A peer-reviewed version of this preprint was published in PeerJ on 12 June 2017.

AliTV – interactive visualization of whole genome comparisons

Markus J. Ankenbrand[1,*], **Sonja Hohlfeld**[1,2,*], **Thomas Hackl**[2,3], and **Frank Förster**[2,4]

[1]**Department of Animal Ecology and Tropical Biology, Julius Maximilian University, Würzburg, Germany**
[2]**Department of Bioinformatics, Julius Maximilian University, Würzburg, Germany**
[3]**Department of Civil and Environmental Engineering, Massachusetts Institute of Technology, Cambridge, Massachusetts, USA**
[4]**Center for Computational and Theoretical Biology, Julius Maximilian University, Würzburg, Germany**
[*]**These authors contributed equally to this work**

ABSTRACT

Whole genome alignments and comparative analysis are key methods in the quest of unraveling the dynamics of genome evolution. Interactive visualization and exploration of the generated alignments, annotations, and phylogenetic data are important steps in the interpretation of the initial results. Limitations of existing software inspired us to develop our new tool AliTV, which provides interactive visualization of whole genome alignments. AliTV reads multiple whole genome alignments or automatically generates alignments from the provided data. Optional feature annotations and phylogenetic information are supported. The user-friendly, web-browser based and highly customizable interface allows rapid exploration and manipulation of the visualized data as well as the export of publication-ready high-quality figures. AliTV is freely available at https://github.com/AliTVTeam/AliTV.

Keywords: Comparative genomics, alignment, visualization

INTRODUCTION

Advances in short- and long-read sequencing and assembly over the last decade (**???**) have made whole genome sequencing a routine task for biologists in various fields. Public sequence databases already contain several thousand of draft and finished genomes (**?**), with many more on the way (**?**). In particular, high throughput sequencing projects of pathogen strains related to recent outbreaks (**?**), and large scale ecological studies targeting microbial communities and pan genomes of populations using metagenome and single cell sequencing approaches contribute in this process (**??**). These rich data sets can be explored for large-scale evolutionary processes using comparative genomics and whole genome alignments, revealing genomic recombinations (**???**), islands and horizontal gene transfer (**???**) as well as the often related dynamics of mobile or endogenous viral elements (**??**). Other applications of whole genome comparisons include the analysis of paleopolyploidization events (**?**) and quantitative measurements of intra-tumour heterogeneity (**?**).

However, to facilitate proper interpretation of the obtained whole genome comparisons, visualization is key. One of the first tools to provide an interactive graphical representation of aligned genomes is the multiple whole genome alignment program `Mauve` (**?**). `Mauve` represents genomes in a co-linear layout with homologous syntenic blocks indicated by colors and connecting lines. The interactive stand-alone viewer `ACT` (**?**), in addition to alignment blocks, supports the representation of genomic annotations, such as genes. The `R` library `genoPlotR` (**?**) and the `Python` based application `EasyFig` (**?**), both also based on a co-linear layout and supporting feature annotations, lack interactive analysis features as they are designed to generate static figures.

In addition to co-linear layouts, tools using circular representations of genomes have been developed. `BLASTatlas` (**?**) and `BRIG` (**?**) use multiple concentric rings to represent data of individual genomes, with `BRIG` also providing an interactive graphical interface. `GenomeRing` (**?**) uses a circular representation as well, however, places all genomes on the same ring and syntenic blocks are connected with arcs extending into the center of the ring.

The web-based comparative genomics software `Sybil` (**?**) provides interactive co-linear visualization of multiple whole genome alignments with feature annotations and also supports a

phylogenetic tree alongside the alignments. The software builds on a relational `Chado` database schema and, therefore, requires upload and import of custom data sets prior to analysis.

During our analysis of existing software, we found that interactive tools are useful for data exploration, but offer limited support for the figure export and at low qualities. Scripting-based tools provide higher levels of customization and figure quality, however, require familiarity with the respective language, thus often rendering the generation of figures time-consuming. For web- and database-based suites, such as `Sybil`, the upload and import procedure complicate utilization and limit applicability.

Here we present our stand-alone application `AliTV` (Alignment Toolbox and visualization) designed for interactive visualization of multiple whole genome alignments. `AliTV` aims to enable researches to either directly read or automatically generate new whole genome alignments, rapidly explore the results, manipulate and customize the visualization and, at the end of the day, export appealing, publication-grade figures. `AliTV` reads sequence and annotation or alignment data in common formats (FASTA, GenBank, GFF, MAF, Newick, ...), and internally computes alignments using `lastz` (**?**). The user-friendly interface is built on the state-of-the-art `D3.js` JavaScript framework and can be utilized in a platform independent manner with common web browsers. Genomes are represented in a highly customizable co-linear layout including annotations and an optional phylogenetic tree. The tree is not computed by `AliTV` but has to be provided during data generation. Also the order of genomes is not automatically optimized to minimize rearrangements. Customizations to the figure by the user can be saved, reloaded, and exported to high quality `SVG` files.

METHODS

Our tool `AliTV` is divided into two parts. The first non-interactive part is required for the generation of the input files for our interactive viewer. The second part represents that interactive viewer in the form of a `SVG` file embedded in a `HTML5` website. The latest version of our code can be obtained from `GitHub` (https://github.com/AliTVTeam/AliTV). It is planned to adjust `AliTV` in order to integrate it into the `BioJS` registry (https://biojsnet.herokuapp.com/, **?**). The general design of `AliTV` assures, that `AliTV` runs on different hard- and software platforms, e.g. Linux, MacOSX, and Windows. The following sections describe those parts in more detail.

Data Preparation

The data preparation is performed by a single `Perl` script named `alitv.pl`. This script uses a set of different `Perl` modules to import incoming data and generate valid `JSON` input data for our visualization engine described in the next paragraph. One of our aims is to support as many different input formats for sequence and annotation information as possible. Therefore, we used the well tested and broadly accepted `BioPerl` as basis for our modules (**?**).

The script `alitv.pl` uses a `YAML` file to specify the different input files. Moreover, an easy-to-use-mode is available which requires only a couple of input files and generates the required `YAML` file on the fly. This generated `YAML` settings file might be used to reproduce `AliTV` results or can be used as starting point to alter configuration parameters.

During the preparation step, `AliTV` requires all-vs-all alignments of the complete sequence set. Those alignments are generated or user provided. The current version of `alitv.pl` requires `lastz` to generate all alignments in `MAF` format (**?**). Nevertheless, `BioPerl` supports a broad range of alignment formats. Therefore, other programs can easily be added to the list of supported alignment programs. Moreover, the ability to use existing alignments allows a huge time benefit, when `AliTV` parameters are changed to optimize the visualization via `YAML` settings file in a non-interactive manner. Thus future versions of `alitv.pl` will support caching of alignments based on checksums to avoid unnecessary recalculations.

The final result of our `alitv.pl` is a `JSON` file, which can be load into our interactive visualization page.

Interactive Visualization

`AliTV` is implemented in JavaScript. Our code is documented using `JSDoc 3` (version 3.4.0 http://usejsdoc.org/, 02.06.2016). `AliTV` generates a `SVG` which is presented within a browser using `HTML5`. A tutorial is available at https://alitv.readthedocs.io/en/latest/index.html.

To gain advanced application possibilities we use different libraries. The JavaScript library `D3.js 3.5.17` (http://d3js.org/, 06.06.2016) provides a wide range of pre-built functions for calculating and drawing the interactive figure. In addition, `AliTV` employs `JQuery 2.2.4` (https://jquery.com/,

Table 1. Chloroplast genomes of the parasitic and non-parasitic plants used in the case study.

| Species	Accession	Life-style	Reference
Olea europaea	NC_013707	non-parasitic	**?**
Lindenbergia philippensis	NC_022859	non-parasitic	**?**
Cistanche phelypaea	NC_025642	holo-parasitic	**?**
Epifagus virginiana	NC_001568	holo-parasitic	**?**
Orobanche gracilis	NC_023464	holo-parasitic	**?**
Schwalbea americana	NC_023115	hemi-parasitic	**?**
Nicotiana tabacum	NC_001879	non-parasitic	**?**

06.06.2016) to ease access to several parts of the figure. This is helpful for hiding selected sequences, genes or links. `JQueryUI 1.11.4` (https://jqueryui.com/, 06.06.2016) gives us the possibilities to add user-friendly interactions to `AliTV`. With sliders the user has the chance to specify values for link length and link identity. Context menus offer direct and native interactions with the figure.

To guarantee correct code functionality we engineer `AliTV` according to the Test Driven Development. First we write an automated test case that defines a new function. Then we add the minimum amount of code to make the test pass. Finally we refactor the code to accepted standards. We use `Jasmine 2.3` (http://jasmine.github.io/, 06.06.2016), as framework for testing our JavaScript code. The tests can run either via the SpecRunner or the command line using the taskrunner `grunt 1.0.0` (http://gruntjs.com/, 06.06.2016).

RESULTS AND DISCUSSION

To demonstrate the capabilities of `AliTV` we describe a short case study using seven published chloroplast genomes (table 1). Four of the chloroplasts belong to parasitic plant species and three to non-parasitic ones. Parasitic plants rely much less or not at all on photosynthetic activity, a trait that should be reflected in the genomic structure of their chloroplast genomes. To assess this hypothesis the chloroplast genomes were downloaded from NCBI and processed with `alitv.pl`. For demonstration purposes, the chloroplast genome of *Nicotiana tabacum* was split in two pieces to represent an unfinished genome with more than one contig, and the genome sequence of *Schwalbea americana* was reverse-complemented (flipped). The pair-wise whole genome alignments are visualized by `AliTV` (figure 1A). The left-hand side of the display panel shows the phylogenetic tree for the seven species with species names as tip labels (parasitic plants are highlighted with an asterisk). The tree has been created provided in accordance to NCBI taxonomy (**?**). Next to the tip labels, each genome is drawn as a scaled and annotated horizontal bar. The orientation of the *S. americana* genome was swapped back to match the orientation of the other genomes, indicated by the tick coordinates in reverse order (0 on the right side). *N. tabacum* is represented by two bars as the sequence has been split into two parts. On those bars features (e.g. genes or inverted repeats (IRs)) are shown as either rectangles or arrows. Alignments between adjacent genomes are represented as colored ribbons. The bottom legend shows the default color scale from red to green corresponding to low and high identity respectively.

The most striking observation is that three of the chloroplast genomes have drastically reduced sizes. All of those are parasitic (table 1). Interestingly the chloroplast genome size of *S. americana* is similar to that of the non-parasitic plants. This can be explained by the life style of *S. americana* which is hemi-parasitic in contrast to the other parasitic plants which are holo-parasites. The features shown are the IR regions as arrows, the hypothetical chloroplast open reading frames as orange and the genes of the ndh family as pink rectangles. First, it can be seen that there is a big variation in size of the inverted repeats. While the IR of *Orobanche gracilis* is the shortest with roughly 5000 bp, that of *S. americana* is the largest with roughly 35 000 bp. Second, there are less genes of the ndh family on *Cistanche phelypaea*, *Epifagus virginiana*, *O. gracilis*, and *S. americana*. Members of the ndh gene family encode subunits of the NADH dehydrogenase-like complex, which is involved in chlororespiration (**?**). However they are not required for plant growth under optimal conditions (**?**). The absence of ndh genes in chloroplasts of parasitic plants has been studied in detail in **?**. Loss of ndh genes has also been reported for photosynthetic plants such as some conifers and orchids (**??**). Looking at the pairwise similarities of adjacent genomes, it is apparent that the non-parasitic plants (e.g. *Olea europaea* and *Lindenbergia philippensis*) have high overall sequence identity. In

3/5

AliTV – interactive visualization of whole genome comparisons
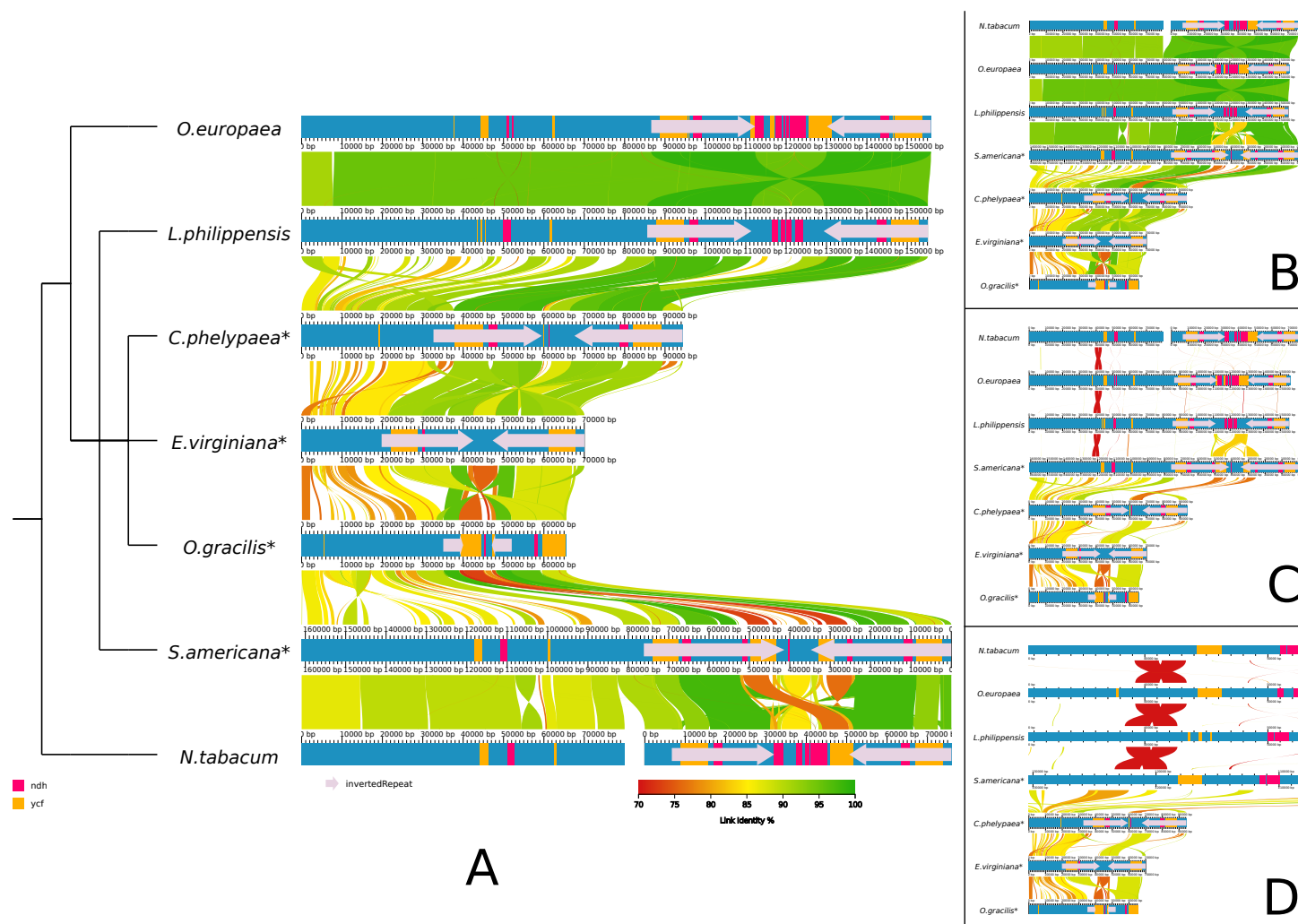
Figure 1. Whole genome alignment of seven chloroplasts visualized by `AliTV`. Species names were italicized and parasites marked with asterisks ex post. **(A)** Default layout with a phylogenetic tree on the left-hand side and genomes represented by co-linear horizontal bars on the right; genes and inverted repeats are displayed as rectangles and arrows, respectively; colored ribbons connect corresponding regions in the alignment. **(B-D)** Customized layouts: **(B)** reordered genomes, non-parasitic plants at the top and holo-parasitic plants at the bottom. **(C)** links filtered by identity (only those with 50 % to 90 % identity are drawn). **(D)** zoom in on a potential segmental duplication (red 'X'-shaped links) in the top four genomes.

contrast, the sequence similarity within parasitic plants is lower. This observation can help framing a hypothesis about the evolutionary pressure on chloroplasts of parasitic plants. Another interesting observation is the distribution of missing regions of *C. phelypaea* in comparison to *L. philippensis*. Missing regions are distributed all over the genome and the order of the remaining parts remains stable. **?** describe an inversion in the large single copy region of *S. americana* compared to non-parasitic plants which is clearly visible by the link to *N. tabacum* around the 115 kbp position. All these observations can be made by simply looking at the raw figure created by `alitv.pl` and visualized by `AliTV`. However the figure can be analyzed interactively in more detail. One shortcoming of the linear representation of whole genome alignments is the limited comparability of non-adjacent sequences. Therefore, `AliTV` provides a way for the user to re-order the genomes on the figure (figure 1B). If reordering causes inconsistencies with the phylogenetic tree, the tree is hidden and a warning message is displayed. Furthermore, the links can be filtered by their alignment identity. The default setting is to display only links with minimal identity of 70%. But sometimes it might be interesting to look at regions with less similarity. To see these regions it is also important to hide large regions with high similarity. This can be achieved by changing the identity via a slider (figure 1C). After setting the identity range to 50 % to 90 % red 'X'-shaped links between *N. tabacum*, *O. europaea*, *L. philippensis*, and *S. americana* become apparent. For detailed inspection of regions of interest, `AliTV` provides a zoom function (figure 1D). This way the exact location of the alignments can be traced to the locations of psaA and psaB. Moreover `AliTV` provides functions like alignment length filtering, selective hiding of sequences, links and features, change of orientation (reverse complement) and rotation of circular chromosomes. Finally, it is possible to tweak many graphical parameters, such as colors, labels or spacing, directly via the interface to produce a publication quality figure which can be saved in `SVG` format. Furthermore, the current state can be saved in `JSON` format in order to share it with collaborators or continue the work with `AliTV` at a later time.

CONCLUSION

The case study demonstrates the suitability of `AliTV` as a tool for visualizing and analyzing whole genome comparisons. `AliTV` can be used to easily create a figure that show cases many genomic features at once. Furthermore the rich interactive features enable the exploratory analysis and discovery of previously unknown features. Thus novel hypotheses can be generated that can then be validated with experimental methods. So `AliTV` is a useful tool that will help scientists to find biologically meaningful information in the vast amount of genomic data.

ACKNOWLEDGMENTS