RecDetec – Detecting Recombination and Phylogenetic Information Along Alignments

Heiko A. Schmidt[#], Moritz B. Reinhardt, Tung Lam Nguyen and Arndt von Haeseler

Center for Integrative Bioinformatics Vienna (CIBIV), Max F. Perutz Laboratories (MFPL), Vienna, Austria; University of Vienna, Medical University of Vienna, Vienna, Austria

Running title: Detecting Recombination and Phylogenetic Information

Keywords: recombination detection; phylogenetic information; alignment assessment; overlapping recombinations; bootscanning

[#]Corresponding author: heiko.schmidt@univie.ac.at

word counts:

abstract: 137 words

main text: 3520 words

Abstract

Recombination is a common mechanism, occurring with high rates especially in RNA viruses. Thus, phylogeny reconstruction of viruses is challenging and detecting recombinants and recombination break points has become an important task when analyzing viral genomes. If the data contain several recombinants or, moreover, recombinants originating from overlapping recombinations (i.e. independent recombination events that involve parental strains from the same group), recombination detection gets difficult. Here, we utilize the information obtained from initial ML phylogenetic reconstructions on sliding windows to reveal recombinants and recombination breakpoints. Our approach detects complex recombination patterns without having to recompute the phylogenetic trees. The approach is implemented in the software RecDetec. The software also allows for using parallel computing platforms to reduce runtime. An illustrative example highlights the utility of RecDetec. Finally, our results are compared to other approaches.

Introduction

Determining the history of a viral strain has become a common task in virus research, e.g. in genotyping strains (4, 2) or when reconstructing the migration paths of an epidemic (5, 19, 20). However, variation of evolutionary rates, different selection pressures (either over time or along the genome), reassortment and recombination (14, 10) confound the reconstruction of the history. Especially in viruses recombination is a major force promoting adaptation, e.g., for an effective evasion of the immune or other defense systems of their hosts (10), or for promoting drug resistance (3). Thus, phylogenetic reconstruction for viruses is still a challenging task.

In Eukaryotes and DNA viruses recombination usually requires enzyme-mediated double

strand breaks (16), whereas in RNA viruses and retroviruses RdRP (RNA-dependent RNA polymerase) and RT (reverse transcriptase) can switch their RNA-template during replication, thus, connecting the information of potentially different parental template sequences (27). Rates as high as 2-3 RNA recombination events per genome and replication have been reported for HIV (11). Although recombination certainly also occurs among identical sequences, for a recombination event to be detectable, the parental strains co-infecting the same cell have to exhibit a certain degree of sequence divergence. If recombination takes place between divergent parents, the different histories of the recombined parts can be investigated by phylogenetic approaches. While the detection of a single recombinant strain in a collection of aligned sequences is not too difficult, this task is hard if the alignment contains several recombinants. The analysis gets even more difficult with overlapping recombinations, that is, recombinants originating from independent recombination events having parents belonging to the same group of reference strains in overlapping genomic regions.

Owing to the importance of recombination methods abound to detect recombination in genomic sequences (e.g., (13, 22)), among them the popular bootscanning (23), a sliding window approach. Bootscan implementations (15, 17, 2) use an alignment as input. The sequences of the alignment are typically assigned to disjoint (reference) groups and for each group the consensus sequence is computed or an arbitrary sequence is selected to represent its group (23). For a putatively recombinant query group bootscanning then searches for possible parents within the reference groups and the corresponding recombination breakpoints. To that end the alignment is split into overlapping windows and each window is subjected to a bootstrap analysis (8, 7). Finally, the bootstrap support for clustering the query group with each of the reference groups are collected for each window and plotted along the alignment. Under the assumption that the query group will get high bootstrap support if clustered with its parental subtypes in the respective windows, the recombination

breakpoints can be detected (cf. Fig. 1). However, if the user redefines query or reference groups or wants to focus on a relevant subset of sequences, one has to redo the complete time consuming analysis. There are two stand-alone implementations (15, 17) of bootscanning for Microsoft Windows. In addition, some web tools employ bootscanning for subtyping query sequences of specific viruses (4, 2).

Here, we present RecDetec to study recombination using maximum likelihood (ML) phylogenies. RecDetec finds simple recombinants and also detects overlapping recombination. Furthermore, the user can redefine reference and query groups or exclude strains without repeating the time consuming phylogenetic inference. Finally, RecDetec also analyzes the phylogenetic information content in the alignment. RecDetec is an exploratory tool that runs on Linux/Unix, MacOSX and Windows.

Materials and Methods

Recombination analysis with RecDetec

RecDetec takes as input a multiple sequence alignment of genomic sequences to which the sliding-window approach (cf. Fig. 1) is applied, where window size and step size can be specified by the user. RecDetec computes two support values. First it computes bootstrap supports for each window based on maximum likelihood trees inferred with IQPNNI (18) which results in *ML bootscanning* diagrams. Second, RecDetec determines the support values using the Quartet Puzzling (QP) method implemented in the TREE-PUZZLE program (28, 25) to produce *QP-scanning* diagrams. Contrary to the original bootscanning description, where user-defined groups of sequences are represented as consensus sequence or by an arbitrary representative, RecDetec reconstructs trees for all sequences. All splits, i.e. branches, found during the bootstrap analysis are collected and their frequencies are counted

for each window. The sequences are usually assigned to user-defined disjoint groups, namely one query group where recombinants are suspected and a set of reference groups containing potential parents. Groups should typically comprise sequences with the same phylogenetic background tracing back to a single common ancestor, but also other user-defined groups can be analyzed. Groups may comprise pure (i.e. not recombined) sequences of the same subtype or taxonomic group, but can also consist of recombinant forms. Each group can contain one or more sequences.

Finally, the bootstrap values for query group/reference group clusters are computed. To that end the support values are computed for branches separating the query group sequences $\{Q_1 \dots Q_l\}$ together with the sequences of exactly one reference group $\{G_1 \dots G_k\}$ from the remaining sequences $\{R_1 \dots R_m\}$. The support value for a branch separating $\{Q_1 \dots Q_l, G_1 \dots G_k\}$ from the rest is computed for each query group reference group pair and for each window along the alignment. These values are then plotted along the alignment. In addition, RecDetec offers the possibility to visualize the bootstrap consensus tree for any window employing the FigTree software (http://tree.bio.ed.ac.uk/software).

RecDetec offers further analyses at no additional computation cost. RecDetec can plot the bootstrap (8, 7) or puzzle support (25, 28) values for each user-defined group and each sliding window. The resulting diagrams visualize the group support as a measure of phylogenetic stability of groups along the alignment. This allows to detect genomic regions where the support of a group is lost or to detect unreasonable groups, if they are not supported at all.

Analyzing subsets of sequences: In RecDetect the user can exclude interfering recombinant sequences when generating bootscanning or group support diagrams, if they would otherwise obstruct the signal. In the following we describe how RecDetec obtains the support values for the respective diagrams in this case.

For the complete set of sequences, i.e. no sequences excluded, the support for a common

subtree of sequences $\{W_1 \dots W_k\}$ (e.g. the query group with one reference group) can be obtained directly from the collection of bootstrap data. This is straightforward, because a split that bipartites all sequences into $\{W_1 \dots W_k\}$ and the remaining sequences $\{R_1 \dots R_m\}$ is unique.

If one sequence $\{X\}$ is excluded when obtaining the support of the group $\{W_1 \dots W_k\}$ against the remaining taxa $\{R_1 \dots R_m\}$, $\{X\}$ can cluster with $\{W_1 \dots W_k\}$ (Fig. 2a) or with the remaining taxa $\{R_1 \dots R_m\}$ (Fig. 2b). In the special case of $\{X\}$ being located between $\{W_1 \dots W_k\}$ and $\{R_1 \dots R_m\}$ (cf. Fig. 2c) both relevant splits exist in the same tree. For visualization we use the split from b_i and b_j that maximizes the support value. Note, that after excluding a sequence $\{X\}$ it is included neither in $\{W_1 \dots W_k\}$ nor in $\{R_1 \dots R_m\}$.

When excluding more sequences, say $\{X_1 \dots X_q\}$, the support values of the relevant splits can still be collected by examining splits separating $\{W_1 \dots W_k\}$ from the remaining sequences $\{R_1 \dots R_m\}$, where each 'excluded' sequence clusters either with $\{W_1 \dots W_k\}$ or $\{R_1 \dots R_m\}$. From all 2^q possible splits of that kind we use for visualization the split with maximal support.

Visualizing phylogenetic information content: Two means have been implemented to visualize the phylogenetic information content in the windows induced sub-alignment. First, likelihood mapping (29) is used to measure phylogenetic information in each window using the maximum likelihood values for the three trees relating four sequences (quartets). If one tree has a high likelihood compared to the others, the quartet is called a *resolved quartet*. If the likelihood from two trees are more or less the same and larger than the third likelihood, the quartet is a *partly resolved quartet*, otherwise it is an *unresolved quartet*. The percentages of resolved, partly resolved and unresolved quartets are plotted along the alignment. A high percentage of unresolved quartets marks alignment windows with little phylogenetic information (26, 29).

Second, the amount of parsimony informative sites (30) can be displayed. We use the following extended definitions of parsimony informative sites: A parsimony *informative site* is an alignment column that contains at least two different nucleotides and at least two of the nucleotides occur at least twice. A parsimony informative site is *partly informative*, if some nucleotides nucleotides occur only once, or if gaps or other ambiguous characters (like N) occur at that site. All other alignment columns are *(parsimony) uninformative*. Among these, two types of constant sites are defined. *Completely constants sites* contain only one nucleotide in all sequences, while *constant sites* can contain gaps or ambiguous characters (like N) besides one single nucleotide. The counts for these categories can also be plotted for each window and display the parsimony phylogenetic information content.

Parallel execution of the phylogenetic reconstructions: To save computation time, RecDetec supports execution on parallel computing platforms. For instance, freely available scheduling or middleware software can distribute the tasks of the RecDetec workflow to parallel computing platforms like clusters, grids, cloud computing environments or just local multicore machines (9, 32, 31). After the phylogenetic bootstrap analyses are finished the reconstructed trees can then be imported into RecDetec for the final analysis. However, the reconstructions can just as well be performed inside RecDetec without employing a cluster.

Results

An example scenario with overlapping recombination: We simulate a dataset assuming a scenario of overlapping recombinations, i.e. two independent recombinants have parental strains in the same reference group.

To this end, we generate an alignment of 7000bp length and eleven 'genomes' ($A1$, $A2$, $B1$, $B2$, $C1$, $C2$, $D1$, $D2$, $O1$, $O2$, X) containing three regions with different evolutionary rates.

In the first 500bp of the sequences the evolutionary rates were 50-fold increased, while the last 500bp of the sequences have a 50-fold reduced evolutionary rate with respect to regions 501-6500bp. Sequences have been simulated using seq-gen (21) according to the following evolutionary scenario. The dataset comprises a recombinant sequence $\{X\}$ as query group and four reference groups $\{A1, A2\}$, $\{B1, B2\}$, $\{C1, C2\}$, $\{D1, D2\}$ and outgroup $\{O1, O2\}$, where $\{B1, B2\}$ is also recombinant creating the scenario of overlapping recombinations. The evolutionary history of the sequences is depicted in Fig. 3a. The recombination events lead to chimeric sequences for $\{X\}$ and $\{B1, B2\}$. $\{X\}$ shares a common history (Fig. 3b) with $\{C1\}$ in region **a** (1-2000bp) and with $\{A2\}$ otherwise (2001-7000bp), while group $\{B1, B2\}$ shares a common history (Fig. 3c) with $\{A1, A2\}$ in regions **a**, **b**, **d** (1-3500bp and 5001-7000bp) and with $\{D2\}$ in region **c** (3501-5000bp). The underlying phylogenetic trees for the four regions **a**-**d** are depicted in Fig. 4a-d.

RecDetec analysis: Tree reconstructions are performed for ML bootscanning with window size 300, step size 25. For the initial recombination analysis, we assume the sequences are grouped based on some prior knowledge (like, for instance, preliminary phylogenetic analysis of the first 2000bp of the genomes) into the reference groups $\{A1, A2, B1, B2\}$, $\{C1, C2\}$, $\{D1, D2\}$ and $\{O1, O2\}$.

Prior to the recombination analysis, we assess the phylogenetic information in the alignment and visualize the phylogenetic information content based on likelihood mapping (Fig. 4e). The red curve displays the fraction of unresolved quartets. It is high at the ends of the alignment, indicating only very little or no phylogenetic information. Now we can use the parsimony informative sites diagram (Fig. 4f) to determine whether the lack of phylogenetic information is due to small sequence diversity or noise. On the right end of the alignment the number of constant sites is very high implying low diversity, whereas the left end of the alignment has a high number of parsimony informative sites (up to 100%, Fig. 4f). However,

since the corresponding fraction of resolved quartets is low in the phylogenetic information plot (Fig. 4e), this indicates that the phylogenetic signal is lost due to the high mutation rate in that part of the alignment. This analysis shows that the first and the last 500bp of the alignment are not suitable for phylogenetic analysis. We will ignore these regions in the following.

Next we choose $\{X\}$ as query group for ML bootscanning (Fig. 4g). In region **a** we observe high bootstrap support for the cluster joining $\{X\}$ and $\{C1, C2\}$ (red curve), whereas the support drops to 0 elsewhere. In regions **b** and **d** we observe high support for a cluster of $\{X\}$ with $\{A1, A2, B1, B2\}$ (turquoise curve). No grouping of $\{X\}$ with any reference group is observed in the region **c**. Fig. 4g shows a clear signal for different evolutionary histories before and after position 2000 ($\{X\}$ being related to $\{C1, C2\}$ and to $\{A1, A2, B1, B2\}$). But no statement about the phylogenetic history of $\{X\}$ in region **c** is possible.

To further elucidate this, we plot the group support for $\{A1, A2, B1, B2\}$ excluding the recombinant $\{X\}$ (Fig. 4h). The plot reveals that the reference group has no phylogenetic support in region **c**, i.e., a subtree with sequences $\{A1, A2, B1, B2\}$ is not found (bootstrap support close to zero). Bootstrap consensus trees reconstructed for region **c** confirm this. However, in this tree we observe that $\{B1, B2\}$ groups with $\{D1, D2\}$ in region **c**. Thus, $\{B1, B2\}$ is possibly a recombinant strain. To investigate this we plot an ML bootscan diagram with query group $\{A1, A2\}$, excluding the putatively recombinant groups $\{X\}$ and $\{B1, B2\}$ from the analysis (Fig. 4i). This plot shows no signal that region **c** of $\{A1, A2\}$ was exchanged by recombination. Then we use $\{B1, B2\}$ as query group and exclude $\{X\}$ and $\{A1, A2\}$. Fig. 4j shows that $\{B1, B2\}$ clusters 'correctly' with $\{O1, O2\}$ in regions **a**, **b** and **d** (magenta curve), while it clusters with $\{D1, D2\}$ in region **c** (cyan curve). This supports that $\{B1, B2\}$ is indeed a recombinant form.

Since the group $\{A1, A2, B1, B2\}$ contains pure and recombinant sequences, we will re-

analyze the putatively recombinant groups $\{X\}$ and $\{B1, B2\}$ with reference groups $\{A1, A2\}$, $\{C1, C2\}$, $\{D1, D2\}$ and $\{O1, O2\}$. One recombinant can obscure the signal of the other (overlapping) recombinant in the diagrams because they share common subtrees with the same parents in some regions. Thus, the two recombinant groups will be analyzed separately, each group will act as as query excluding the other recombinant group when plotting the diagram. Now the bootscan plot (Fig. 4k) for $\{X\}$ (excluding $\{B1, B2\}$) shows nicely the recombination pattern for $\{X\}$ as being a recombinant of $\{C1, C2\}$ (region **a**, red curve) and $\{A1, A2\}$ (regions **b-d**, blue curve). Likewise, Fig. 4l shows that $\{B1, B2\}$ is indeed a recombinant of sequences related to $\{A1, A2\}$ (regions **a**, **b**, **d**, blue curve) and $\{D1, D2\}$ (region **c**, turquoise curve). Thus, we could show that $\{X\}$ and $\{B1, B2\}$ are indeed overlapping recombinants and detected the regions where the different recombination break points are located. Please note, that the time consuming phylogenetic analysis was only run once at the beginning.

Simplot analysis: In addition, we analyzed this dataset using bootscanning where groups are represented by their consensus sequences as implemented in SimPlot (15) (using the same parameters for window size, step width, and evolutionary model as above). The SimPlot bootscan diagram (Fig. 5) shows the recombination breakpoint at 2000bp and the relationship of $\{X\}$ with $\{C1, C2\}$ before and with $\{A1, A2, B1, B2\}$ after the breakpoint. However, there are no hints that $\{A1, A2, B1, B2\}$ contains recombinant sequences.

GARD analysis: We analyze the dataset with GARD (12), another ML-based recombination detection tool which uses a genetic algorithm to determine recombination break points. GARD identifies breakpoints at about 500bp, 2000bp, 3500bp, 5000bp and 6500bp, marking the boundaries of all six genomic regions from the simulation. However, GARD cannot detect whether the breakpoints were caused by recombination events or by changing evolutionary

rates.

Discussion

The bootscanning approach has proven useful in many recombination studies during the past decades. In contrast to previous bootscanning implementations, RecDetec generates support values based on ML approaches. Support values can either be obtained from ML phylogenies producing ML bootscanning diagrams or by Quartet Puzzling producing QP-scanning diagrams. While the former performs more rigorous tree searches, the latter typically produces the ML-based QP support values more quickly. ML approaches have the advantage of employing a well-established statistical framework, which is known to produce good results in practice. Although the recombination detection tool GARD (12) could identify the boundaries of all six genomic regions, not all of these are caused by recombination events. RecDetec also found the breakpoints (Fig. 4), but also allows for analysis to find the causes of the different regions.

As mentioned, other bootscanning implementations usually reduce sequence groups to one representative or consensus sequence. While this saves running time, consensus can lead to artificial sequences which do not well reflect the features of the represented sequences (6). We show that the signal of wrongly defined groups (e.g., joining pure and recombinant sequences) can easily get lost in the consensus sequence, leaving no hint (cf. Fig. 5) that the true underlying structure contains overlapping recombinants. Such incompletely recovered histories can easily lead to wrong assumptions about the history of infectious virus strains. Keeping all sequences separate as in RecDetec has the additional advantage that trees can typically be reconstructed more accurately due to the additional information present (24). Separate sequences, on the other hand, can lose support for a joint cluster due to several reasons. The support might be lost because related recombinant sequences cluster in a joint

subtree, but also due to lack of phylogenetic signal. With separate sequences, however, RecDetec can assess the phylogenetic stability of the groups along the alignment. If groups are not stable along the alignment closer examination is required, and RecDetec offers means to examine whether the instability of groups in an alignment region was caused by recombinants or by the regional lack of phylogenetic information. Since it is crucial for bootscan analyses to define groups of sequences comprising related pure sequences or related recombinants based on prior knowledge, visualizing the phylogenetic stability of groups is a valuable tool to assess the quality of user-defined groupings. Undetected recombinant groups or subtypes may exist even in well-studied viruses such as HIV, making the assessment of reference groups for recombinants even more important. While HIV-1 subtype G was assumed to be a pure subtype for a long time, it was shown to be a recombinant form (1) and thus possibly confounding the phylogenetic signal of related sequences.

In a phylogenetic analysis, known (and yet unknown) recombinants may naturally disturb the subtree support of their parental groups because in different genomic regions they cluster with their respective relatives. To analyze such cases RecDetec allows for excluding (putatively recombinant) sequences without having to re-compute the phylogenetic reconstructions. This enables the analysis of recombination or phylogenetic stability in the presence of several recombinants and even overlapping recombinants. By studying recombinant groups separately, excluding the other overlapping recombinants in turn, it is possible to examine their relationships.

Plotting phylogenetic information and informative sites diagrams allow for quickly detecting genomic regions with very low or high divergence, thus, containing no information or accumulated noise. This, together with the group support plots, is necessary to correctly interpret bootscan results to find out whether the loss of support of relationship is caused by recombination, by effects of data quality or by wrong assumptions about reference groups.

The assessment of phylogenetic information along an alignment is certainly an important task prior to many phylogenetic analysis of other (even non-recombinant) datasets.

Finally, we point out that RecDetec is an exploratory tool to detect and analyze complex evolutionary patterns. We showed that it was possible to identify and isolate different recombinants by excluding sequences and, thus, to visualize their individual relationships.

In summary, RecDetec offers flexible ways to detect (overlapping) recombination events, to assess phylogenetic informativeness of genomic regions prior to the actual phylogenetic reconstruction or to examine the support of a joint subtree of sequences of interest along a given alignment. This would be helpful not only to study viral sequences known to recombine, but also to other phylogenetic analyses not dealing with recombination to detect why and where some subtrees are not well supported in a phylogenetic reconstruction or why some reconstructions do not work at all. RecDetec adds complementary analyses and assessments which were not available by other bootscanning implementations. Furthermore, it makes bootscanning analyses accessible to a wider range of operating systems and makes use of modern maximum likelihood methods for this kind of analysis.

Acknowledgments

The authors would like to thank Martina Kutmon for preparing an early version of the software and for suggesting its name. Financial support by the Austrian Science Fund (FWF, I760) to AvH is gratefully acknowledged.

References

[1] **Abecasis AB, Lemey P, Vidal N, de Oliveira T, Peeters M, Camacho R, Shapiro B, Rambaut A, Vandamme AM**: 2005. Recombination is confounding the

303 early evolutionary history of HIV-1: subtype G is a circulating recombinant form. J.
304 Virol. **81**:8543–8551.

305 [2] **Alcantara LCJ, Cassol S, Libin P, Deforche K, Pybus OG, Van Ranst M,**
306 **Galvão-Castro B, Vandamme AM, de Oliveira T**: 2009. A standardized frame-
307 work for accurate, high-throughput genotyping of recombinant and non-recombinant
308 viral sequences. Nucl. Acids Res. **37**:W634–W642.

309 [3] **Althaus CL, Bonhoeffer S**: 2005. Stochastic interplay between mutation and recom-
310 bination during the acquisition of drug resistance mutations in human immunodeficiency
311 virus type 1. J. Virol. **79**:13572–13578.

312 [4] **de Oliveira T, Deforche K, Cassol S, Salminen M, Paraskevis D, Seebregts C,**
313 **Snoeck J, Janse van Rensburg E, Wensing AMJ, van de Vijver DA, Boucher**
314 **CA, Camacho R, Vandamme AM**: 2005. An automated genotyping system for
315 analysis of HIV-1 and other microbial sequences. Bioinformatics **21**:3797–3800.

316 [5] **de Oliveira T, Pybus OG, Rambaut A, Salemi M, Cassol S, Ciccozzi M,**
317 **Rezza G, Castelli Gattinara G, D'Arrigo R, Amicosante M, Perrin L, Colizzi**
318 **V, Perno CF, Benghazi Study Group**: 2006. Molecular epidemiology: HIV-1 and
319 HCV sequences from Libyan outbreak. Nature **444**:836–837.

320 [6] **D'haeseleer P**: 2006. What are DNA sequence motifs? Nat. Biotechnol. **24**:423–425.

321 [7] **Efron B, Halloran E, Holmes S**: 1996. Bootstrap confidence levels for phylogenetic
322 trees. Proc. Natl. Acad. Sci. USA **93**:13429–13434.

323 [8] **Felsenstein J**: 1985. Confidence limits on phylogenies: An approach using the boot-
324 strap. Evolution **39**:783–791.

[9] **Gentzsch W**: 2007. Sun grid engine: Towards creating a compute power grid. *In* Proceedings of the 1st International Symposium on Cluster Computing and the Grid (CCGRID 2001), p. 35. IEEE Computer Society, Washington, DC, USA.

[10] **Holmes EC**: 2008. Evolutionary history and phylogeography of human viruses. Annu. Rev. Microbiol. **62**:307–328.

[11] **Jetzt AE, Yu H, Klarmann G, Ron Y, Preston BD, Dougherty JP**: 2000. High rate of recombination throughout the human immunodeficiency virus type 1 genome. J. Virol. **74**:1234–1240.

[12] **Kosakovsky Pond S, Posada D, Gravenor MB, Woelk CH, Frost SDW**: 2006. Gard: a genetic algorithm for recombination detection. Bioinformatics **22**:3096–2098.

[13] **Lemey P, Posada D**: 2009. Introduction to recombination detection. *In* **Lemey P, Salemi M, Anne-Mieke V** (eds.), The Phylogenetic Handbook: a Practical Approach to Phylogenetic Analysis and Hypothesis Testing, 2nd ed., pp. 489–514. Cambridge University Press, Cambridge.

[14] **Lemey P, Salemi M, Anne-Mieke V**: 2009. The Phylogenetic Handbook: a Practical Approach to Phylogenetic Analysis and Hypothesis Testing. 2nd ed. Cambridge University Press, Cambridge.

[15] **Lole KS, Bollinger RC, Paranjape RS, Gadkari D, Kulkarni SS, Novak NG, Ingersoll R, Sheppard HW, Ray SC**: 1999. Full-length human immunodeficiency virus type 1 genomes from subtype C-infected seroconverters in India, with evidence of intersubtype recombination. J. Virol. **73**:152–160.

[16] **Martin DP, Biagini P, Lefeuvre P, Golden M, Roumagnac P, Varsani A**: 2011. Recombination in eukaryotic single stranded dna viruses. Viruses **3**:1699–1738.

[17] **Martin DP, Lemey P, Lott M, Moulton V, Posada D, Lefeuvre P**: 2010. RDP3: a flexible and fast computer program for analyzing recombination. Bioinformatics **19**:2462–2463.

[18] **Minh BQ, Vinh LS, von Haeseler A, Schmidt HA**: 2005. pIQPNNI – parallel reconstruction of large maximum likelihood phylogenies. Bioinformatics **21**:3794–3796.

[19] **Paraskevis D, Pybus O, Magiorkinis G, Hatzakis A, Wensing AMJ, van de Vijver DA, Albert J, Angarano G, Asjo B, Balotta C, Boeri E, Camacho R, Chaix ML, Coughlan S, Costagliola D, De Luca A, de Mendoza C, Derdelinckx I, Grossman Z, Hamouda O, Hoepelman AIM, Horban A, Korn K, Kuecherer C, Leitner T, Loveday C, Macrae E, Maljkovic I, Meyer L, Nielsen C, Op de Coul ELM, Ormaasen V, Perrin L, Puchhammer-Stockl E, Ruiz L, Salminen M, Schmit JC, Schuurman R, Soriano V, Stanczak JJ**: 2009. Tracing the HIV-1 subtype B mobility in Europe: a phylogeographic approach. Retrovirol. **6**:49.

[20] **Pybus OG, Suchard MA, Lemey P, Bernardin FJ, Rambaut A, Crawford FW, Gray RR, Arinaminpathy N, Stramer SL, Busch MP, Delwart EL**: 2012. Unifying the spatial epidemiology and molecular evolution of emerging epidemics. Proc. Natl. Acad. Sci. USA **109**:in press.

[21] **Rambaut A, Grassly NC**: 1997. Seq-Gen: An application for the Monte Carlo simulation of DNA sequence evolution along phylogenetic trees. Comput. Appl. Biosci. **13**:235–238.

[22] **Salminen M, Martin D**: 2009. Detecting and characterizing inividual recombination events. *In* **Lemey P, Salemi M, Anne-Mieke V** (eds.), The Phylogenetic Handbook:

a Practical Approach to Phylogenetic Analysis and Hypothesis Testing, 2nd ed., pp. 515–544. Cambridge University Press, Cambridge.

[23] **Salminen MO, Carr JK, Burke DS, McCutchan FE**: 1995. Identification of breakpoints in intergenotypic recombinants of HIV type 1 by bootscanning. AIDS Res. Hum. Retroviruses **11**:1423–1425.

[24] **Sanderson MJ, Shaffer HB**: 2002. Troubleshooting molecular phylogenetic analyses. Annu. Rev. Ecol. Syst. **33**:49–72.

[25] **Schmidt HA, Strimmer K, Vingron M, von Haeseler A**: 2002. TREE-PUZZLE: Maximum likelihood phylogenetic analysis using quartets and parallel computing. Bioinformatics **18**:502–504.

[26] **Schmidt HA, von Haeseler A**: 2009. Phylogenetic inference using maximum likelihood methods. *In* **Lemey P, Salemi M, Anne-Mieke V** (eds.), The Phylogenetic Handbook: a Practical Approach to Phylogenetic Analysis and Hypothesis Testing, 2nd ed., pp. 181–209. Cambridge University Press, Cambridge.

[27] **Simon-Loriere E, Holmes EC**: 2011. Why do RNA viruses recombine? Nat. Rev. Microbiol. **9**:617–626.

[28] **Strimmer K, von Haeseler A**: 1996. Quartet puzzling: A quartet maximum–likelihood method for reconstructing tree topologies. Mol. Biol. Evol. **13**:964–969.

[29] **Strimmer K, von Haeseler A**: 1997. Likelihood–mapping: A simple method to visualize phylogenetic content of a sequence alignment. Proc. Natl. Acad. Sci. USA **94**:6815–6819.

[30] **Swofford DL, Olsen GJ, Waddell PJ, Hillis DM**: 1996. Phylogeny reconstruction.

In **Hillis DM, Moritz C, Mable BK** (eds.), Molecular Systematics, 2nd ed., pp. 407–514. Sinauer Associates, Sunderland, Massachusetts.

[31] **Thain D, Tannenbaum T, Livny M**: 2005. Distributed computing in practice: the Condor experience. Concurr. Comput.-Pract. Exp. **17**:323–356.

[32] **Tschager T, Schmidt HA**: 2012. DAGwoman: enabling DAGman-like workflows on non-Condor platforms. *In* Proceedings of the 1st ACM SIGMOD Workshop on Scalable Workflow Execution Engines and Technologies (SWEET 2012). ACM, New York, NY, USA.

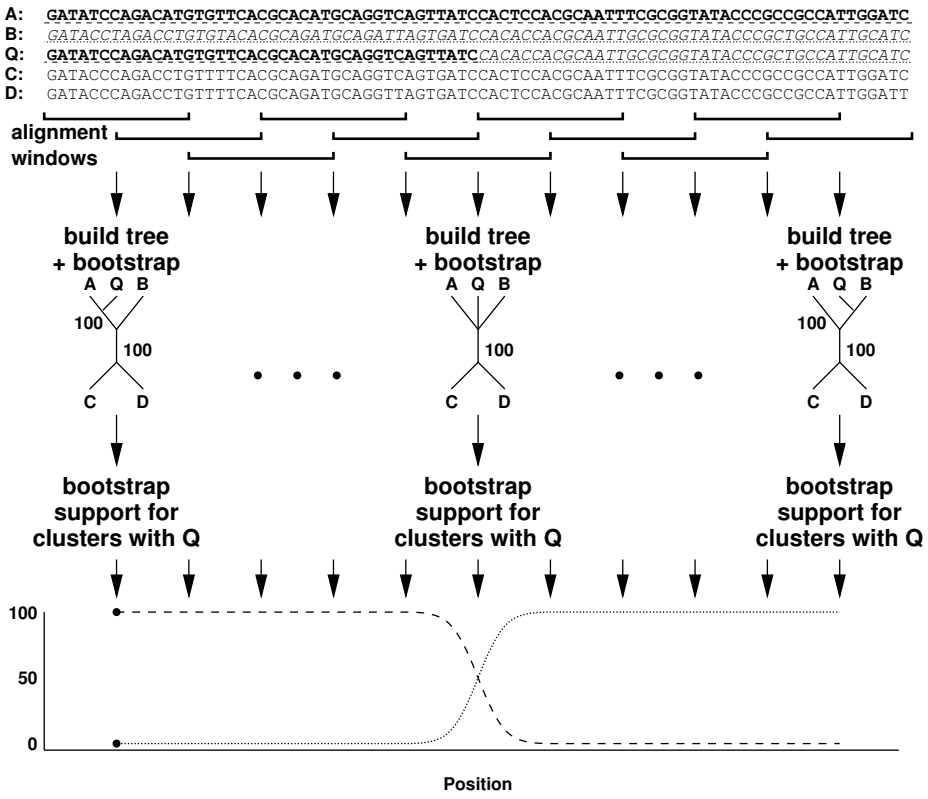

Figure 1 – Principle of bootscanning analysis.

Four reference groups $\{A\}$, $\{B\}$, $\{C\}$ and $\{D\}$ plus a query group $\{Q\}$ serve as input. Divide the alignment into overlapping windows. For each window bootstrap tree reconstruction is performed and the number of $\{Q, A\}$ and $\{Q, B\}$ branches in the bootstrap trees are evaluated for each window. The resulting bootscan plot at the bottom shows the case that Q is a recombinant containing the first half sequence of A (dashed curve or underline) and the second half of B (dotted curve or underline).

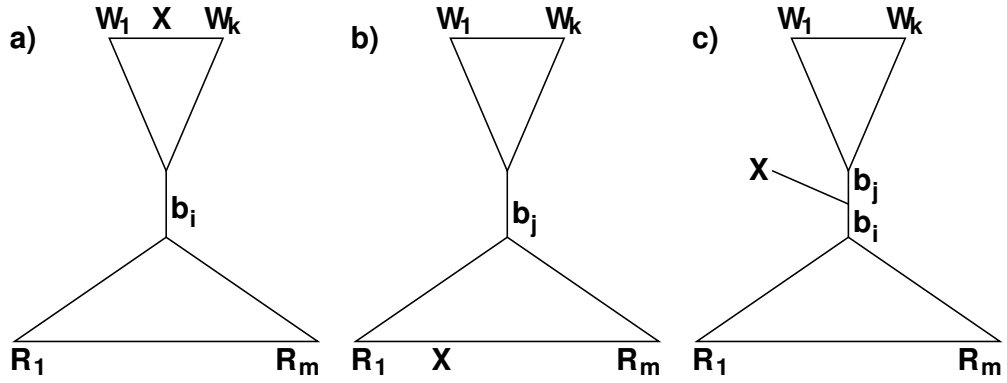

Figure 2 – Ignoring sequences in the analysis.

There are 3 scenarios when determining the support of $\{W_1 \dots W_k\}$ in reconstructed bootstrap trees excluding a sequence X. The triangles depict subtrees containing the sequences at their leaves. (a) If X is located within (but not basal to) the subtree of $\{W_1 \dots W_k\}$ then b_i is used a the bootstrap support. (b) If X is not located within the subtree of $\{W_1 \dots W_k\}$ but among (but not basal to) the remaining sequences $\{R_1 \dots R_m\}$ then b_j gives the bootstrap support. (c) In the special case that X is located between the subtrees of $\{W_1 \dots W_k\}$ and $\{R_1 \dots R_m\}$ then b_i and b_j exist at the same time in a tree. In all cases $\max(b_i, b_j)$ is used as the bootstrap support for $\{W_1 \dots W_k\}$.

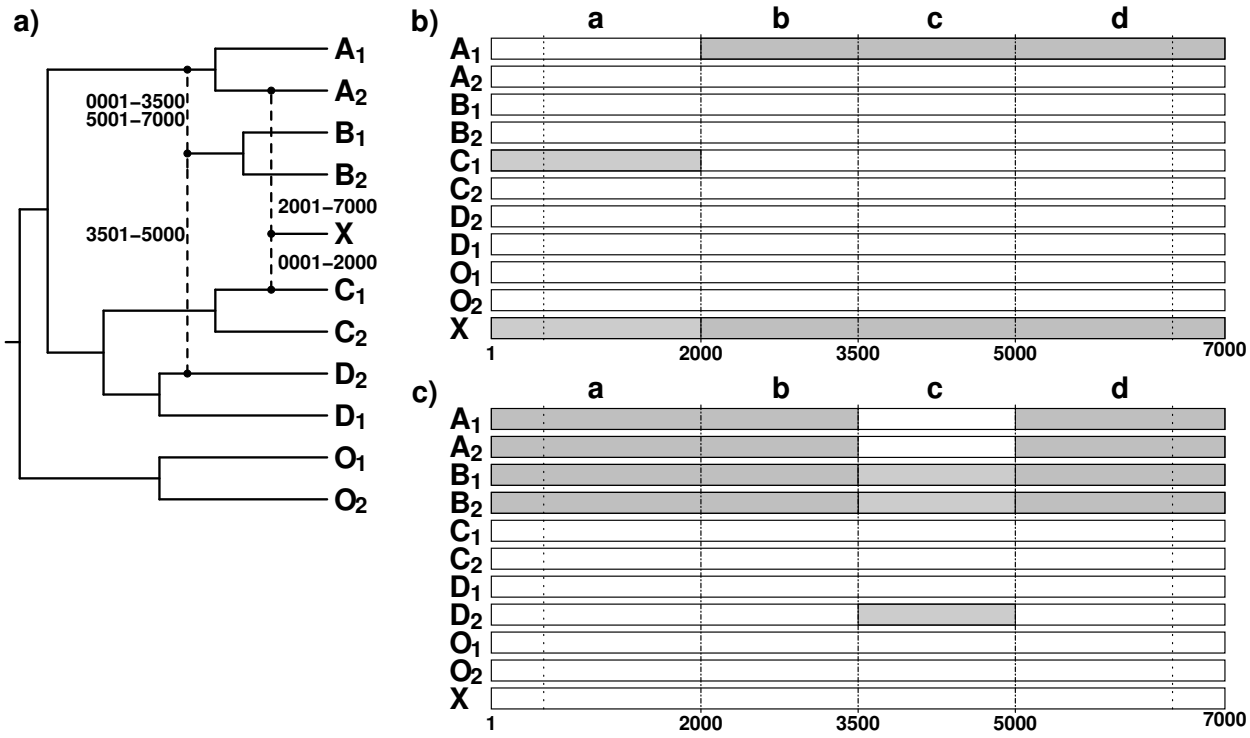

Figure 3 – Simulating an overlapping recombination scenario.

(a) The recombination graph shows the two overlapping recombinations, where the dashed lines depict the different genomic sources of the recombinant genomes. X arises from a recombination of an ancestor of $C2$ with an ancestor of $A1$, while $\{B1, B2\}$ arise from a recombination of the ancestor of $\{A1, A2\}$ with $D2$. The recombinations are overlapping because X and $\{B1, B2\}$ both share a common history with sequences from the reference group $\{A1, A2\}$ in the same regions **b** and **d**. The relationships are reflected accordingly by the shaded areas in the sequence alignment: (b) X and $C1$ share a common history in region **a**, X and $A1$ share a common history in regions **b**,**c**,**d**. (c) $\{B1, B2\}$ and $D2$ share a common history in region **c**, otherwise $\{B1, B2\}$ and $\{A1, A2\}$ share a common history.

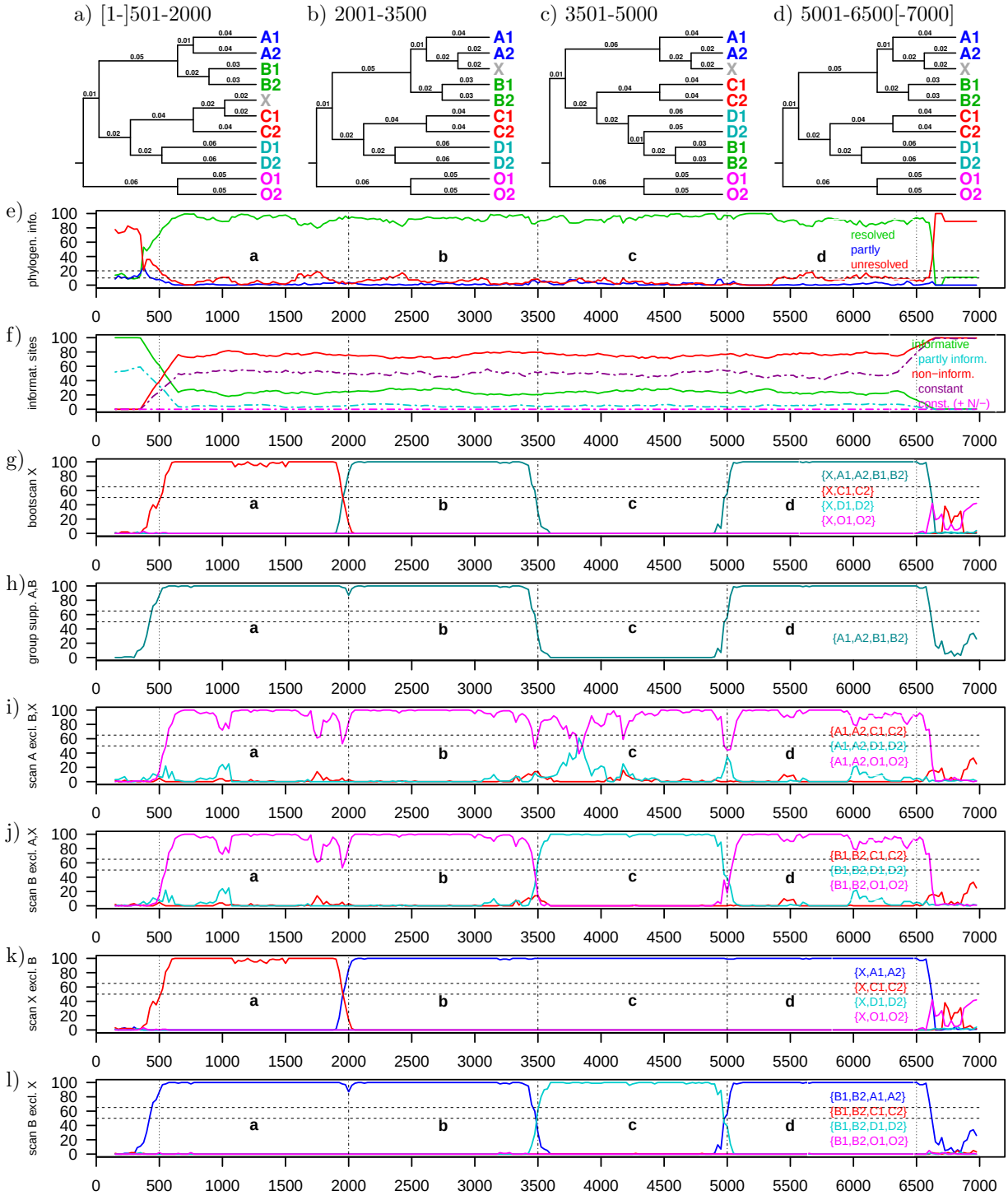
a) [1-]501-2000
b) 2001-3500
c) 3501-5000
d) 5001-6500[-7000]
A1
A2
B1
B2
X
C1
C2
D1
D2
O1
O2
e)
phylogen. info.
resolved
partly
unresolved
f)
informat. sites
informative
partly inform.
non−inform.
constant
const. (+ N/−)
g)
bootscan X
{X,A1,A2,B1,B2}
{X,C1,C2}
{X,D1,D2}
{X,O1,O2}
h)
group supp. A,B
{A1,A2,B1,B2}
i)
scan A excl. B,X
{A1,A2,C1,C2}
{A1,A2,D1,D2}
{A1,A2,O1,O2}
j)
scan B excl. A,X
{B1,B2,C1,C2}
{B1,B2,D1,D2}
{B1,B2,O1,O2}
k)
scan X excl. B
{X,A1,A2}
{X,C1,C2}
{X,D1,D2}
{X,O1,O2}
l)
scan B excl. X
{B1,B2,A1,A2}
{B1,B2,C1,C2}
{B1,B2,D1,D2}
{B1,B2,O1,O2}
a
b
c
d
0
500
1000
1500
2000
2500
3000
3500
4000
4500
5000
5500
6000
6500
7000

Fig. 4

Figure 4 – Analysis of an example with overlapping recombinations.

(a)-(d) evolutionary histories for the genomic regions 1-2000, 2001-3500, 3501-5000 and 5001-7000 (regions **a**-**d**), where in regions 1-500 and 6501-7000 the branch lengths have been extended and reduced by factors of 50, respectively. (e) Visualization of phylogenetic signal along the alignment by likelihood mapping along the alignment. (f) Visualization of the content of informative sites along the alignment. (g) ML bootscan plot for the recombinant $\{X\}$. (h) ML bootstrap support for the erroneously grouped sequences of $\{A1, A2, B1, B2\}$. (i) ML bootscan plot for $\{A1, A2\}$ only, excluding $\{X\}$ and $\{B1, B2\}$. (j) ML bootscan plot for $\{B1, B2\}$ only, excluding $\{X\}$ and $\{A1, A2\}$. (k) ML bootscan plot for $\{X\}$ when excluding $\{B1, B2\}$. (l) ML bootscan plot for recombinant group $\{B1, B2\}$ excluding sequence $\{X\}$. For more details see text. The vertical lines mark the recombination break points and the border to the regions of increased or decreased variability at the ends.

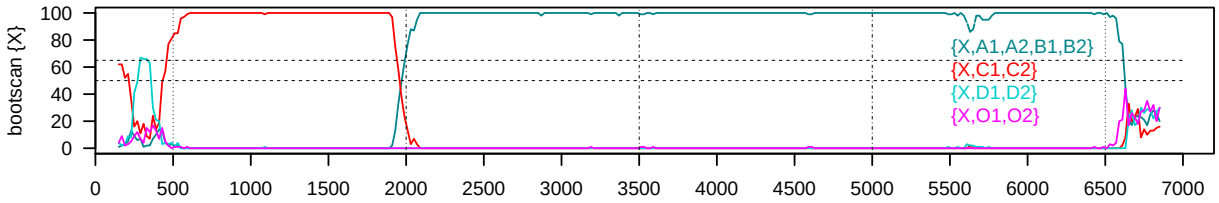

Figure 5 – Bootscan with groups condensed to sequences.

Diagram from results of SimPlot which condenses groups to consensus sequences plotted the same way and using the same window size as in Fig. 4 for comparability.