

Uncovering Genomic Reassortments Among Influenza Strains by Enumerating Maximal Bicliques

Niranjan Nagarajan
Ctr for Bioinf. and Comp. Biol.
Institute for Advanced Computer Studies
University of Maryland, College Park
niranjan@umiacs.umd.edu

Carl Kingsford
Dept. of Computer Science and
Ctr for Bioinf. and Comp. Biol.
Institute for Advanced Computer Studies
University of Maryland, College Park
carlk@cs.umd.edu

Abstract

The evolutionary histories of viral genomes have received significant recent attention due to their importance in understanding virulence and the corresponding ramifications to public health. We present a novel framework to detect reassortment events in influenza based on the comparison of two distributions of phylogenetic trees, rather than a pair of, possibly unreliable, consensus trees. We show how to detect all high-probability inconsistencies between two distributions of trees by enumerating maximal bicliques within a defined incompatibility graph. In the process, we give the first quadratic delay algorithm for enumerating maximal bicliques within general bipartite graphs. We demonstrate the utility of our approach by applying it to several sets of influenza genomes (both human- and avian-hosted) and successfully identify all known reassortment events and a few novel candidate reassortments. In addition, on simulated datasets, our approach correctly finds implanted reassortments and rarely detects reassortments where none were introduced.

1. Introduction

The influenza genome consists of eight disjoint RNA segments, each functioning as a small chromosome. When a single host is simultaneously infected with two strains of the influenza virus, the progeny virus particles may contain a mixture of segments from the two parent strains. It is believed that the pandemic strains of 1957 and 1968, for example, arose when such reassortment events incorporated novel immunogenic genes to which the human population had little immunity¹. Reassortments¹ are very common

¹Note the distinction from rearrangements; aside from being a fundamentally different biological process, rearrangements allow for a portion of a chromosome to be replaced and are not known to occur in influenza.

within strains hosted by the avian population. Recent work has shown that reassortment events among human-hosted strains also play a large role in the inter-pandemic evolution of the virus, with at least three reassortment events occurring between 1999 and 2004 [5].

Efforts to detect reassortment events among influenza strains using genomic data have been based on constructing a fixed phylogeny relating each segment of the strains under study. These trees are then compared to detect disagreements between the trees (Fig 1). This approach has several limitations. First, such an inspection provides no quantitative measure of confidence that a putative reassortment is real. Second, it provides no assurance that all reassortments that are present have been found. Third, because the sequences of thousands of influenza genomes are now available, with more being sequenced all the time, it has become prohibitively time-consuming to compare the large trees by hand. Finally, comparing fixed phylogenies does not take into account the large amount of uncertainty in estimating the evolutionary history of an influenza segment.

Ideally, we want to be able to detect reassortment events or the lack of them with confidence and despite any uncertainties in the phylogeny. This is the Achilles heel for the application of algorithms for comparing fixed gene trees (e.g. [15, 6, 14, 13, 3, 11]) to reassortment detection. While these tools provide automated procedures for detecting disagreements between phylogenies they do not take into account the possibility that putative disagreements may result from errors and uncertainties within the phylogenetic reconstruction. Phylogenies are particularly difficult to estimate for influenza, because although influenza evolves rapidly, we consider thousands of closely related strains at once, some of which have been sampled quite close in time. It is therefore nearly certain that, in practice, any single inferred tree contains errors. While taking a consensus of trees may eliminate false features, it also reduces the reso-

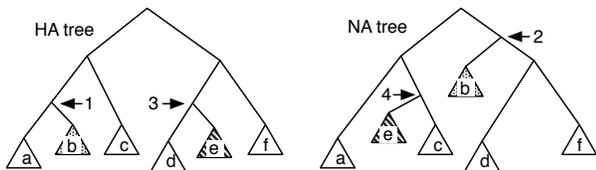


Figure 1. Schematic showing the effect of two reassortment events on a pair of phylogenetic trees.

Triangles represent subtrees. The taxa within the shaded subtrees are reassortants: strains within subtree labeled “b” inherited one segment from strains at time 1 and another segment from time 2. Similarly, strains in subtree “e” inherited segments from strains at times 3 and 4. Note that several reassortments may be present within a single data set, and we would like to find them all.

lution at which we can examine the tree and discards useful information. Thus, it is important in the case of influenza that methods for detecting disagreements between evolutionary histories are robust against errors in estimating those histories.

Another set of tools that are likely candidates for detecting reassortments are those devoted to detecting *recombination* between sequences (e.g. [12, 2, 16]). While, reassortment and recombination are biologically very different processes they result in similar exchanges of genetic material; reassortment can be viewed as a form of rearrangement, where the rearrangement breakpoints only occur at a small set of locations. In the general rearrangement detection problem, the multitude of possible breakpoint locations is the focus of most methods [2] and typically the candidate parental sequences and putative recombinants are part of the input [12, 16]. In the reassortment detection problem, in contrast, the reassorted taxa are the subject of interest and specifically here we wish to recover all discernible signals of reassortment events and the associated sets of resulting taxa. Further, methods such as the popular four-gamete rule that are typically used to designate haplotype blocks are not applicable to viral datasets as the infinite-sites assumption is typically violated.

In this work, we explore a novel approach specifically designed for the problem of detecting reassortments in the presence of uncertain phylogeny. Our approach is based on comparing weighted *sets* \mathcal{S}_1 and \mathcal{S}_2 of possible trees, rather than two fixed phylogenies. The weight of each tree is assumed to be related to the probability that it is the true tree. We then ask:

Is there a high-probability set of edges in the trees of \mathcal{S}_1 that are incompatible with a high-probability set of edges in the trees of \mathcal{S}_2 ?

Such a high-probability disagreement between the sets indicates that it is unlikely that the true trees represent the same evolutionary history (we make this statement more precise later), and hence that it is likely that a reassortment of the segments has occurred. Furthermore, if the above question is answered in the affirmative we would like to know sets of taxa that are likely to be the product of such events.

Weighted sets of trees can be readily obtained using either bootstrap sampling or using Bayesian MCMC approaches to phylogenetic tree reconstruction. Bayesian approaches are particularly attractive here as they provide an *ensemble* of possible trees for each gene or segment with an associated posterior probability based on the distribution from which they were sampled. They are also popular because of their speed and accuracy. This *distribution* of trees elegantly captures the uncertainty in phylogenetic reconstruction: if a single tree is heavily weighted, it is more likely to be the correct one. Interestingly, while tools like MrBayes [19] and BEAST [4] allow us to easily sample from a distribution of trees, in many contexts, subsequent analysis is often restricted to just using the consensus tree.

In this paper, we present a novel approach to uncovering reassortments by finding high-probability disagreements between distributions of trees, rather than just consensus trees. We do this by recasting the problem as one of finding maximal bicliques [1, 9, 10] within a bipartite graph such that the weights of the node sets (defined later) of the bicliques are sufficiently high. As part of this work, we show that the maximum biclique problem is NP-hard even when restricted to the incompatibility graphs that are of interest to us. We also show that there is no polynomial delay algorithm (i.e. one that uses polynomial time between successive instances in an enumeration) to enumerate all “high-weight” maximal biclique unless $P = NP$. This hardness result holds for general bipartite graphs, outside the context of comparing phylogenetic trees. Despite this theoretical difficulty, we give the first quadratic delay algorithm to enumerate all maximal bicliques (without the restriction to “high-weight” bicliques). In addition, the algorithm introduces a heuristic that speeds up the enumeration of high-weight maximal bicliques in practice. Maximal biclique enumeration has been used in several other contexts as well, and thus our methods are of independent interest outside of detecting reassortments. For example, Li et al. [9] discuss the application of maximal biclique enumeration to problems such as mining closed pattern sets, listing web communities and studying protein interaction networks. In Section 4.2 we discuss how the bicliques can be used to discover candidate sets of taxa resulting from a reassortment event. Finally, in Section 5 we present computational results on both real and simulated datasets demonstrating the usefulness of our approach for automatically detecting sets of reassortant taxa with high fidelity.

2. Problem Formulation

We restrict our attention to considering only two segments at a time that we call segment 1 and segment 2. Independent results from all pairs of the 8 influenza segments can be combined as a post-processing step to get a complete history of reassortments. In order to capture the uncertainty in the phylogenetic reconstruction, the input is assumed to be two collections $\mathcal{S}_1 = \{T_1^1, T_2^1, \dots, T_{s_1}^1\}$ and $\mathcal{S}_2 = \{T_1^2, T_2^2, \dots, T_{s_2}^2\}$ of trees where the trees of \mathcal{S}_1 represent the possible histories of one segment while those of \mathcal{S}_2 represent the possible histories of the other. The trees all relate the same set of taxa (leaves). Each tree is associated with a probability $p_i(T_j^i)$, indicating the chance that it is the correct tree for its segment. Bayesian MCMC approaches to phylogenetic reconstruction will directly produce such collections of trees and their probabilities. We wish to find *all* high-probability incongruencies between these two collections of phylogenies \mathcal{S}_1 and \mathcal{S}_2 and we do this by: 1) building a bipartite graph on nodes that represent the edges in the trees and 2) enumerating all maximal bicliques in the graph that satisfy certain conditions. We present the details in the following subsections.

2.1. Reduction to Maximal Biclique

To detect incongruencies between phylogenies we rely on the following well-known observation: every edge (u, v) in a tree divides the taxa into two disjoint sets — those within the clade rooted at node u and those outside of that clade. Such a bipartition is called a *split*, and we denote such a split by $X|Y$, where X and Y are nonempty, disjoint sets of taxa such that $X \cup Y$ is the complete set of taxa.

Definition 1 (Incompatible splits) *Two splits $X_1|Y_1$ and $X_2|Y_2$ are incompatible if all four intersections $X_1 \cap X_2$, $X_1 \cap Y_2$, $Y_1 \cap X_2$, and $Y_1 \cap Y_2$ are non-empty.*

It is easy to see that two incompatible splits cannot be in the same tree and conversely, every tree can be thought of as a collection of compatible splits. Let $\mathbf{splits}(T)$ be the set of splits that make up tree T . Two trees T and T' are *incompatible* if there is a split $X_1|Y_1 \in \mathbf{splits}(T)$ that is incompatible with some split $X_2|Y_2 \in \mathbf{splits}(T')$.

Generalizing this, given two ensembles $\mathcal{S}_1, \mathcal{S}_2$ of trees that we want to compare, we can construct a bipartite incompatibility graph where vertices on the “left” side correspond to splits seen in some tree of \mathcal{S}_1 and those on the “right” correspond to splits in some tree of \mathcal{S}_2 . Edges connect incompatible splits (Fig. 2). More formally:

Definition 2 (Incompatibility graph) *Given two sets \mathcal{S}_1 and \mathcal{S}_2 of trees, the incompatibility graph between \mathcal{S}_1 and \mathcal{S}_2 is the bipartite graph $\mathcal{B}(\mathcal{S}_1, \mathcal{S}_2) = (V_1 \cup V_2, E)$ such*

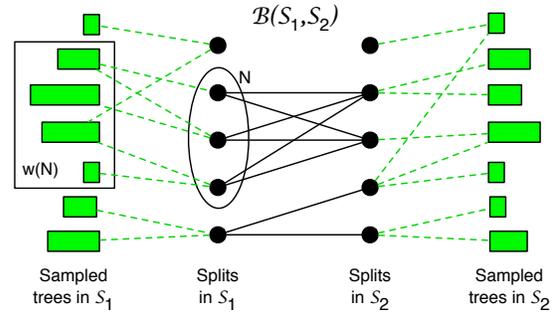


Figure 2. Incompatibility graph $\mathcal{B}(\mathcal{S}_1, \mathcal{S}_2)$ and the computation of weights for sets of nodes.

The circles and the edges between them make up the incompatibility graph. Circles represent splits that occur in some tree of \mathcal{S}_1 (left) or \mathcal{S}_2 (right). Solid edges connect incompatible splits. Shaded boxes represent sampled trees. The varying sizes of the boxes indicate varying probabilities associated with each tree. Dashed lines indicate which splits belong to which trees. The weight of a node set N on one side of $\mathcal{B}(\mathcal{S}_1, \mathcal{S}_2)$ is given by the sum of the probabilities of the trees that contain some split in N .

that, for $i \in \{1, 2\}$, $V_i = \bigcup_{T \in \mathcal{S}_i} \mathbf{splits}(T)$ and $E = \{(u, v) : \text{split } u \in V_1 \text{ is incompatible with split } v \in V_2\}$.

As part of the input we assume that each tree in the given ensembles ($T \in \mathcal{S}_i$ for $i \in \{1, 2\}$) is associated with a probability $p_i(T) = \Pr[T = \mathbf{true}_i]$, where \mathbf{true}_i is the “true” evolutionary tree. Such probabilities are readily computed from Bayesian MCMC estimators that are used to generate the ensembles of trees. Typically, because of uncertainty in phylogenetic reconstruction, $p_i(T)$ is low for every tree in each ensemble. However, we can use these probabilities to assign a probability (which we also refer to as *weight*), to any subset of nodes representing splits of a single ensemble (i.e. those contained on one side of the incompatibility graph).

Definition 3 (Weights for node sets) *The weight of a node set $N_i \subseteq V_i$ ($i \in \{1, 2\}$) is given by*

$$w(N_i) = \Pr[\mathbf{splits}(\mathbf{true}_i) \cap N_i \neq \emptyset] = \sum_{T \in \mathcal{S}_i : \mathbf{splits}(T) \cap N_i \neq \emptyset} p_i(T). \quad (1)$$

In other words, $w(N)$ is an estimate of the probability that the true tree contains some split from the set N (Fig. 2). Programs such as MrBayes will output $w(N)$ directly when N is a singleton set containing only one split as the estimate for the support for a given split. The definition above extends this to sets of splits of size > 1 .

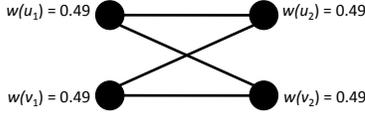


Figure 3. A simple example showing the necessity of considering bicliques. Any single edge suffices to show an incompatibility with probability < 0.5 . Assuming u_1 and v_1 are incompatible and u_2 and v_2 are incompatible, then this 4-node biclique indicates an incompatibility between the true trees with at least probability 0.98^2 . If even a single edge were missing above, then there would be a compatible choice of splits, and the trees would not necessarily be incompatible.

A *biclique* of a bipartite graph $B = (V_1 \cup V_2, E)$ is given by a pair of subsets of nodes (V'_1, V'_2) such that $V'_i \subseteq V_i$ and such that an edge exists in B for every pair $(u, v) \in V'_1 \times V'_2$. A biclique is *maximal* if no nodes can be added to it to make a larger biclique. In the context of the incompatibility graph between two sets of trees, maximal bicliques represent the largest sets of mutually incompatible choices for splits. If $w(V'_1)$ and $w(V'_2)$ are high, the true tree for segment $i \in \{1, 2\}$ is likely to contain a split u_i from the set V'_i . But because $V'_1 \cup V'_2$ is a biclique, any choices for u_1 and u_2 will be incompatible. Therefore, it is likely that the true trees are incompatible. Hence, we can find *all* likely incompatibilities between \mathcal{S}_1 and \mathcal{S}_2 by finding all t -maximal bicliques, defined below.

Definition 4 (t -maximal bicliques) A t -maximal biclique is a maximal biclique (V'_1, V'_2) such that $w(V'_1) > t$ and $w(V'_2) > t$ for $0 < t \leq 1$.

If a t -maximal biclique exists then the probability that the two true trees are incompatible is at least t^2 .

Figure 3 gives an example of a biclique of mutually incompatible splits that indicates that the trees are almost certainly incompatible (probability > 0.96), but for which no single pair of incompatible splits indicates a probability of more than 0.5 that the trees are incompatible. Therefore, in order to be sure to identify all incompatibilities, it is necessary to consider general bicliques rather than simply single edges within the incompatibility graph. Similarly, a majority consensus tree would omit both splits u_1 and v_1 of Fig. 3 and so this disagreement may not be apparent.

The following properties of bicliques are useful for searching for large bicliques: while a node-maximum biclique is not necessarily an edge-maximum one, it is true that a node-maximal biclique is always edge-maximal as well, and vice versa. In addition, because the weights $w(N)$ defined in (1) imply that $w(N) \leq w(N \cup \{u\})$ for any u , node-maximal bicliques are also maximal according to ei-

ther the sum or product of the weights of their node sets. Hence, by finding all node-maximal bicliques, we find the largest, highest probability sets of splits that are guaranteed to be incompatible. These sets of incompatible splits imply that the two segments likely had different evolutionary histories and can be post-processed to uncover which strains have likely been involved in reassortments (Sec. 4.2). Unfortunately, finding high-weight node-maximal bicliques is computationally difficult.

3. Computational Complexity

We first investigate the computational complexity of finding maximum and maximal bicliques, when restricted to our setting of incompatibility graphs generated from ensembles of trees. Finding the *maximum* biclique within a bipartite graph can be easy or hard depending on how cliques are scored. If we want to maximize the number of nodes in the biclique then there is a polynomial-time algorithm to do so [22]. However, if we want to maximize the number of edges in the biclique, the problem becomes NP-hard [17] and also hard to approximate within a factor of $|V|^\epsilon$, $\epsilon > 0$ unless $RP = NP$ [21].

In our setting, the problem of finding the biclique (V'_1, V'_2) that maximizes the natural scoring function $w(V'_1) \times w(V'_2)$ is a generalization of the maximum edge biclique problem. However, only certain kinds of graphs are possible – those that can be generated based on the splits in sets of trees. Unfortunately, the problem is still NP-hard.

Theorem 1 Finding the biclique (V'_1, V'_2) with the largest score $w(V'_1) \times w(V'_2)$ in the incompatibility graph for two sets of trees, \mathcal{S}_1 and \mathcal{S}_2 , is NP-hard.

Proof. We reduce from the maximum edge biclique problem. Let $G = (U \cup V, E)$ be the bipartite graph in which we wish to find the maximum edge biclique. We construct two set of splits A, B such that for $u \in U$, $v \in V$ and $(u, v) \in E$, $a(u) \in A, b(v) \in B$ and $a(u)$ and $b(v)$ are incompatible. Specifically, for each $u_i \in U$ we add four taxa w_u, x_u, y_u, z_u . Let A_u be the set $\{w_i, x_i, y_i, z_i : i \in U \text{ and } i \neq u\}$ that includes all the taxa except those associated with node u . For each $u \in U$, we create a split $a(u)$ defined by

$$a(u) = [A_u \cup \{w_u, x_u\}] \mid \{y_u, z_u\} \quad (2)$$

that splits the taxa associated with u and includes all the other taxa on one side. For every $v \in V$, define $\bar{E}_v = \bigcup_{(u,v) \notin E} \{w_u, x_u, y_u, z_u\}$, which is the set of taxa associated with a node $u \in U$ for which there is no edge

$\{u, v\} \in G$. We add a split $b(v)$ to B defined by

$$b(v) = \left[\bar{E}_v \cup \left(\bigcup_{(u,v) \in E} \{w_u, y_u\} \right) \right] \mid \left[\bigcup_{(u,v) \in E} \{x_u, z_u\} \right]. \quad (3)$$

Then $a(u)$ and $b(v)$ are incompatible if there is an edge $\{u, v\} \in G$. If there is no such edge, then $\{y_u, z_u\}$ is disjoint from $\bigcup_{(u,v) \in E} \{x_u, z_u\}$, and therefore the right set of $a(u)$ is disjoint from the right set of $b(v)$. By Definition 1, $a(u)$ and $b(u)$ are compatible, and therefore there is an edge in the incompatibility graph iff there is an edge in G . Each split yields a tree to which we assign equal probability. The reduction is then completed by noting that the score of a biclique is proportional to the number of its edges. \square

While an efficient algorithm to find the maximum biclique would be useful (say, to quickly test for reassortments in the dataset), we really want to enumerate *all* t -maximal bicliques in order to find all possible reassortments within a large set of taxa. Fortunately, enumerating all maximal bicliques is possible in time polynomial in the size of the output (which may be exponential in the size of the input). In fact, there are polynomial delay algorithms that will always output a new maximal biclique in polynomial (in size of the input) time [1, 9]. Unfortunately, no such algorithms exist for enumerating the t -maximal bicliques that we are most interested in:

Theorem 2 *There exists no polynomial delay algorithm for enumerating t -maximal bicliques in an incompatibility graph unless $P = NP$.*

Proof. We prove the theorem by showing that the existence of such an algorithm will give a polynomial time algorithm for the balanced biclique problem (largest node biclique (V'_1, V'_2) where $|V'_1| = |V'_2|$) which is known to be NP-hard [8]. For a given instance of the balanced biclique problem with bipartite graph $G = (U \cup V, E)$, we construct the sets of trees as in the proof for Theorem 1. A t -maximal biclique in the corresponding incompatibility graph can be converted into a balanced biclique of size t by removing nodes, and therefore we can search over $t \in [1.. \min(|V_1|, |V_2|)]$ to find the largest balanced biclique in polynomial time, giving a contradiction unless $P = NP$. This reasoning applies to any bipartite graph, showing that the problem is hard in general bipartite graphs. \square

4. Algorithm for Detecting Reassortments

Notwithstanding the negative computational complexity results of the previous section, in practice, all t -maximal bicliques can be enumerated within a reasonable amount of time. We describe our approach below, which is a modification of the consensus approach for maximal biclique

enumeration described in [1]. If no t -maximal bicliques are detected, this is strong evidence that there has been no reassortment event within the taxa. If some t -maximal bicliques are found, they must be post-processed in order to reveal the set of taxa that are the result of a reassortment event.

4.1. Enumerating t -Maximal Bicliques

In order to detect t -maximal bicliques, we extend the MICA algorithm for enumerating maximal bicliques proposed by Alexe et al. [1]. Let P be the set of all maximal star subgraphs that have a single vertex in V_1 . We maintain a data structure C that stores the maximal bicliques discovered through the current step. The cliques in both P and C are represented by only their right subsets (which uniquely determine their left subsets in a maximal biclique). We therefore denote a biclique by (\diamond, V) , where the diamond indicates the completely determined — but un-computed — subset of V_1 induced by V . The algorithm for enumerating t -maximal bicliques is then as follows:

Algorithm 1. List- t -Maximal-Bicliques:

1. Prune P such that for $(\diamond, V) \in P$, $w(V) \geq t$.
 2. Initially, C contains P .
 3. For each pair of bicliques $B' = (\diamond, V') \in P$ and a biclique $B'' = (\diamond, V'') \in C$:
 - (a) Compute a consensus biclique $B''' = (\diamond, V' \cap V'' = V''')$, which takes the intersection of the sets on the right side of the bicliques.
 - (b) If B''' is a non-empty biclique that doesn't occur in C and $w(V''') \geq t$, add B''' to C .
 4. If new bicliques were added in step 3, repeat step 3 for the new bicliques.
 5. For each $M = (\diamond, V) \in C$, compute $U = \bigcap_{n \in V} \mathbf{neighbors}(n)$. If $w(U) > t$ output (U, V) as a t -maximal biclique.
-

This algorithm extends the MICA approach [1] in two important ways. First, it allows for enumeration of t -maximal bicliques in a more efficient way and the algorithm has a runtime that is polynomial in the number of bicliques where the right side satisfies the t threshold but the left side may not. The check for such bicliques is done in lines 1 and 3(b) of the algorithm. Bicliques that fail these checks can never be merged with any other biclique to form a t -maximal biclique because the weight function is monotonic i.e. $w(V) < w(V \cup \{v\})$ and the formation of a consensus only reduces the size of the right node set. Hence, these checks do not eliminate any t -maximal bicliques.

The second refinement that improves the runtime further is the use of a more efficient data structure for storing the set of discovered cliques C . As proposed in [1], these N sets can be stored in a sorted list and binary searched in $O(n \log N)$ time (taking n time to do each comparison). Because $N = O(2^n)$, searching for a set with this naive data structure takes $O(n^2)$ time. Instead, we use a binary trie, where each level corresponds to the presence or absence of a particular node within the set. There are at most 2^n such sets, and the tree contains at most n levels. Testing for the presence of a given set, therefore, takes $O(n)$ time. This results in a delay of $O(n^2)$ instead of the $O(n^3)$ delay in the MICA algorithm, because we save a factor of $O(\log N) = O(n)$ time. Further, instead of $O(nN)$ space, we use $O(N)$ space. Thus, we have shown:

Theorem 3 *All maximal bicliques of a bipartite graph can be enumerated with quadratic delay using $O(N)$ space, where N is the total number of maximal bicliques.*

Proof. The proof of correctness for the algorithm is similar to that in [1]. Due to space constraints we omit it. \square

To our knowledge, this is the first quadratic delay algorithm for this problem; the algorithm proposed in [10] requires $O(n^3)$ preprocessing time while the algorithm in [9] is a cubic delay algorithm.

4.2. Recovering the Reassorted Taxa

The existence of large bicliques in the incompatibility graph for two segments provides significant evidence for reassortment events in their history. To probe these events further, we need to identify the taxa that are likely to be reassortants. We can recover some of this information from the incompatibility graph. Let $X|Y$ and $A|B$ be a pair of incompatible splits. From the definition of incompatibility it follows that we can write $X|Y = X'X''|Y'Y''$ and $A|B = X'Y'|X''Y''$ such that $X' \cup X'' = X$, $Y' \cup Y'' = Y$. Thus, any pair of incompatible splits define four sets of taxa, and each of these sets are candidates for the taxa that have resulted from a reassortment.

Definition 5 (Reassortment candidates) *Given incompatible splits $X'X''|Y'Y''$ and $X'Y'|X''Y''$ the four reassortment candidates are X' , X'' , Y' , Y'' .*

Labelling each edge in the incompatibility graph with these sets we can now ask the question “Are there t -maximal bicliques where the edges all share at least one label?” In particular, bicliques that are non-stars (both sides have more than one node) provide an unambiguous reassortment hypothesis as they can have only one common label:

Theorem 4 *Edges of a non-star biclique can share at most one label. Star bicliques with more than one edge can share two or four labels.*

Proof. We first prove the star biclique case. Let $X = X_1|X_2$, $Y = Y_1|Y_2$, $Z = Z_1|Z_2$ be three splits such that (X, Y) and (X, Z) represent two edges of a star biclique that share a label. Then, without loss of generality, let $X_1 \cap Y_1$ be the shared label such that $X_1 \cap Y_1 = X_1 \cap Z_1$. Because $Y_1 \cup Y_2 = Z_1 \cup Z_2$, this implies that $X_1 \cap Y_2 = X_1 \cap Z_2$ (i.e. the edges share two labels). Clearly if X, Y, Z share three labels then they will share all four labels. Consider now the non-star biclique case. Let $W = W_1|W_2$ be another split ($X \neq W$) such that (W, Y) , (W, Z) are edges and X, Y, Z, W represent a non-star biclique with more than one shared label. Without loss of generality, let these labels be $A = X_1 \cap Y_1 = X_1 \cap Z_1$ and $B = X_1 \cap Y_2 = X_1 \cap Z_2$ and let $W_1 \cap Y_1 = A$. If $W_1 \cap Y_2 = B$, then this implies that $X_1 = W_1$ and that $X = W$. So it must be that $W_2 \cap Y_2 = B = W_2 \cap Z_2$ and that $W_2 \cap Y_1 = W_2 \cap Z_1$ because $Y_1 \cup Y_2 = Z_1 \cup Z_2$. So, from $W_1 \cap Y_1 = A = W_1 \cap Z_1$ we get that $Y_1 = Z_1$ and $Y = Z$ which contradicts the assumption that $Y \neq Z$. \square

We exploit Thm. 4 to break the problem into an independent subproblem for each possible edge label. For each edge label A , we create a subgraph of the incompatibility graph that contains only edges that are labeled with A . We search for t -maximal bicliques in each of these subgraphs separately and aggregate the discovered t -maximal bicliques into a single list. The set of non-star bicliques can then be used to identify possible sets of taxa with sequences resulting from a reassortment.

5. Computational Results

We present results on human influenza, avian influenza, and two artificial data sets (Table 1). For each experiment, MrBayes [19] was used to construct an ensemble of 1001 candidate trees (GTR model with gamma distributed rate variation among sites) and their probabilities by sampling after every 200 iterations (following an MCMC burn-in period of 100,000 iterations). We then used an implementation of the algorithm described above to enumerate all the t -maximal bicliques, setting the confidence threshold $t = 0.9$. Running times to enumerate t -maximal bicliques were a few seconds for each of the data sets (Table 2), a substantial improvement over the tens of minutes needed using the MICA algorithm [1].

5.1. Reassortments in Human Influenza

We considered a set of 259 genomes from human-hosted H3N2 influenza isolates collected in New York State between 1998 and 2005 and sequenced by the Influenza Genome Sequencing Project [7]. These 259 genomes represent a superset of the genomes analyzed in Holmes et al. [5],

Data set	I	n_1	n_2	m
Human NYS	259	832	786	14360
Avian H5N1	35	69	87	831
Mock No Re.	259	365	531	4265
Mock 1 Re.	259	365	481	5338

Table 1. Problem sizes on four influenza data sets. Column I gives the number of influenza isolates considered. Columns n_1 , n_2 , and m give the total number of nodes (on each side) and edges in the incompatibility graph. Sizes for only one mock replicate are shown.

Data set	Candidates	Confirmed	Time
Human NYS	19	6	9 sec
Avian H5N1	2	1	2 sec
Mock No Re.	0	0	4 sec
Mock 1 Re.	7	1	5 sec

Table 2. Results. The “Candidates” column lists the number of sets of taxa supported by t -maximal, non-star bicliques ($t = 0.9$). “Confirmed” gives the number of the candidates that are known or are likely to be true reassortments. The last column lists the time taken to enumerate the t -maximal bicliques.

in which several reassortment events were uncovered by visual inspection of the trees. All 8 segments are available for each of these isolates, but we focussed on the antigenically important HA and NA segments. Our analysis revealed 19 sets of taxa that uniquely label t -maximal bicliques. Of these sets, three sets exactly match the previously known reassortment events. While the other sets seem to be associated with these known events, three more single-isolate sets reveal interesting hypotheses that deserve further investigation ($\{A/New\ York/105/2002\}$, $\{A/New\ York/177/1999\}$, $\{A/New\ York/289/1998\}$). We also performed a similar analysis comparing the NA and PA segments. While no reassortment events have previously been reported between these segments, our analysis suggested several likely reassortment sets (in particular $\{A/New\ York/135/2002\}$ and $\{A/New\ York/96/2002, A/New\ York/128/2002\}$), further underscoring the utility of automated analysis.

5.2. Reassortments in Avian Influenza

Reassortments are even more common among avian-hosted influenza. We considered 35 of the avian high-pathogenic H5N1 avian influenza isolates that were analyzed in Salzberg et al. [20]. These isolates were collected in 2005 and 2006 from Europe, the Middle East, northern Africa, and Vietnam. Previous manual analysis strongly

suggested that one isolate, A/Nigeria/1047-62/2006, was a reassortment, having derived 4 of its segments from one strain and 4 from another strain. We again applied the Mr-Bayes tree sampling to generate ensembles of trees for the HA and NA segments.

The incompatibility graph contains 831 edges labeled with 586 candidate taxa sets (as in Definition 5). The known reassortment is detected: only 2 candidate taxa sets are supported by a non-star, t -maximal biclique. One of these candidate taxa sets, supported by 17 t -maximal bicliques, is a singleton set that contains exactly the previously found reassortant A/Nigeria/1047-62/2006. The other set, supported by a single t -maximal biclique, consists of 8 Nigerian and Niger isolates closely related to the strain that donated its NA segment to A/Nigeria/1047-62/2006.

5.3. Checking on Artificial Datasets

The results above suggest that our methodology can detect known and suspected reassortments in real data (both avian and human) by producing a small candidate set. To double check the effectiveness of the method in a setting where the truth is completely known, we considered two artificially generated data sets, one in which a single reassortment was implanted and one in which no reassortments were modeled.

To generate a set of sequences with no reassortments expected, we built a neighbor-joining tree relating the HA segments of the isolates considered in Holmes et al. [5]. We then discarded the sequences and evolved new sequences along this tree using Seq-Gen [18] using the F84 evolutionary model. As parameters to the evolutionary model, we used sequence length, background base frequencies, and transition/transversion (Ti/Tr) ratio estimated from the real HA sequences. This resulted in a randomized “HA-like” collection of sequences. We then repeated this process, using the HA-tree, but choosing the length, background base frequencies, and Ti/Tr ratio estimated from the real NA sequences. This resulted in an “NA-like” collection of sequences that evolved with the same relationships as the HA-like sequences. Hence, we hope to find no reassortments.

Running the detection algorithm resulted in no t -maximal bicliques found, which is the desired result. We double checked this by repeating the experiment 9 times with new random instances and got similar results. These tests underscore the utility of the t -maximal biclique approach for confidently ruling out reassortments in datasets.

To create a mock-reassortment data set, we generated a random “HA-like” collection of sequences as described above following the neighbor-joining tree built on the Holmes et al. [5] isolates. We then chose an edge at random and moved the subtree leading from that edge to a different random place within the tree. In our dataset the randomly

chosen clade happened to define a set of 8 taxa. On this mock-reassortment tree, we then evolved random sequences using model parameters estimated from real NA sequences. This resulted in a set of “NA-like” sequences that should exhibit a single reassortment relative to HA.

Running the detection algorithm described above resulted in 7 candidate taxa sets supported by non-star, t -maximal bicliques. One of these 7 sets corresponds exactly to the implanted 8-taxa reassortment. We repeated this experiment for 6 additional reassortment sets ranging in size from 14 to 22 taxa and obtained similar results. Note that while the candidate taxa present possible explanations for the events represented by the t -maximal bicliques, they are not necessarily the only explanations. We are currently working on a statistical test (based on the changes in distance of putative reassortants from non-reassorted taxa in the two segments) that could help prune the set of hypotheses further.

6. Discussion

The t -maximal biclique approach proposed here presents a rigorous and exact way to find *all* well-supported disagreements between phylogenetic histories. It is thus a principled method for handling uncertainty in phylogenetic reconstruction and exploits the full power of Bayesian MCMC sampling approaches, rather than just analyzing a consensus tree.

As demonstrated by our tests on real and artificial datasets, the resulting bicliques can be successfully mined to identify known reassortment events. In fact, in the two collections of avian and human influenza isolates considered, the t -maximal biclique approach identifies all the known reassortments. This provides quantitative support that these reassortments are in fact real.

The problem of enumerating maximal bicliques is of independent interest in other areas of bioinformatics and computer science and here we present an efficient quadratic-delay algorithm based on the consensus approach. It remains an open question whether a quadratic delay algorithm can be devised that only uses polynomial space.

References

- [1] G. Alexe, S. Alexe, and Y. C. et al. Consensus algorithms for the generation of all maximal bicliques. *Disc. App. Math.*, 145:11–21, 2004.
- [2] C. X. Chan, R. Beiko, and M. Ragan. Detecting recombination in evolving nucleotide sequences. *BMC Bioinf.*, 7(412), 2006.
- [3] A. W. M. Dress and D. H. Huson. Constructing splits graphs. *IEEE/ACM Trans. in Comput. Biol. and Bioinf.*, 1:109–115, 2004.
- [4] A. Drummond and A. Rambaut. BEAST: Bayesian evolutionary analysis by sampling trees. *BMC Evol. Biol.*, 7(214), 2007.
- [5] E. Holmes, E. Ghedin, and N. M. et al. Whole-genome analysis of human influenza A virus reveals multiple persistent lineages and reassortment among recent H3N2 viruses. *PLoS Biol.*, 3:1579–1589, 2005.
- [6] D. Huson, T. T. Klöpper, and P. L. et al. Reconstruction of reticulate networks from gene trees. *LNCS*, 3500:233–249, 2005.
- [7] Influenza genome sequencing project. <http://www3.niaid.nih.gov/research/resources/mscs/Influenza/>.
- [8] D. Johnson. The NP-completeness column: An ongoing guide. *J. of Algorithms*, 8:438–448, 1987.
- [9] J. Li, G. G. Liu, and H. L. et al. Maximal biclique subgraphs and closed pattern pairs of the adjacency matrix: A one-to-one correspondence and mining algorithms. *IEEE Trans. on Knowledge and Data Engineering*, 19:1625–1637, 2007.
- [10] K. Makino and T. Uno. New algorithms for enumerating all maximal cliques. In *Proc. of SWAT*, pages 260–272, 2004.
- [11] K. McBreen and P. Lockhart. Reconstructing reticulate evolutionary histories of plants. *TRENDS in Plant Science*, 11:398–404, 2006.
- [12] V. Minin, K. K.S. Dorman, and F. F. et al. Phylogenetic mapping of recombination hotspots in human immunodeficiency virus via spatially smoothed change-point processes. *Genetics*, 175:1773–1785, 2007.
- [13] T. Munzner, F. Guimbretière, and S. T. et al. TreeJuxtaposer: Scalable tree comparison using focus+context with guaranteed visibility. *ACM Trans. Graph.*, 22:453–462, 2003.
- [14] L. Nakhleh, T. Warnow, and C. Linder. Reconstructing reticulate evolution in species: theory and practice. In *Proc. of RECOMB*, pages 337–346, 2004.
- [15] R. Page. GeneTree: comparing gene and species phylogenies using reconciled trees. *Bioinformatics*, 14:819–820, 1998.
- [16] D. Paraskevis, K. Deforche, and P. L. et al. SlidingBayes: exploring recombination using a sliding window approach based on bayesian phylogenetic inference. *Bioinformatics*, 21:1274–1275, 2005.
- [17] R. Peeters. The maximum edge biclique problem is NP-complete. *Disc. App. Math.*, 131:651–654, 2003.
- [18] A. Rambaut and N. Grassly. Seq-Gen: An application for the Monte Carlo simulation of DNA sequence evolution along phylogenetic trees. *Comput. Appl. Biosci.*, 13:235–238, 1997.
- [19] F. Ronquist and J. Huelsenbeck. MRBAYES 3: Bayesian phylogenetic inference under mixed models. *Bioinformatics*, 19:1572–1574, 2003.
- [20] S. Salzberg, C. Kingsford, and G. C. et al. Genome analysis linking recent european and african influenza (H5N1) viruses. *Emerg. Infect. Dis.*, 13:713–718, 2007.
- [21] J. Tan. Inapproximability of maximum weighted edge biclique and its applications. *LNCS*, 4978:282–293, 2008.
- [22] M. Yannakakis. Node deletion problems on bipartite graphs. *SIAM J. Comput.*, 10:310–327, 1981.