **2**, or as bright orange in the Mauve viewer. We thus move the interactive LCB weight slider to the left again **(3)** to arrive at a weight of 1564 that excludes most spurious small matches, and retains all valid large matches. We can now recompute a full alignment with the weight set to 1564.

3.7.7. The Interactive Mauve Alignment Viewer

To start the genome alignment, click on the "Align..." button. Once Mauve finishes its global alignment of the four *Yersinia* genomes, we are ready to interactively inspect the results. Should the alignment fail, please check the console log (*see* **Note 4**). The Mauve Alignment viewer enables manual evaluation of both the proposed global homology and the nucleotide level alignment in the context of genome annotation. *See* **Note 5** for details on manual editing of the genome alignment. To further analyze the *Yersinia* alignment results, it is important to understand the design and functions of the viewer.

3.7.7.1. Alignment Viewer Description

The alignment display is organized into one horizontal "panel" per input genome sequence. Each genome's panel contains the name of the genome sequence and a scale showing sequence coordinates for that genome. Additionally, each genome panel contains one or more colored block outlines that surround a region of the genome sequence that has been aligned to part of another genome. Some blocks may be shifted downward relative to others; such blocks are in the reverse complement (inverse) orientation relative the reference genome. Regions outside blocks were too divergent in at least one genome to be aligned successfully. Inside each block Mauve draws a similarity profile of the genome sequence. The height of the similarity profile corresponds to the average level of conservation in that region of the genome sequence. Areas that are completely white were not aligned, presumably because they contain lineage specific sequence. The height of the similarity profile is calculated to be inversely proportional to the average alignment column entropy over a region of the alignment.

In **Fig. 1**, colored blocks in the first genome are connected by lines to similarly colored blocks in the remaining genomes. These lines indicate which regions in each genome are homologous. If many genomic rearrangements exist, these lines may become overwhelming (*see* **Note 6**). The boundaries of colored blocks indicate the breakpoints of genome rearrangement.

3.7.7.1.1. Navigation The alignment display is interactive, providing the ability to zoom in on particular regions and shift the display to the left and right. Navigating through the alignment visualization can be accomplished by using the control buttons on the toolbar immediately above the display. Alternatively, keyboard shortcuts allow rapid movement through the alignment display. The keystrokes `Ctrl+up arrow` and `Ctrl+down arrow` zoom the display in and out, whereas `Ctrl+left arrow` and `Ctrl+right arrow` shift left and right, respectively. When moving the mouse over the alignment display

Mauve will highlight the aligned regions of each genome with a black vertical bar, and clicking the mouse will vertically align the display on the selected orthologous site.

3.7.7.2. Viewing Annotated Features

If the aligned genome sequences were in GenBank files containing annotated features Mauve will display the annotated features immediately below the sequence similarity profiles. Annotated CDS features show up as white boxes, tRNAs are green, rRNAs are red, and misc_RNA features are blue. Features annotated on the reverse strand are on a track immediately below the forward strand track. Repeat_region features are displayed in red on a third annotation track. Mauve displays the product qualifier when the mouse cursor is held over a feature. When a feature is clicked, Mauve shows a detailed listing of feature qualifiers in a popup window. For performance reasons, the annotated sequence features appear only when the display has been zoomed in to view less than 1 Mbp of genome sequence.

3.7.8. Analyzing/Interpreting the Results

Now that the viewer has been explained in detail, we can use it to analyze the results from the *Yersinia* genome alignment. These four *Yersinia* genomes have a rich and well-studied evolutionary history ***(38–40)***. In **ref**. ***40*** it was reported that transmission by fleabite is a recent evolutionary adaptation that distinguishes *Y. pestis*, the agent of plague, from *Y. pseudotuberculosis* and all other enteric bacteria. The high level of sequence similarity between *Y. pestis* and *Y. pseudotuberculosis* implies that only a few minor genetic changes were needed to induce flea-borne transmission. The question is thus; can we identify these changes using the Mauve genome alignment system? From the global view presented in **Fig. 3** it is difficult to see individual nucleotide substitutions and deletions, so we need to exploit some of the advanced features of the interactive viewer, such as the zoom and gene annotation. **Figure 4** shows a zoomed in view of the one gene responsible for adhesion to the gut of host organisms. The variability in conservation of this gene could contribute to the differences in pathogenicity between *Y. pestis* and *Y. pseudotuberculosis* reported in **ref**. ***(40)***. Additionally, using the Mauve alignment viewer we can see that the plasmid *pMT1*, which mediates infection of the plague flea vector, is present in all of the *Y. pestis* genomes, but not in *Y. pseudotuberculosis*.

A number of recent studies indicate that microbial genomes evolve and adapt by integrating novel genetic elements through lateral transfer ***(41–43)***. Such

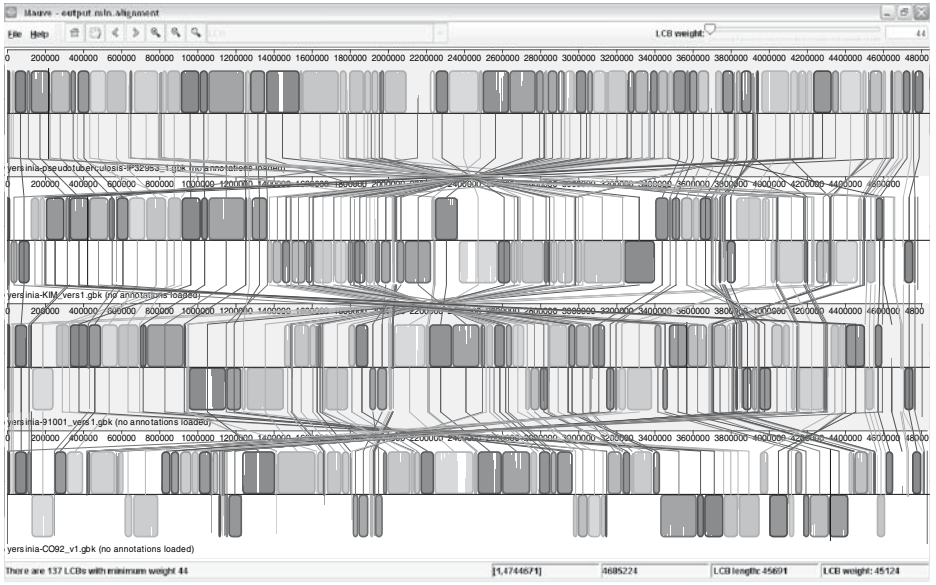

Fig. 3. A Mauve alignment of *Yersinia pseudotuberculosis* IP32953, *Yersinia pestis* KIM, *Y. pestis* 910001, and *Y. pestis* CO92. Notice how inverted regions among the four genomes are clearly depicted as blocks below a genome's center line. The crossing locally collinear block connecting lines give an initial look into the complicated rearrangement landscape among these four related genomes.

novel regions are commonly referred to as genomic islands (GIs) and appear as large gaps in the genome sequence alignment. Mauve identifies large alignment gaps that correspond to putative islands and saves their sequence coordinates in a ".islands" file. Similarly, regions conserved among all genomes are frequently referred to as backbone, and appear as large regions of the alignment that contain only small gaps. Mauve saves the sequence coordinates of backbone segments in a ".backbone" file. Despite their recent speciation, *Y. pestis* and *Y. pseudotuberculosis* contain a number of GIs.

3.7.9. Using the Mauve Command-Line Tool to Align Large Genomes: D. melanogaster *and* D. yakuba

In contrast to the previous example, where each genome contained around 5 million nucleotides, the two *Drosophila* genomes involved in this comparison are each over 100 million nucleotides in size. Although Mauve can efficiently analyze smaller microbial genome comparisons in memory, genomes as large

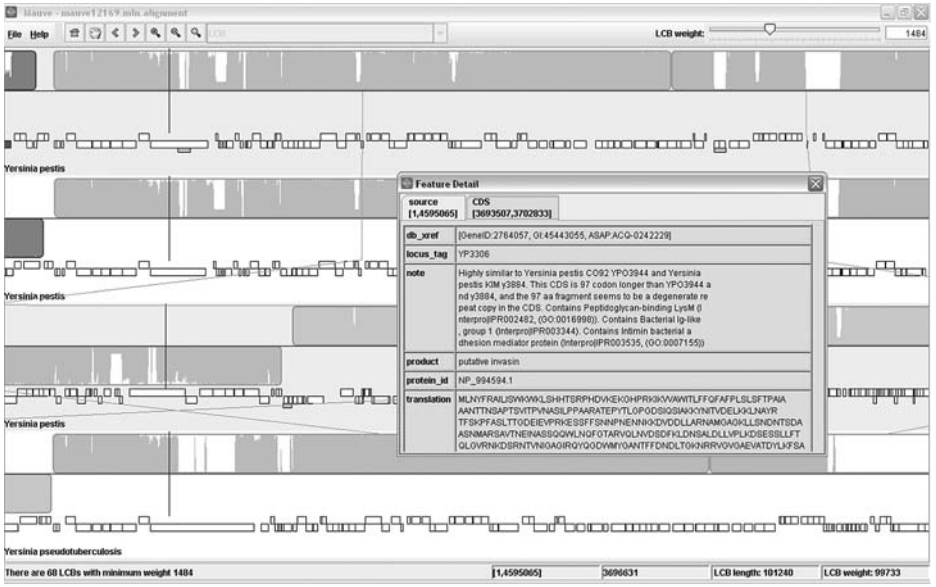

Fig. 4. A closer look at the alignment among the four *Yersinia*. In the regions indicated by the black vertical bar we see a putative orthologous gene with varying levels of conservation among the four genomes. The large white region in the *Yersinia pseudotuberculosis* genome indicates lineage-specific content—either because of ancestral deletion from the *Yersinia pestis* genome or an insertion in the *Y. pseudotuberculosis* lineage. A third possible explanation could be lateral transfer of a homologous allele from an unknown strain. The "Feature Detail" window gives a detailed listing of feature qualifiers for annotated genes which can be displayed by right-clicking a gene and selecting "view GenBank annotation" from the pop-up menu.

as Drosophila would require 2–3 GB of RAM to align in memory. Fortunately, Mauve can utilize two or more scratch disks to reduce memory requirements during generation of unique local alignments—the most memory-intensive part of the alignment process. Thanks to this feature, aligning the two complete *Drosophila* genomes using the Mauve Aligner can be accomplished in four steps:

1. Create sorted mer lists (.sml) for each genome using the scratch-path parameter to identify two or more disks that can be used. The same command must be run once to create an SML file for each genome and once to generate the local alignments (MUMs).

Fig. 5. The Mauve command-line tool listing its available program arguments.

a. Command to generate the SML file for *D. melanogaster*:
   `>mauveAligner –mums –scratch-path=/disk1 –scratch-path=/disk2 –output=drosophila.mums dmel.gbk dmel.gbk.sml dyak.fas dyak.fas.sml`
b. Run again to generate the SML file for *D. yakuba*:
   `>mauveAligner –mums –scratch-path=/disk1 –scratch-path=/disk2 –output=drosophila.mums dmel.gbk dmel.gbk.sml dyak.fas dyak.fas.sml`
c. And one more time to generate the local alignments (MUMs):
   `>mauveAligner –mums –scratch-path=/disk1 –scratch-path=/disk2 –output=drosophila.mums dmel.gbk dmel.gbk.sml dyak.fas dyak.fas.sml`

2. Generate initial alignment anchors without actually aligning:
   `>mauveAligner –match-input=drosophila.mums –weight=10000 –no-gapped-alignment –no-recursion –no-lcb-extension –output=drosophila_lcb.mums dmel.gbk dmel.gbk.sml dyak.fas dyak.fas.sml`
3. Configure MUSCLE alignment parameters to reduce memory usage and speed up the gapped alignment process:
   `>mauveAligner –match-input=drosophila_lcb.mums –lcb-match-input –muscle-args="-stable –maxiters 1 –diagsl -sv" –output-alignment=drosophila.xmfa dmel.gbk dmel.gbk.sml dyak.fas dyak.fas.sml`
4. View the output files using the Mauve alignment display (*see* **Note 3**).
   `>java –Xmx1200m –jar Mauve.jar drosophila.xmfa`

Analyzing the results with the Mauve alignment display, we can see evidence of the two main lineages of gypA and gypB gypsy elements most likely resulting from multiple lateral transfer events reported in **ref**. ***(44)***. *See* **Note 7** for converting the alignment output file into a signed gene-order permutation matrix for use with systems for rearrangement phylogeny such as the GRIMM/MGR server or BADGER. Additional command-line options are shown by running mauveAligner with no arguments, *see* **Fig. 5**.

4. Notes

1. When aligning divergent genomes, the seed size parameter can be reduced to between 9 and 13.
2. If alignment is unacceptably slow or using too much memory, try using ClustalW instead of MUSCLE.
3. If encountering a java.lang.OutOfMemoryError: Java heap space error, try increasing the heap space by running Mauve with the –Xmx command or closing any unused Mauve alignment windows.
4. If the console window is not present or has been closed, it can be reopened in the Help menu->show console.
5. Similar to all existing genome alignment systems, it is possible that Mauve may generate inaccurate alignments. Alignments can be manually edited using the Cinema-MX ***(45)*** alignment editor that has been incorporated into the Mauve interface. To access this feature, right click on the suspect LCB and then select "Edit this LCB." Then, the Cinema-MX alignment editor window will appear and allow for dynamic alignment correction, and when finished the Mauve interface will update with the adjusted alignment.
6. In the Mauve alignment viewer the LCB connecting lines can be hidden (or made visible again) by typing Shift+L (pressing shift and L simultaneously).
7. The LCBs generated by Mauve in the alignment output file can be transformed into a signed gene-order permutation matrix by supplying the –permutation-matrix-output=<directory> command-line argument. The permutation matrix makes suitable input to systems for rearrangement phylogeny such as the GRIMM/MGR server or BADGER.

Acknowledgments

This work was funded in part by National Institutes of Health grant GM62994-02. A.E.D. was supported by NLM grant 5T15LM007359-05. T.J.T. was supported by Spanish Ministry MECD research grant TIN2004-03382.

References

1. Needleman, S. B. and Wunsch, C. D. (1970) A general method applicable to the search for similarities in the amino acid sequence of two proteins. *J. Mol. Biol*. **48,** 443–453.
2. Smith, T. F. and Waterman, M. S. (1981) Identification of common molecular subsequences. *J. Mol. Biol*. **147,** 195–197.
3. Higgins, D. G. and Sharp, P. M. (1988) CLUSTAL: a package for performing multiple sequence alignment on a microcomputer. *Gene* **73,** 237–244.
4. Notredame, C., Higgins, D. G., and Heringa, J. (2000) T-Coffee: a novel method for fast and accurate multiple sequence alignment. *J. Mol. Biol*. **302,** 205–217.
5. Edgar, R. C. (2004) MUSCLE: multiple sequence alignment with high accuracy and high throughput. *Nucleic Acids Res*. **32,** 1792–1797.
6. Lee, C., Grasso, C., and Sharlow, M. F. (2002) Multiple sequence alignment using partial order graphs. *Bioinformatics* **18,** 452–464.
7. Abouelhoda, M. I. and Ohlebusch, E. (2003) A local chaining algorithm and its applications in comparative genomics. *Algorithms in Bioinformatics, Proceedings* **2812,** 1–16.
8. Haas, B. J., Delcher, A. L., Wortman, J. R., and Salzberg, S. L. (2004) DAGchainer: a tool for mining segmental genome duplications and synteny. *Bioinformatics* **20,** 3643–3646.
9. Hampson, S. E., Gaut, B. S., and Baldi, P. (2005) Statistical detection of chromosomal homology using shared-gene density alone. *Bioinformatics* **21,** 1339–1348.
10. Hampson, S., McLysaght, A., Gaut, B., and Baldi, P. (2003) LineUp: statistical detection of chromosomal homology with application to plant comparative genomics. *Genome Res*. **13,** 999–1010.
11. Tesler, G. (2002) GRIMM: genome rearrangements web server. *Bioinformatics* **18,** 492–493.
12. Spang, R., Rehmsmeier, M., and Stoye, J. (2002) A novel approach to remote homology detection: Jumping alignments. *Journal of Computational Biology* **9,** 747–760.
13. Calabrese, P. P., Chakravarty, S., and Vision, T. J. (2003) Fast identification and statistical evaluation of segmental homologies in comparative maps. *Bioinformatics* **19,** i74–i80.
14. Darling, A. E., Mau, B., Blattner, F. R., and Perna, N. T. (2004) GRIL: genome rearrangement and inversion locator. *Bioinformatics* **20,** 122–124.
15. Durbin, R. (1998) *Biological Sequence Analysis: Probabalistic Models of Proteins and Nucleic Acids*. Cambridge University Press, Cambridge, UK, pp. xi, 356.
16. Fitch, W. M. (2000) Homology a personal view on some of the problems. *Trends Genet*. **16,** 227–231.
17. Larget, B., Kadane, J. B., and Simon, D. L. (2005) A Bayesian approach to the estimation of ancestral genome arrangements. *Mol. Phylogenet. Evol*. **36,** 214–223.

18. Bourque, G. and Pevzner, P. A. (2002) Genome-scale evolution: reconstructing gene orders in the ancestral species. *Genome Res*. **12,** 26–36.
19. Wu, S. and Gu, X. (2003) Algorithms for multiple genome rearrangement by signed reversals. *Pac. Symp. Biocomput*. 363–374.
20. Lu, C. L., Wang, T. C., Lin, Y. C., and Tang, C. Y. (2005) ROBIN: a tool for genome rearrangement of block-interchanges. *Bioinformatics* **21,** 2780–2782.
21. Yancopoulos, S., Attie, O., and Friedberg, R. (2005) Efficient sorting of genomic permutations by translocation, inversion and block interchange. *Bioinformatics* **21,** 3340–3346.
22. Holder, M. and Lewis, P. O. (2003) Phylogeny estimation: traditional and Bayesian approaches. *Nat. Rev. Genet*. **4,** 275–284.
23. Yang, Z., Ro, S., and Rannala, B. (2003) Likelihood models of somatic mutation and codon substitution in cancer genes. *Genetics* **165,** 695–705.
24. Lunter, G., Ponting, C. P., and Hein, J. (2006) Genome-Wide Identification of Human Functional DNA Using a Neutral Indel Model. *PLoS Comput. Biol*. **2,** e5.
25. Darling, A. C., Mau, B., Blattner, F. R., and Perna, N. T. (2004) Mauve: multiple alignment of conserved genomic sequence with rearrangements. *Genome Res*. **14,** 1394–1403.
26. Brudno, M., Malde, S., Poliakov, A., et al. (2003) Glocal alignment: finding rearrangements during alignment. *Bioinformatics* **19,** i54–i62.
27. Ovcharenko, I., Loots, G. G., Giardine, B. M., et al. (2005) Mulan: multiple-sequence local alignment and visualization for studying function and evolution. *Genome Res*. **15,** 184–194.
28. Treangen, T. J. and Messeguer, X. (2006) M-GCAT: interactively and efficiency constructing large-scale multiple genome comparision frameworks in closely related species. *BMC Bioinformatics* **7,** 433.
29. Kurtz, S., Phillippy, A., Delcher, A. L., et al. (2004) Versatile and open software for comparing large genomes. *Genome Biol*. **5,** R12.
30. Hohl, M., Kurtz, S., and Ohlebusch, E. (2002) Efficient multiple genome alignment. *Bioinformatics* **18,** S312–S320.
31. Blanchette, M., Kent, W. J., Riemer, C., et al. (2004) Aligning multiple genomic sequences with the threaded blockset aligner. *Genome Res*. **14,** 708–715.
32. Bray, N. and Pachter, L. (2004) MAVID: constrained ancestral alignment of multiple sequences. *Genome Res*. **14,** 693–699.
33. Brudno, M., Do, C. B., Cooper, G. M., et al. (2003) LAGAN and Multi-LAGAN: efficient tools for large-scale multiple alignment of genomic DNA. *Genome Res*. **13,** 721–731.
34. Choi, K. P., Zeng, F., and Zhang, L. (2004) Good spaced seeds for homology search. *Bioinformatics* **20,** 1053–1059.
35. Henz, S. R., Huson, D. H., Auch, A. F., Nieselt-Struwe, K., and Schuster, S. C. (2005) Whole-genome prokaryotic phylogeny. *Bioinformatics* **21,** 2329–2335.

36. Saitou, N. and Nei, M. (1987) The neighbor-joining method: a new method for reconstructing phylogenetic trees. *Mol. Biol. Evol*. **4,** 406–425.
37. Cleri, D. J., Vernaleo, J. R., Lombardi, L. J., et al. (1997) Plague pneumonia disease caused by Yersinia pestis. *Semin. Respir. Infect*. **12,** 12–23.
38. Carniel, E. (2003) Evolution of pathogenic Yersinia, some lights in the dark. *Adv. Exp. Med. Biol*. **529,** 3–12.
39. Chain, P. S., Carniel, E., Larimer, F. W., et al. (2004) Insights into the evolution of Yersinia pestis through whole-genome comparison with Yersinia pseudotuberculosis. *Proc. Natl. Acad. Sci. USA* **101,** 13,826–13,831.
40. Hinnebusch, B. J. (2005) The evolution of flea-borne transmission in Yersinia pestis. *Curr. Issues Mol. Biol*. **7,** 197–212.
41. Perna, N. T., Plunkett, G., 3rd, Burland, V., et al. (2001) Genome sequence of enterohaemorrhagic Escherichia coli O157:H7. *Nature* **409,** 529–533.
42. Hsiao, W. W., Ung, K., Aeschliman, D., Bryan, J., Finlay, B. B., and Brinkman, F. S. (2005) Evidence of a large novel gene pool associated with prokaryotic genomic islands. *PLoS Genet*. **1,** e62.
43. Tettelin, H., Masignani, V., Cieslewicz, M. J., et al. (2005) Genome analysis of multiple pathogenic isolates of Streptococcus agalactiae: implications for the microbial "pan-genome". *Proc. Natl. Acad. Sci. USA* **102,** 13,950–13,955.
44. Terzian, C., Ferraz, C., Demaille, J., and Bucheton, A. 2000) Evolution of the Gypsy endogenous retrovirus in the Drosophila melanogaster subgroup. *Mol. Biol. Evol*. **17,** 908–914.
45. Lord, P. W., Selley, J. N., and Attwood, T. K. (2002) CINEMA-MX: a modular multiple alignment editor. *Bioinformatics* **18,** 1402–1403.