HIV Classification using Coalescent Theory

Ingo Bulla[1,*], Anne-Kathrin Schultz[1], Fabian Schreiber[1,2], Ming Zhang[3], Thomas Leitner[3], Bette Korber[3,4], Burkhard Morgenstern[1], Mario Stanke[1]

1 Abteilung Bioinformatik, Institut für Mikrobiologie und Genetik, Georg-August-Universität Göttingen, Goldschmidtstr. 1, 37077 Göttingen, Germany

2 Department für Geo- und Umweltwissenschaften & GeoBioCenter LMU Ludwig-Maximilians-Universität München, Richard-Wagner-Straße 10, 80333 München, Germany

3 Theoretical Division, Los Alamos National Laboratory, Los Alamos, NM 87545, USA

4 The Santa Fe Institute, Santa Fe, NM 87501, USA

∗ Corresponding author (ibulla@uni-goettingen.de)

Abstract

Existing coalescent models and phylogenetic tools based on them are not designed for studying the genealogy of sequences like those of HIV since recombinants with multiple cross-over points between the parental strains frequently arise in HIV. Hence, ambiguous cases in the classification of HIV sequences into subtypes and Circulating Recombinant Forms (CRFs) have been treated with ad hoc methods in lack of tools based on a comprehensive coalescent model accounting for complex recombination patterns.

We developed the program ARGUS that scores classifications of sequences into subtypes and recombinant forms. It reconstructs Ancestral Recombination Graphs (ARGs) that reflect the genealogy of the input sequences given a classification hypothesis. An ARG with maximal likelihood is approximated using a Markov Chain Monte Carlo approach. ARGUS was able to distinguish the correct classification with a low error rate from plausible alternative classifications in simulation studies with realistic parameters. Comparison of ARGUS with the (to our knowledge) only available other tool suitable for the considered task, resulted in ARGUS performing noticeably better on the examined test data set. We applied our algorithm to decide between two recently debated alternatives in the classification of CRF02 of HIV-1 and find that CRF02 is indeed a recombinant of subtypes A and G.

Availability: ARGUS is implemented in C++ and the source code is available at `http://gobics.de/software`.

Supplementary Information: Details of the algorithm are described in the online supplementary material.

Author Summary

Accurate classification of HIV and other viral sequence data is crucial for epidemiological studies and for developing potential drugs and vaccines. This task is challenging, however, since HIV is one of the most genetically variable organisms known. The M group of HIV-1, which is responsible for the HIV pandemic, is divided into subtypes. Intersubtype recombination is extremely common in HIV-1. Identifying intersubtype recombinants is therefore important from many perspectives, giving insights into such issues as molecular epidemiology, viral evolution, and indirectly, the frequency of dual infections.

Most methods for these tasks are based on a subtype-wise alignment and depend on the underlying classification system of HIV-1. The current one has been developed step by step during the process of discovery of new HIV sequences. Due to the history of creation of the current HIV-1 nomenclature, it contains several inconsistencies. Moreover, the current classification system is somehow arbitrary like all complex classification systems that were created manually. To this end, we developed an algorithm addressing a subproblem of the task to generate the classification system of HIV systematically.

Introduction

A coalescent model incorporating recombination was first introduced by Hudson (1). In the presence of recombination, the genealogy of a set of sequences can be represented as a so-called Ancestral Recombination Graph (ARG) rather than a tree (2). In Human Immunodeficiency Virus (HIV) recombination is frequent and recombinant forms of the virus that spread are called Circulating Recombinant Forms (CRFs). In current nomenclature, sequences of the epidemiologically most relevant clade, HIV-1 Group M, are classified into 9 subtypes and 43 CRFs (see LANL-database, http://www.hiv.lanl.gov). The CRFs have (usually multiple) recombination breakpoints between unrecombined segments of the 'pure' parental subtypes.

Algorithms for subtype classification and breakpoint detection of HIV-1 sequences are based on the classification system of HIV-1 (3). Hence, their quality highly depend on this system. But due to the evolution of HIV-1 nomenclature, developed in real time in conjunction with emerging knowledge of the global diversity of the virus, the current nomenclature has anomalies. E.g., the phylogenetic distance between subtypes B and D is relatively small compared to that of other pairs of subtypes. In fact, it is more like the distance of a pair of subsubtypes (3).

Furthermore, several questions about the current classification system of sequences are unanswered or debated: There are 8 complete and over 400 partial HIV-1 Gr. M genomes in the LANL-database which are "undefined", i.e., belong neither to a subtype nor to a CRF (this is due to many reasons, e.g., parental sequences have not been identified yet). Additionally, in the nomenclature system CRF02 is considered to be a recombinant of subtype A and G, but recently some evidence has been presented in the literature suggesting that CRF02 would actually be better described as a subtype and subtype G as a recombinant (4; 5). A few years ago it was also debated whether CRF01 is in fact a pure subtype (6).

Moreover, assignments in the database are reflected on diverse classification strategies employed by different investigators in the primary literature. Therefore it is desirable to have tools for classifying the HIV sequence set systematically based on a comprehensive coalescent model accounting for complex recombination patterns.

While the problem of recombination has been well documented in HIV-1, recombination also occurs in the other lentiviruses. Moreover, intra-segmental recombination has been reported from a large variety of other viruses, e.g., coronaviruses, flaviviruses, alphaviruses, rotaviruses, influenzaviruses, hantaviruses, arenaviruses and avian oncoviruses (7; 8; 9; 10; 11; 12; 13; 14; 15; 16). Furthermore, recombination plays also a role in other species, like bacteria (17; 18). Thus, it is possible that many virus systems would benefit from a systematic classification that explicitly includes recombination (19).

Our work addresses the simpler subproblem to score classifications of given input sequences of some virus species. 'Classification' here denotes a partitioning of the input sequences into several subtypes and, if applicable, CRFs. To score a classification, we reconstruct ARGs of the input sequences under restrictions determined by the classification (see Figure 2 for an example). These restrictions are imposed in order to ensure that the reconstructed ARGs do not contradict the classification under consideration. Then, we find the ARG with maximal likelihood by means of Markov Chain Monte Carlo methods. The likelihood of the most likely ARG is interpreted as a score for the classification.

Finally, we apply ARGUS, the implementation of our algorithm, to decide between the two main hypotheses regarding CRF02.

Although up to now about 50 tools for classification of HIV-1 genomes, identification of recombinants, and precise breakpoint detection have been developed – e.g. RIP (http://www.hiv.lanl.gov/content/sequence/RIP/RIP.html), REGA (20), Recco (21) –, to our knowledge, our particular problem fits only roughly into the scope of one algorithm (VisRD, described in 5). A comparison with respect to performance and scope between our algorithm and VisRD is carried out in Section "Results – Comparison".

Moreover, the software package Lamarc (22) allows for sampling ARGs, but it assumes that recombination events only involve one breakpoint. However, HIV recombinants usually have more than one breakpoint. Moreover, Lamarc does not perform an explicit breakpoint detection, but tries to find them

by chance. Although this approach is suitable for most situations, it will not lead to satisfying results in case of highly recombining viruses with multiple breakpoints.

Up to now, researchers confronted with the task to classify genome sequence data strongly affected by recombination with multiple crossovers normally segmented the aligned genomes in recombination-free parts (Simplot (23) is a widespread tool for this task) and analyzed these parts with traditional phylogenetic tools. The information stemming from different parts of the genome had to be assembled by ad hoc methods (4; 24; 25).

Methods

Overview

A classification is a partition of the input sequences $\{s_1, \ldots, s_n\}$ into disjoint sets $P_1, \ldots, P_{m_p}$ and $R_1, \ldots, R_{m_r}$, where each set P_i contains the sequences belonging to the ith (pure) subtype and each set R_i the ones belonging to the ith CRF. An example of a classification is given in the upper part of Figure 1. The algorithm takes the input sequences and a hypothesis classification and scores the classification by constructing a likely ARG for the sequences under this classification.

Notice that CRFs (in our notation) are allowed to be composed of less than three epidemiologically independent sequences violating the definition of a CRF (see 3). Nevertheless, in applications they will normally be composed of sequences allowing for calling them CRFs according to (3).

Preparing step

All input sequences are aligned and only positions composed entirely of non-gaps are further considered. Then the breakpoints of the input sequences designated as recombinants are identified and the subtypes of the resulting segments are classified by jumping profile Hidden Markov Model (jpHMM), introduced by Schultz et al. (26; 27). An example of this step is shown in the lower part of Figure 1. We use jpHMM for breakpoint detection since we need a fully automatically working tool.

jpHMM is a probabilistic generalization of the jumping-alignment approach introduced by Spang et al. (28). Given a partition of the aligned input sequence family into known sequence subtypes (in our case $P_1, \ldots, P_{m_p}$), this model can jump between states corresponding to these different subtypes, depending on which subtype is locally most similar to a query sequence. Jumps between different subtypes are indicative of intersubtype recombinations.

More strictly, assume the subtypes occurring in the considered classification are denoted by $S_1, \ldots, S_{m_p}$ and the length of an alleged recombinant sequence is ℓ. Then jpHMM provides a mapping $f : \mathbb{N}_\ell \to \{S_1, \ldots, S_{m_p}\}$, i.e., f assigns a subtype to each position of the query sequence. This mapping will also be called the segmentation of the query sequence.

For each CRF a single segmentation is calculated by applying jpHMM to one of the sequences of the CRF. See Section "Details – jpHMM" for additional details.

Coalescent model

The non-recombinant coalescent genealogy was introduced by Kingman (29; 30) and extended to the recombinant case by Hudson (1). Apart from Hudson's back-in-time algorithm, the spatial algorithm of Wiuf et al. (31) allows for constructing the coalescent with recombination process. We apply a formulation of the recombinant genealogy similar to the one introduced in (22). Like Kuhner et al. (22), we discard lineages not contributing at least one site to the sample and we discard recombination events that do not separate at least two sites contributing to the sample. The difference of our formulation consists in allowing for multiple breakpoints. I.e., the probability of a recombination event taking place is the same as in (32) and (22), but multiple breakpoints can occur due to the recombination event. Adapting the

framework of Hudson (32) and Kuhner et al. (22) to our setting is a simplification, but justified by the fact that the affected quantities play a very minor role in our algorithm. Using jpHMM with a sufficiently low jump probability to predict the breakpoints prevents the recombination events from yielding too many breakpoints, which would lead to strongly fragmented sequences. As customary (33; 34), we call a recombinant genealogy an Ancestral Recombination Graph (ARG).

The likelihood of an ARG G is the product of the probability of the (sequence) data D with respect to the ARG, $P(D|G)$, and the probability of the ARG given the evolution parameters, $P(G|\Theta, r)$. Here $\Theta = 2N\mu$ and $r = C/\mu$, where N is the effective population size, C is the per-site recombination rate and μ is the per-site neutral mutation rate. See Section "Details – Coalescent model" for additional details.

Restrictions to the genealogy

We approximate the most likely ARG G of the input HIV sequences, where we impose restrictions to the approximating ARG $\hat{G}$ according to the classification to be scored. The score of the classification is then given by $P(D|\hat{G})$. In detail, the restrictions are given in the following (see Figure 2 for the symbols).

In the lower part of the ARG, sequences of subtypes which are composed of more than one sequence or are present in the genome of a CRF are only allowed to coalesce and only with sequences of the same subtype. Sequences of multi-sequence CRFs ($\times$) are only allowed to coalesce and only with sequences of the same CRF. Sequences from different subtypes can only coalesce if they are the only sequence of their subtype left ($\star$). Furthermore, coalescing sequences of the same subtype ($\times$) generate a sequence belonging to the same subtype as their children.

The last (or only) sequence of a CRF ($\times$) is obliged to recombine. (Multiple) breakpoints have to be chosen such that the parental subtypes get separated and recombination events have to take place until all parental subtypes are separated. The final sequences generated by this process only contain one subtype. This subtype is interpreted as the subtype of these (final) sequences. They are allowed to coalesce with other sequences of the same subtype. No sequence of their subtype is allowed to coalesce with a sequence of another subtype before having coalesced with these sequences.

This set of restriction rules (a more formal description is given in the supplements) is imposed in order to enforce ARGs which are plausible under the condition that the underlying classification is a reasonable one. This would not be the case, e. g., if subtype A sequences coalesced with subtype C sequences before coalescing within their respective subtype or the two CRF1 sequences recombined independently of each other without coalescing first.

These rules imply the assumption that intra-subtype recombinations and the recombinations occurring before the subtypes were formed are negligible. The negligence of the first type of recombination simply means that we do not aim to obtain a classification resolving finer than the level of subtypes. Neglecting the second type can be justified by the HIV subtypes being separated by founder effects (35). Since alternative methods are restricted by far stronger assumptions, we refrain from analyzing our simplifications in more detail.

Markov Chain Monte Carlo

After having constructed an initial ARG, the likelihood of the ARG is iteratively maximized using Markov Chain Monte Carlo (MCMC) methods. More precisely, in a first step an initial ARG G_1 is sampled from the conditional coalescent distribution. Thereafter, by modifying the ARG slightly and accepting or rejecting these modifications based on how they affect the likelihood of the ARG, a Markov chain $\{G_i\}_{i\in\mathbb{N}}$ is generated. If no significant improvement of the likelihood seems achievable any more, the chain is stopped at the current chain position n and the probability of the data D given the most likely ARG

$$\max_{i=1,\ldots,n} P(D|G_i) \tag{1}$$

is interpreted as a score for the classification (see Section "Details – Score" for details). We stop the MCMC algorithm at step $n > M$ if

$$\frac{\max_{i=1,\ldots,n} P(D|G_i)P(G_i|\Theta,r,R)}{\max_{i=1,\ldots,n-M} P(D|G_i)P(G_i|\Theta,r,R)} < 1+\delta$$

with $M = 1000$ and $\delta = 10^{-8}$. Here $P(G|\Theta,r,R)$ denotes the probability of the ARG G given the parameters Θ and r and restricted to our rules R. Due to the small influence of this quantity we used $P(G|\Theta,r)$ instead of $P(G|\Theta,r,R)$ for the sake of simplicity where appropriate. Normally, the MCMC algorithm is carried out several times for different initial ARGs. Details about the MCMC algorithm are explained in the Supplementary Material. The parameter Θ was estimated with Lamarc 2.1.3 (36).

Extension to unknown subtypes

In the genome of several CRFs, segments are commonly classified to belong to an unknown subtype. In order to address classification problems involving unknown subtypes, we extend the restriction rules: We additionally interprete a sequence generated by a recombination event and belonging only to one, unknown subtype as the only sequence of its subtype left (cf. rules for coalescent events in Section "Restrictions to the genealogy"). This case is illustrated in Figure 3.

jpHMM is not able to detect segments belonging to an unknown subtype and, to our knowledge, up to now no tool is available for automatically segmenting sequences into known and unknown subtypes. Hence, segments belonging to an unknown subtype have to be added manually in the classification after the application of jpHMM.

Details

jpHMM

Applying jpHMM only to one sequence of each CRF (and assigning the calculated segmentation to all sequences belonging to this CRF) could seem questionable if jpHMM yielded strongly dissimilar results for different sequences s_a and s_b of the same CRF. But such diverging results of jpHMM would also indicate that the whole classification is rather poor since one should obviously not assign s_a and s_b to the same CRF. Hence, the behavior of ARGUS to reconstruct a genealogy of low likelihood in this case (due to assigning an inappropriate segmentation to either s_a or s_b) will correct for the restricted application of jpHMM.

Coalescent model

In coalescent theory, time is traversed backwards starting at the tips, generating genealogical events (i.e. coalescent events and recombination events) according to their rate, until only one node is left (called the root node). The rate of coalescence is $k(k-1)/\Theta$, where k is the number of active lineages, and the rate of recombination is rs, where s is the length of the genome region in which a valid recombination event might occur, summed over all lineages (valid means not to be discarded because it does not contribute to the sample, cf. Section "Overview – Coalescent model"). The prior probability of the ARG G is

$$P(G|\Theta,r) = \left(\frac{2}{\Theta}\right)^{N_C} r^{N_R} \exp\left[\sum_i -\left(\frac{k_i(k_i-1)}{\Theta} + rs_i\right) t_i\right]$$

where N_C is the number of coalescent events and N_R the number of recombination events in G, k_i the number of active lineages between the ith and $(i+1)$th genealogical event, s_i the sum of valid sites in that interval, t_i the length of the time interval between the ith and $(i+1)$th genealogical event (see 22).

Since we assume that mutations at different sites are independent, $P(D|G)$ can be easily calculated sitewise (37). For the mutation process, a General Time Reversible (GTR) model (38) with mutation rate varying among the sites is used. The variation is modeled by a gamma distribution (39) and the parameters of the GTR model were estimated with Findmodel (www.hiv.lanl.gov/content/sequence/findmodel/findmodel.html).

Scoring

Instead of Equation 1 one could also interpret the likelihood of the ARG

$$\max_{i=1,\ldots,n} P(D|G_i)P(G_i|\Theta, r, R)$$

as a score for the classification. Nevertheless, using $P(G|\Theta, r, R)$ to score a classification makes only sense if the tip sequence data is sampled randomly. Obviously, this is absolutely not the case for our applications. Hence, we neglect $P(G|\Theta, r, R)$ and only consider $P(D|G)$ (Users who find a way to estimate Θ and r for their data can incorporate this knowledge manually). Anyways, normally the difference of $P(D|G)P(G|\Theta, r, R)$ for different classifications is strongly dominated by $P(D|G)$.

Results

Test settings

In order to test ARGUS, three different settings are tested:

T1 a representative selection of 40 HIV-1 Gr. M sequences from subtypes A-K

T2 a classification involving all features offered by ARGUS except unknown subtypes

T3 two classifications corresponding to the situations that

- CRF02 is a CRF and G a subtype or
- CRF02 is a subtype and G a CRF,

resp.

The first test setting is intended to show the ability of ARGUS to construct a phylogenetic tree (without recombination). This is a subproblem also arising when reconstructing an ARG and, hence, must be accomplishable by ARGUS. The second setting demonstrates the ability of ARGUS to differentiate between similar classifications of real-world complexity. We refrain from incorporating unknown subtypes since this would prohibit performing the test fully automatically. Finally, the last setting shows the applicability of ARGUS to the task of further analyzing subtype G and CRF02, to determine whether they are more likely to be recombinants or ancestral subtypes; this question was raised by earlier analysis in (4; 5).

Parameter estimation

The model uses two different mutation rates since increasing their number does not improve the results, whereas a constant mutation rate performs considerably worse (data not shown). The per-site recombination rate C is set to 10^{-4} (40). The parameter Θ is estimated by applying Lamarc 2.1.3 to 10 randomly chosen HIV-1 Gr. M sequences, classified as pure subtypes in the LANL HIV sequence database. Disabling recombination and growth, Lamarc yields $\Theta = 1.25$. The gamma distribution parameter of the GTR is estimated to be $\alpha = 0.416$ by Findmodel. The length of the simulated sequences is 8500 bp, which is approximately the length of the HIV-1 Gr. M sequences used in the application in Section "Empirical data" (after removal of gap-affected positions).

Simulation studies

T1 - Without recombination

As an initial test and to verify that the method can correctly perform the easier task of constructing a phylogenetic tree (without recombinations), 40 representative HIV-1 Gr. M sequences (7 from subtype A, 7 B, 11 C, 3 D, 3 F, 3 G, 2 H, 2 K, 2 J) are chosen (using FigTree, see http://tree.bio.ed.ac.uk/software/figtree/). Then ARGUS is applied to score the trivial classification (i.e., all sequences belong to one 'subtype'). The most likely ARG achieved by the MCMC algorithm is compared to the phylogenetic tree in Figure 7 of (26). As desired, in our tree all sequences belonging to the same subsubtype or subtype, resp., first coalesce with the other sequences of the subsubtype or subtype, resp., before coalescing with sequences from another subsubtype or subtype, resp. The remaining sequences of the subtype cluster like follows:

$$((((A,G)(H,J))((B,D)(F,K))),C)$$

In (26) the tree has the form

$$((((A,G),J),(C,H)),((B,D)(F,K))).$$

We consider this sufficiently similar given that the branch lengths before the split into subtypes J, C, and H are very short.

T2 - With recombination

We choose two original classifications and for each original classification a number of alternative classifications for testing (Figures 4, 5). We perform the following steps for each original (true) classification in our test setting:

1. Simulate an ARG according to the original classification
2. Simulate the mutation process on the ARG (from the root downwards), thereby obtain simulated tip sequences
3. Score both the original as well as one or more plausible alternative classifications using the simulated tip sequences

When the original classification scores higher than the test classifications, this indicates that ARGUS works for the analyzed setting. In the first part, we test 9 classifications of 15 sequences, in the second part 6 classifications of 15 sequences (Figures 4 and 5).

The ARGs are simulated by sampling them with respect to the coalescent distribution, conditioned on the ARG fulfilling the restrictions imposed by the original classification. Notice that sequence data stemming from such ARGs in general does not pose the typical application situation for ARGUS: Normally a classification algorithm is applied to (sub-)species well separated by founder effects (35). Nevertheless, the chosen testing method allows for highlighting the boundaries of applicability of ARGUS.

In the first test, the original classification has three pure subtypes and two CRFs with three sequences each. The first CRF is equidistantly segmented into three parts belonging to the first two subtypes and the second CRF is equidistantly segmented into ten parts from all three subtypes. The first tested classification (denoted by C1.1) matches the original classification. The other eight (false) classifications (denoted by C1.2-C1.9) are slight modifications of the original one:

- in C1.2 and C1.3, resp., the fourth triple does not belong to a CRF but to the first and second subtype, resp.,
- in C1.4 the fourth triple does not belong to a CRF but constitutes a fourth subtype,

- in C1.5 and C1.6, resp., the last triple belongs to the first and second subtype, resp.,
- in C1.7 the second and third triple belong to the same subtype,
- in C1.8 all triples belong to distinct subtypes,
- in C1.9 the third triple constitutes a third CRF.

Notice that one could make the task more difficult for ARGUS by also testing classifications only differing from C1.1 by one or two sequences (and not a triple), but we suppose that in real-world applications the input sequences are in general groupable with respect to similarity.

In the second test all triples belong to different subtypes. The original classification again constitutes the first test classification (denoted by C2.1). The other five classifications (C2.2.-C2.6) differ from the original one by one triple being assigned to a CRF.

Especially for the first test, the choice of tested classifications is somehow arbitrary. We plan to overcome this drawback by traversing the space of possible classifications automatically in the future.

Notice that comparing two classifications both having no CRFs is not always reasonable. E.g., the classifications assigning the same subtype to all sequences and a different one to each sequence, resp., always score highest among the CRF-free classifications (assuming the MCMC algorithm finds the global maximum).

For both tests of T2 we simulate 9 ARGs and for each ARG we simulate 5 sets of tip sequences, yielding 90 individual tests. The results are shown in Figures 6 and 7.

ARGUS computed a higher score for the original classification than for the alternative classifications in all cases except the following ones. For the first test, ARGUS fails for 2 out of 9 simulated ARGs to always (i.e. for all simulated tip sequences sets) score the original classification highest: For one tip sequences set of the 5th ARG, C.1.7 scores higher than C.1.1 and for one tip sequences set of the 7th ARG jpHMM fails to find any breakpoint in one of the CRFs of C1.1.

For the second part, ARGUS fails for 1 out of 9 simulated ARGs to always score the original classification highest: C.2.2 scores highest for one tip sequences set of the 5th ARG. jpHMM always finds breakpoints in both CRFs of C2.1.

T3 - Simulation of CRF02 case

In Section "Empirical data", we will apply ARGUS to the question whether subtype G is actually a pure subtype and CRF02 is a recombinant form (like assumed in 3) or G is a recombinant form and CRF02 is a pure subtype (like claimed in 4). The classification systems which best describe the genealogical situation assumed in (3) and (4), resp., are given in Figure 8. We choose to use two sequences per subtype and CRF, resp., since at the time of the beginning of our project, only two full-length sequences of subtype J were available and jpHMM occasionally experience difficulties in case of varying number of sequences per subtype or CRF, resp.

In order to verify whether ARGUS can theoretically distinguish between these two concrete classifications, we first simulate the data and apply the same testing method as in Section "T2 - With recombination". This simulation test is in preparation to the application using the real sequences in Section "Empirical data". More precisely, our test is composed of two parts: In the first one, C.02 is the original classification, whereas in the second one C.G is the original classification. For both parts C.02 as well as C.G are used as test classifications. We generate 10 ARGs and simulate 5 sets of tip sequences for each ARG (for both parts).

When C.02 is scored in the first part, the position of the segment belonging to the unknown subtype has to be provided manually. Of course, this implies that ARGUS is provided with a part of the true classification instead of having to estimate it, which facilitates its task. Nevertheless, the introduced bias is most likely small as the unknown region is only short.

In order to characterize the ARGs with respect to how feasible the task is to decide which classification is the original one, we introduce two simple measures, explained in Figure 9: separating and noise distance.

The results are shown in Figure 10 and 11. Apparently our theoretical considerations about the explanatory power of the ratio of separating and noise distance is supported by these testing results: In the first part (C.02 being the original classification), ARGUS fails for 1 out of 10 simulated ARGs to always (i.e. for all simulated tip sequences sets) score the original classification highest (Applying ARGUS to C.G, for 4 ARGs jpHMM does not predict any recombination in subtype G for any simulated tip sequences set). The ARG yielding misclassifications is the one with the lowest distance ratio. For this ARG, ARGUS fails for 4 tip sequences sets with the wrong classification scoring at most 26 points better than the original one.

In the second part (C.G being the original classification), ARGUS also fails for 1 out of 10 simulated ARGs to always score the original classification highest. The ARG yielding misclassifications is the one with the 2nd lowest distance ratio, with the ones with the lowest and the 5th lowest ratio being quite close to failing. For the 2nd ARG, ARGUS fails for 4 tip sequences sets with the wrong classification scoring at most 11 points better than the original one. For both parts, jpHMM always succeeds in detecting breakpoints in the alleged CRF of the original classification.

It should be stated that for the setting presented in Section "T2 - With recombination", no significant relation between separating and noise distance and the reliability of ARGUS was observable (data not shown). Due to the simplicity of these distance measures and their obvious shortcomings, it has to be expected that they fail their purpose for some settings.

Empirical data

In order to decide which classification from Figure 8 describes the real situation better, we randomly choose two full-length sequences from subtype A, G, H, and J and from CRF02. Applying ARGUS to them (performing 30 runs of the MCMC algorithm for each classification), yields a maximum score of -33,513 for C.02 and -33,714 for C.G. During the tests in Section "Simulation studies – T3" we saw that - even under worse circumstances - the difference between the score of the wrong classification and the score of the right one never exceeded 26 (in case of a misclassification). Since the score of C.02 is higher than the one of C.G by more than 200, ARGUS indicates that the classification currently in use is the preferable one.

Since $P(D|G) = \prod_i P(D_i|G)$ where D_i is the tip sequence data at position i, we can easily analyze which part of the genome supports which classification better. To this end, we plot $\log P(D_i|G) - \log P(D_i|G')$ with G and G' the most likely reconstructed ARGs of the two considered classifications (see Figure 12). Moreover, ARGUS provides the option to visualize the most likely ARG found by the MCMC algorithm using Graphviz (http://www.graphviz.org). For C.02, this visualization is shown in Figure 13 in a processed form.

Abecasis et al. (4) applied monophyly rules to determine whether G or CRF02 is a subtype. They obtained conclusive results for the region corresponding to positions 4393-4802 in HXB2, favoring CRF02 to be a subtype. This region corresponds to positions 3494-3928 in our analysis. Figure 12 shows that our results do not support this conclusion. One has to keep in mind that our method strongly differs from the one used by Abecasis et al. (4). Moreover, on the one hand, Abecasis' method makes use only of a small part of the available information and applies a model of low complexity. On the other hand, our method is only able to test two alternative classifications fitting into the framework of ARGUS. In particular, the genome of CRFs is not allowed to be composed of other CRFs but only subtypes and no recombination events near to the root of the ARG are allowed. Both simplifying assumptions are violated in the real evolutionary history of HIV-1.

Running time

For the test settings discussed in Section "Simulation studies – T2" and "Simulation studies – T3", the running time (on a Dell PowerEdge 2650 2, 80GHz/512 KB Xeon) of the MCMC algorithm lies between 6.9 and 394 min with a mean of 45 min and quartiles of 26, 37, and 55 min. The running time moderately depends on the data: Considering all tested pairs of original and test classifications separately, the minimal mean of the running time is 28 min, the maximal 61 min. The running time of jpHMM is described in (26).

Comparison

We restrict our comparison to the version 3.0 of VisRD since, to our knowledge, VisRD is the only software tool which is suitable to address the task carried out by ARGUS. For the comparison we use the first 5 ARGs generated according to classification C1.1 in Section "Simulation studies – T2". Due to the fact that VisRD has to be operated interactively, we restrict ourselves to a smaller test setting than the ones used in Section "Simulation studies". For each ARG, we simulate three sets of tip sequences with a genome length of 8500 bps, using the simpler Jukes-Cantor model since VisRD does not allow for a GTR model. We apply the taxon ranking analysis of VisRD to these 15 sets of simulated tip sequences with default windows and step size, generating 100 replicate data sets per set using Random Shuffling. VisRD finds no recombination at all. Moreover, the sequence triple VisRD determines to be the most likely to be a recombinant is not one of the two recombinant triplets for 11 out of 15 sets of simulated tip sequences.

Discussion

We presented ARGUS, a classification tool for recombining viruses, particularly HIV. Up to now, researchers intending to classify sequences of strongly recombining viruses had to analyze the sequences separately by segmenting their genome in recombination-free parts and applying traditional phylogenetic tools to them. The information stemming from different parts of the genome had to be assembled by ad hoc methods (if even possible at all). Here ARGUS offers an alternative by applying sophisticated coalescent theory and MCMC based methods and incorporating much larger parts of the available information in an integrated and model-based way.

The recently developed version 3.0 of VisRD does not perform well on the datasets analyzed in this article. Nevertheless, one has to keep in mind that the approach used by VisRD is in principle not adequate for a small number of sequence groups. In fact our test setting is the smallest possible for which the taxon ranking of VisRD can be applied. Since applying ARGUS on a very large number of sequence groups is prohibitive with respect to running time (at least in the current implementation), we can conclude that the scopes of VisRD and ARGUS are roughly exclusive.

Moreover, the application range of ARGUS is limited in two directions: First, ARGUS is not designed to rank different classifications not containing any recombinants. This is due to the fact that the two classifications assigning the same subtype to all sequences and assigning a different subtype to each sequence, respectively, achieve the highest likelihood among all CRF-free classifications by definition. Second, when applying ARGUS one has to keep in mind that there might be very plausible classifications which do not fit into the framework of ARGUS, i.e., classifications incorporating intra-subtype recombination and recombination events in the early history of the ARG.

In this study, we first verified in different test settings that ARGUS possesses the ability to reliably identify the most appropriate classification in most investigated cases. Due to the character of the test method, we can conclude that for input sequences stemming from (sub-)species well separated by founder effects – like HIV-1 – ARGUS classifies correctly with very high probability. Afterwards, we applied ARGUS to real-world HIV-1 Gr. M data in order to address the intensively debated question

whether CRF02 is truly a CRF or rather the alleged subtype G is one. Our results show that the former classification explains the data better.

The fact that we had to run the MCMC algorithm with up to 50 initial ARGs to achieve satisfying results, shows that the MCMC algorithm is not often able to find the global maximum directly, very probably due to getting trapped in a local maximum. We plan to overcome this problem by applying Metropolis Coupled Markov Chain Monte Carlo (MC^3) methods (41).

A standard task after sequencing a new HIV genome is subtyping as performed by jpHMM and other tools: The genome is segmented into regions that are each related to one known pure subtype of HIV. A future application of ARGUS will be to vote between the results of several subtyping tools in case they disagree on the correct classification of the query sequence (easily carried out by replacing the recombination prediction of jpHMM by the ones of the other subtyping tools). This application is a special case for the model in which two or more classifications are compared that differ only in the recombination pattern of one sequence, the other sequences all being pure subtypes. While ARGUS does not search for a recombination pattern itself, its comprehensive model is well suited to compare such patterns.

In the near future, we plan to incorporate additional rules (like intra-subtype recombination and recombination near to the root) into ARGUS and allow for growth of the population and temporally spaced sequence data. In the long run, we will generalize our approach and perform an unconstrained search of the space of ARGs without requiring prior classification of the input sequences into subtypes and recombinant forms.

Acknowledgements

This work was supported by the Deutsche Forschungsgemeinschaft [STA 1009/5-1 to B.M. and M.S.] and an NIH-DOE interagency agreement [Y1-AI-8309 to B.K. and T.L.].

References

1. Hudson RR (1983) Properties of a neutral allele model with intragenic recombination. Theor Pop Biol 23: 183-201.

2. Griffiths RC, Marjoram P (1997) An ancestral recombination graph. In: Donnelly P, Tavaré S, editors, Progress in Population Genetics and Human Evolution, Springer-Verlag, New York. pp. 257-270.

3. Robertson DL, Anderson JP, Bradac JA, Carr JK, Foley B, et al. (2000) HIV-1 nomenclature proposal. Science 288: 55-57.

4. Abecasis AB, Lemey P, Vidal N, de Oliveira T, Peeters M, et al. (2007) Recombination Confounds the Early Evolutionary History of Human Immunodeficiency Virus Type 1: Subtype G Is a Circulating Recombinant Form. J Virol 81: 8543-8551.

5. Lemey P, Lott M, Martin D, Moulton V (2009) Identifying recombinants in human and primate immunodeficiency virus sequence alignments using quartet scanning. BMC Bioinformatics 10: 126.

6. Anderson JP, Rodrigo AG, Learn GH, Madan A, Delahunty C, et al. (2000) Testing the Hypothesis of a Recombinant Origin of Human Immunodeficiency Virus Type 1 Subtype E. J Virol 74: 10752-10765.

7. Lai MM (1992) RNA recombination in animal and plant viruses. Microbiol Mol Biol Rev 56: 61-79.

8. Worobey M, Rambaut A, Holmes EC (1999) Widespread intra-serotype recombination in natural populations of dengue virus. Proceedings of the National Academy of Sciences of the United States of America 96: 7352-7357.

9. Hahn CS, Lustig S, Strauss EG, Strauss JH (1988) Western equine encephalitis virus is a recombinant virus. Proceedings of the National Academy of Sciences of the United States of America 85: 5997-6001.

10. Bergmann M, Garcia-Sastre A, Palese P (1992) Transfection-mediated recombination of influenza A virus. J Virol 66: 7576-7580.

11. Orlich M, Gottwald H, Rott R (1994) Nonhomologous recombination between the hemagglutinin gene and the nucleoprotein gene of an influenza virus. Virology 204: 462-465.

12. Sibold C, Meisel H, Kruger DH, Labuda M, Lysy J, et al. (1999) Recombination in Tula Hantavirus Evolution: Analysis of Genetic Lineages from Slovakia. J Virol 73: 667-675.

13. Jarvis TC, Kirkegaard K (1992) Poliovirus rna recombination: mechanistic studies in the absence of selection. Virology 11: 3135–3145.

14. Charrel RN, de Lamballerie X, Fulhorst CF (2001) The whitewater arroyo virus: Natural evidence for genetic recombination among tacaribe serocomplex viruses (family arenaviridae). Virology 283: 161 - 166.

15. Leitner T (2002) The molecular epidemiology of human viruses. Kluwer Academic publishers.

16. Shaikh R, Linial M, Coffin J, Eisenman R .

17. Feil EJ, Spratt BG (2001) Recombination and the population structures of bacterial pathogens. Annual Review of Microbiology 55: 561-590.

18. Goss EM, Kreitman M, Bergelson J (2005) Genetic Diversity, Recombination and Cryptic Clades in Pseudomonas viridiflava Infecting Natural Populations of Arabidopsis thaliana. Genetics 169: 21-35.

19. Foley B, Fauquet C (2008). We're not as confused as we may think we are: HIV nomenclature and classification in comparison to the nomenclature and classification of other viruses and bacteria. 15th International HIV Dynamics & Evolution meeting, Santa Fe.

20. de Oliveira T, Deforche K, Cassol S, Salminen M, Paraskevis D, et al. (2005) An automated genotyping system for analysis of HIV-1 and other microbial sequences. Bioinformatics 21: 3797-3800.

21. Maydt J, Lengauer T (2006) Recco: recombination analysis using cost optimization. Bioinformatics 22: 1064-1071.

22. Kuhner MK, Yamato J, Felsenstein J (2000) Maximum Likelihood Estimation of Recombination Rates From Population Data. Genetics 156: 1393-1401.

23. Lole KS, Bollinger RC, Paranjape RS, Gadkari D, Kulkarni SS, et al. (1999) Full-length human immunodeficiency virus type 1 genomes from subtype C-infected seroconverters in India, with evidence of intersubtype recombination. J Virology 73: 152-160.

24. Bailes E, Gao F, Bibollet-Ruche F, Courgnaud V, Peeters M, et al. (2003) Hybrid Origin of SIV in Chimpanzees. Science 300: 1713-.

25. Lukashev AN, Lashkevich VA, Ivanova OE, Koroleva GA, Hinkkanen AE, et al. (2005) Recombination in circulating Human enterovirus B: independent evolution of structural and non-structural genome regions. J Gen Virol 86: 3281-3290.

26. Schultz AK, Zhang M, Leitner T, Kuiken C, Korber B, et al. (2006) A jumping profile Hidden Markov Model and applications to recombination sites in HIV and HCV genomes. BMC Bioinformatics 7: 265.

27. Zhang M, Schultz AK, Calef C, Kuiken C, Leitner T, et al. (2006) jpHMM at GOBICS: a web server to detect genomic recombinations in HIV-1. Nucleic Acids Res 34: W463-465.

28. Spang R, Rehmsmeier M, Stoye J (2002) A Novel Approach to Remote Homology Detection: Jumping Alignments. J Comput Biol 9: 747-760.

29. Kingman JFC (1982) The coalescent. Stoch Proc Appl 13: 235 - 248.

30. Kingman JFC (1982) On the genealogy of large populations. J Appl Probab 19A: 27 - 43.

31. Wiuf C, Hein J (1999) Recombination as a point process along sequences. Theoretical Population Biology 55: 248 - 259.

32. Hudson RR (1990) Gene genealogies and the coalescent process. In: D Futuyama JA, editor, Oxford Surveys in Evolutionary Biology. Oxford, USA: Oxford University Press, volume 7, pp. 1-44.

33. Hein J, Schierup MH, Wiuf C (2005) Gene Genealogies, Variation and Evolution: A Primer in Coalescent Theory. Oxford University Press, USA.

34. Wakeley J (2008) Coalescent Theory: An Introduction. Roberts & Company Publishers, USA.

35. Rambaut A, Posada D, Crandall KA, Holmes EC (2004) The causes and consequences of HIV evolution. Nat Rev Genet 5: 52-61.

36. Kuhner MK (2006) LAMARC 2.0: maximum likelihood and Bayesian estimation of population parameters. Bioinformatics 22: 768-770.

37. Felsenstein J (1981) Evolutionary trees from DNA sequences: A maximum likelihood approach. J Mol Evol 17: 368-376.

38. Lanave C, Preparata G, Sacone C, Serio G (1984) A new method for calculating evolutionary substitution rates. J Mol Evol 20: 86-93.

39. Yang Z (1994) Maximum likelihood phylogenetic estimation from DNA sequences with variable rates over sites: Approximate methods. J Mol Evol 39: 306-314.

40. Zhuang J, Jetzt AE, Sun G, Yu H, Klarmann G, et al. (2002) Human Immunodeficiency Virus Type 1 Recombination: Rate, Fidelity, and Putative Hot Spots. J Virol 76: 11273-11282.

41. Altekar G, Dwarkadas S, Huelsenbeck JP, Ronquist F (2004) Parallel Metropolis coupled Markov chain Monte Carlo for Bayesian phylogenetic inference. Bioinformatics 20: 407-415.

Figure Legends

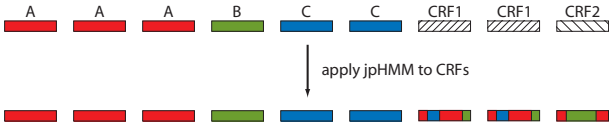

Figure 1. Example of a classification of 9 sequences into 3 subtypes (A, B, C) and 2 CRFs (CRF1, CRF2). At the bottom the recombinants have been segmented and the segments assigned a subtype by jpHMM.

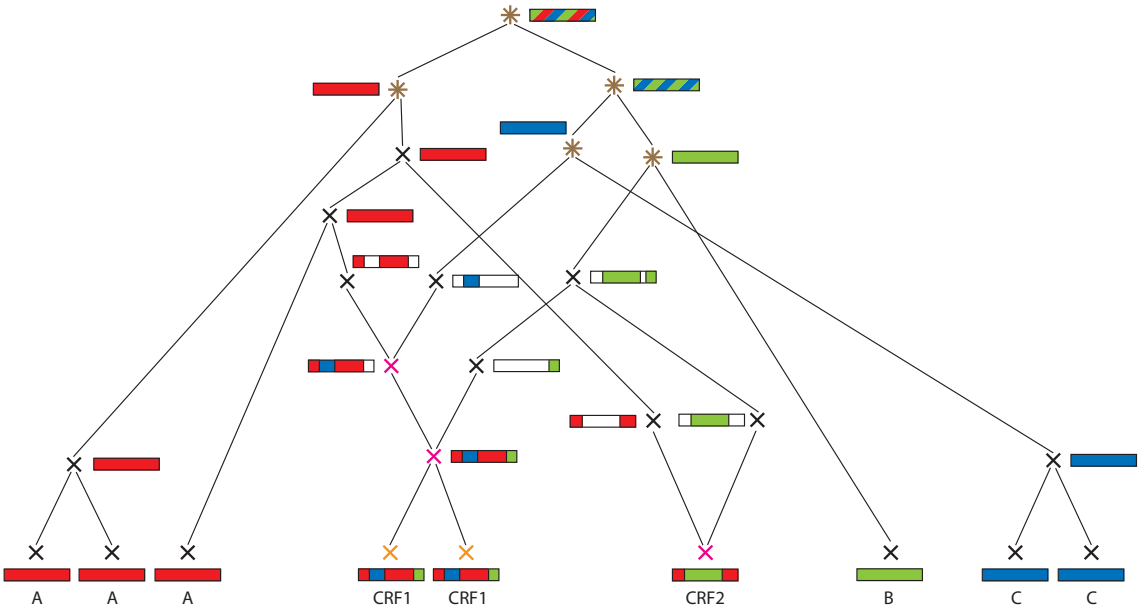

Figure 2. A legal ARG corresponding to the classification given in Figure 1. At the bottom, the nine input (tip) sequences with their classification are shown. The tip sequences are defined to be generated at time zero. Looking from bottom to top (i.e. into the past), two nodes coalescing to one (parental) node, represent the event of these two nodes finding their most recent common ancestor. A node splitting into two parental nodes represents a recombination event. Single-color boxes show the subtype of the node. Horizontally segmented boxes show for a recombinant sequence the parental subtypes of each segment. Diagonally shaded boxes show the different subtypes the node belongs to. White parts in boxes indicate positions not contributing to the tip sequences and, hence, of which we do not keep track. For recombination events, they also illustrate the positions of the recombination breakpoints. For further details, see Section "Methods – Overview – Restrictions to the genealogy".

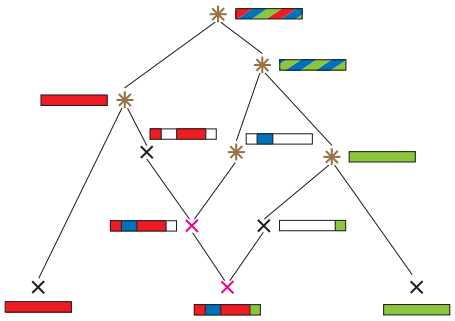

Figure 3. A legal ARG corresponding to a classification having an unknown subtype (blue). For details, see Section "Methods – Overview – Restrictions to the genealogy" and "Methods – Overview – Extension to unknown subtypes" and Figure 2.

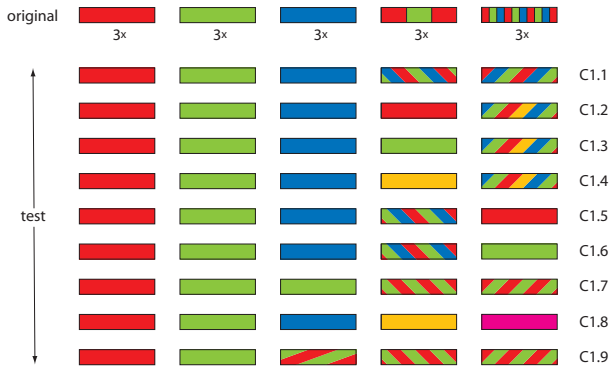

Figure 4. First test of test setting T2. On the top, the original classification is given. Single-color boxes symbolize triples of sequences belonging to a pure subtype (same colors indicate same subtype). The multicolor boxes symbolize sequences belonging to a CRF, showing its segmentation (which has to be provided in order to generate an ARG according to the classification and simulate the mutation process). In the lower part, the tested classifications are given. Single- and multicolor boxes symbolize the same as for the original classifications except that the segmentations of the CRFs are not given (the segmentations used by ARGUS are determined by jpHMM). Instead, the different diagonal patterns symbolize the different CRFs, the colors indicating the subtypes the CRF can be composed of (jpHMM always uses all subtypes available for determination of the segmentation of a CRF).

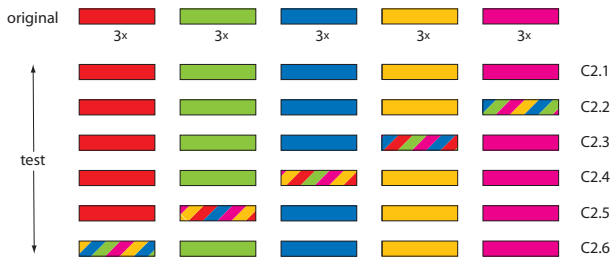

Figure 5. Second test of test setting T2. The same symbolism as in Figure 4 is used.

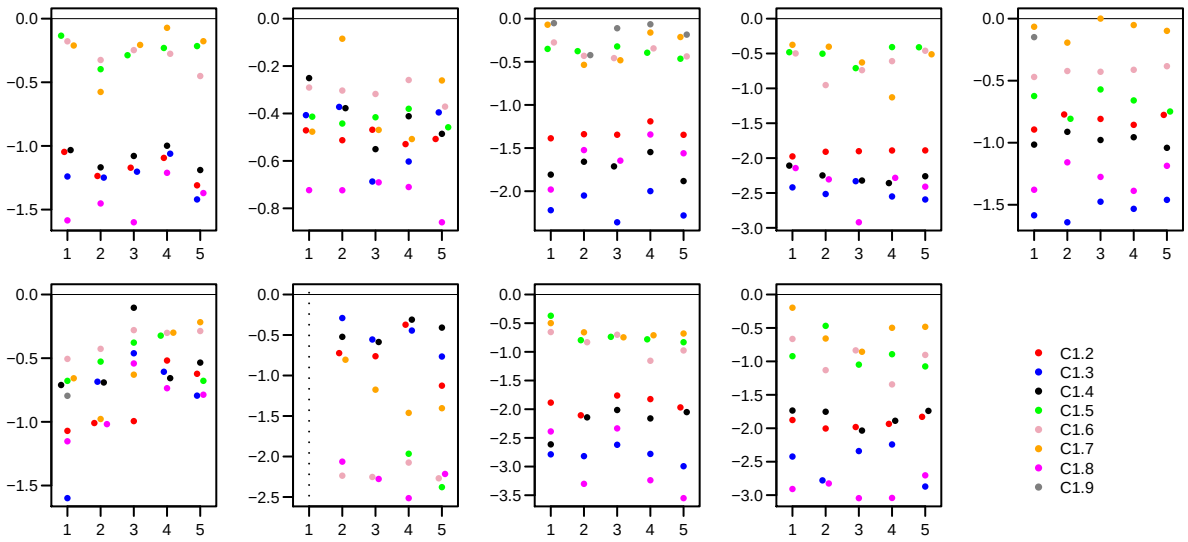

Figure 6. Results for the test setting described in Figure 4. On the vertical axis, $(\log P(D|G_T) - \log P(D|G_O)) \cdot 10^{-3}$ is given, with G_T the most likely ARG for the test (false) classifications C1.2-C.1.9 and G_O the most likely reconstructed ARG for the original classification C1.1. On the horizontal axis, the number of the set of simulated tip sequences is given. All points on a vertical represent tests conducted for the same tip sequences data (For clarity, points with very similar y-values were shifted slightly horizontally). In case a test classification contains one or more CRFs, but jpHMM was not able to detect all (i.e. at least one alleged CRF were diagnosed to belong to a pure subtype), the test results are omitted. In case that jpHMM designated at least one CRF of the original classification C1.1 to belong to a pure subtype, all test results for this tip sequences data set are omitted and a vertical dotted line is drawn instead. Depending on the stability of the results, 10-30 different initial ARGs were used for the MCMC algorithm, but always the same number for tests belonging to the same simulated ARG.

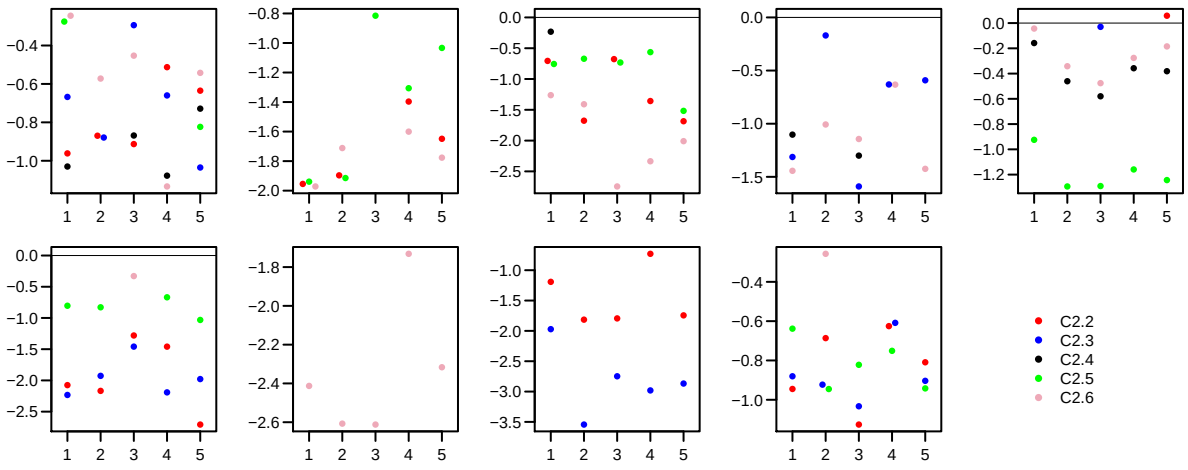

Figure 7. Results for the test setting described in Figure 5. We used 10-100 different initial ARGs for the MCMC algorithm. For details, see Figure 6.

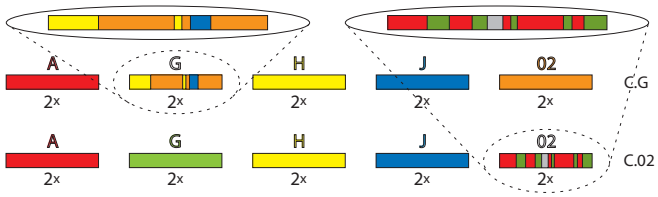

Figure 8. Classifications used in Section "Results – Empirical data" for deciding whether subtype G or CRF02 (=02) is a pure subtype or a recombinant form, resp. The gray segment in the lower segmentation of CRF02, indicates a part of the genome designated to stem from an unknown subtype. Above the classifications, the segmentation of the alleged CRFs is shown magnified.

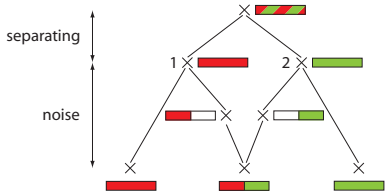

Figure 9. ARG illustrating the "separating distance" and the "noise distance". To be able to detect the recombination event, the nodes labeled 1 and 2 have to be sufficiently different. I.e. the 'separating distance' needs to be large enough. Contrariwise, the larger the 'noise distance', the less precisely the sequences of nodes 1 and 2 can be reconstructed from the tip sequences. I.e. the 'noise distance' should be small. The definition of the separating and noise distance is given in the supplements. For details about the symbolism used in the ARG, see Figure 2.

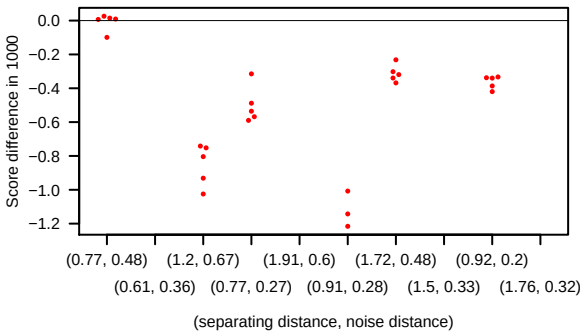

Figure 10. Results for the test setting preparing for the application with C.02 being the original classification. On the vertical axis, $\log(P(D|G_T) - \log P(D|G_O)) \cdot 10^{-3}$ is given, with G_T the most likely ARG for the test classification C.G and G_O the most likely reconstructed ARG for the original classification C.02. On the horizontal axis, the separating and noise distance is given for each of the 10 generated ARGs. The ARGs are sorted by their ratio of separating to noise distance (increasing from left to right). All points on a vertical represent tests conducted for the same ARG with different tip sequence data, with shifting as in Figure 6. In case jpHMM was not able to detect the CRF in C.G (i.e. the alleged CRF was diagnosed to belong to a pure subtype), the test results are omitted. For the MCMC algorithm, 20 different initial ARGs were used.

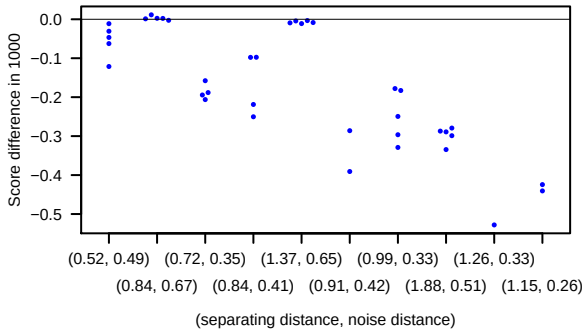

Figure 11. Results for the test setting preparing for the application with C.G being the original classification. On the vertical axis, $(\log P(D|G_T) - \log P(D|G_O)) \cdot 10^{-3}$ is given, with G_T and G_O, resp., the most likely (reconstructed) ARG for C.02 and C.G, resp. For more details (with the role of C.G and C.02 switched), see Figure 10.

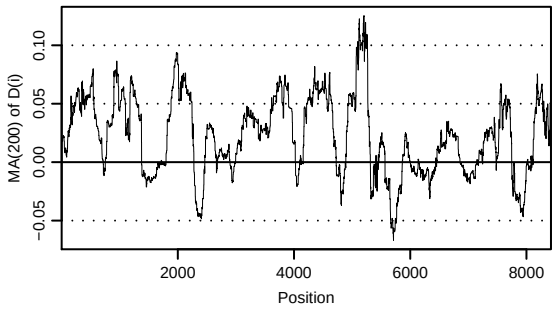

Figure 12. Moving average of $D(i) := \log P(D_i|G) - \log P(D_i|G')$ with an averaging period of 200 positions. Here G and G', resp., are the most likely reconstructed ARGs of the classifications C.02 and C.G, resp. (see Figure 8 for the classifications). At position i the average of position i until $i + 199$ is given.

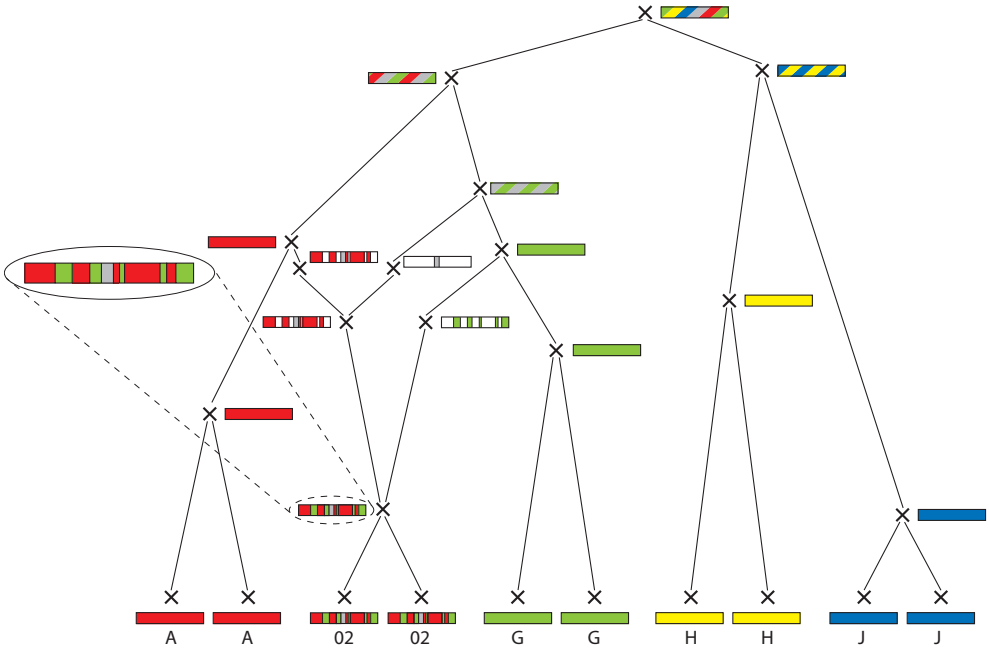

Figure 13. The most likely ARG found by the MCMC algorithm applied to C.02 (see Figure 8) using real HIV-1 Gr. M sequences. The vertical distance of the internal nodes to the tip nodes is drawn proportionally to their time of generation. The genome of one CRF02 sequence is shown magnified. For details about the symbolism used in the ARG, see Figure 2.