Reconstructing (Super)Trees from Data Sets with Missing Distances: Not All Is Lost

George Kettleborough,[1] Jo Dicks,[2] Ian N. Roberts,[2] and Katharina T. Huber*[,3]

[1]The Genome Analysis Centre (TGAC), Norwich Research Park, Norwich, United Kingdom, and School of Computing Sciences, University of East Anglia, United Kingdom

[2]National Collection of Yeast Cultures (NCYC), Institute of Food Research (IFR), Norwich Research Park, Norwich, United Kingdom

[3]School of Computing Sciences, University of East Anglia, United Kingdom

***Corresponding author:** E-mail: katharina.huber@cmp.uea.ac.uk.

Associate editor: Naoki Takebayashi

Abstract

The wealth of phylogenetic information accumulated over many decades of biological research, coupled with recent technological advances in molecular sequence generation, presents significant opportunities for researchers to investigate relationships across and within the kingdoms of life. However, to make best use of this data wealth, several problems must first be overcome. One key problem is finding effective strategies to deal with missing data. Here, we introduce Lasso, a novel heuristic approach for reconstructing rooted phylogenetic trees from distance matrices with missing values, for data sets where a molecular clock may be assumed. Contrary to other phylogenetic methods on partial data sets, Lasso possesses desirable properties such as its reconstructed trees being both unique and edge-weighted. These properties are achieved by Lasso restricting its leaf set to a large subset of all possible taxa, which in many practical situations is the entire taxa set. Furthermore, the Lasso approach is distance-based, rendering it very fast to run and suitable for data sets of all sizes, including large data sets such as those generated by modern Next Generation Sequencing technologies. To better understand the performance of Lasso, we assessed it by means of artificial and real biological data sets, showing its effectiveness in the presence of missing data. Furthermore, by formulating the supermatrix problem as a particular case of the missing data problem, we assessed Lasso's ability to reconstruct supertrees. We demonstrate that, although not specifically designed for such a purpose, Lasso performs better than or comparably with five leading supertree algorithms on a challenging biological data set. Finally, we make freely available a software implementation of Lasso so that researchers may, for the first time, perform both rooted tree and supertree reconstruction with branch lengths on their own partial data sets.

Key words: **phylogenetic trees, rooted trees, partial distance, supertree, lasso, molecular clock, dendrogram.**

Article

Introduction

The ease and speed with which molecular sequence data can now be generated using Next Generation Sequencing (NGS) technologies are enabling evolutionary biologists to embark on exciting, albeit highly challenging, endeavors such as determining the phylogenetic relationships within and between the kingdoms of life, and at a resolution rarely seen previously. NGS has perhaps been most influential at the subspecies level, and data sets encompassing numerous lines, strains or accessions are becoming commonplace. These new data, together with a wealth of legacy data sets typically at a higher taxonomic level, promise the interleaving of species and subspecies within a common evolutionary framework.

Despite major advances in the field of phylogenetics over the last half-century of phylogenetic studies, and in particular large-scale analyses involving hundreds of taxa, we still face many obstacles. Current problems range from data collection and data storage to information extraction and tree or network building. Even with greater access to high performance computing resources, computationally demanding phylogenetic approaches such as Bayesian, likelihood, and parsimony methods may be out of reach for researchers possessing such data sets, given the vastness of tree space. Consequently, distance-based methods have an important role to play, rapidly providing phylogenetic trees or networks that can form a basis for further investigation.

Distance-based methods have been the subject of considerable criticism, centering on their representation of (potentially extensive) character or molecular variation as a single number. Nonetheless, the computational efficiency and ease of use of distance-based tree reconstruction methods make them an attractive option for large data sets, where they can provide a snapshot of the underlying evolutionary relationships quickly and easily. In addition to providing evolutionary insights in their own right, they can be used to provide good starting/guide trees for more sophisticated methods such as the ones above. Furthermore, new genetic distance estimation methods continue to be developed with improved properties and for particular data sets, such as those recently introduced in Joly et al. (2015). Importantly, the new distance measures can be used to exploit potentially massive single nucleotide polymorphism (SNP) data sets such as those derived from NGS reads. They lead to a greater accuracy of distance estimates in the face of key issues such as allelic

variation, polyploidy, and recombination over existing measures and within a simulation framework that allows for further development in this area.

Another key problem in phylogenetics, as indeed in all data analyses, is how to deal with missing data. In the biological arena, missing data tend to arise either from neglecting to collect a sample of interest or from the failure of an experimental assay. Given the growing uptake of high-throughput technologies, it is likely that the latter problem will be most prevalent going forward. Popular distance-based methods such as Neighbor Joining (Saitou and Nei 1987), and BioNJ (Gascuel 1997) (for the inference of unrooted trees) and UPGMA (Sokal 1958) (for rooted trees) require that an input distance matrix does not contain any missing values (i.e., the distance matrix D on a taxa set X is *complete*).

In practice, such a requirement often results in researchers removing certain taxa from an analysis, where including them would give rise to one or more missing values. In addition to losing data, a recent study by Huang and Knowles (2015) showed that excluding such data points from NGS analyses may have further unexpected consequences such as biasing the included loci with regard to their mutation rates. However, simulation studies (see, e.g., Criscuolo et al. 2006) indicate that distance data sets can contain a considerable amount of redundancy, suggesting that not all distance values are required to reconstruct the underlying tree. Consequently, methods capable of reconstructing phylogenetic trees from incomplete, or partial, distance matrices may provide researchers with the ability to analyze the entirety of their data and to minimize or negate consequences such as those identified in Huang and Knowles (2015).

Little research to date has investigated how to make use of this redundancy and, out of what has been done, most has focused on how to exploit it to reconstruct a phylogenetic tree. However, another fundamental issue of interest is that of "uniqueness." Here, this concept refers to the following question: Can we find the set of pairwise distances which are sufficient to identify the tree of relationship on which they have evolved? Formalized in terms of when a set of pairs of taxa "lassos" a phylogenetic tree (Dress et al. 2011), it turns out that this question is surprisingly difficult to answer when the tree in question is unrooted (see, e.g., Dress et al. 2011; Huber and Steel 2014, for recent partial theoretical results), and not much is known about the rooted tree case.

Fortunately, the uniqueness problem becomes much more tractable when, in addition to being rooted, the sought after tree is also "equidistant" (meaning that the distance from the root of the tree to each of its leaves is the same) (see, e.g., Huber and Popescu 2013; Huber KT and Kettleborough G, submitted, for some mathematical characterizations of this problem). Sometimes referred to as equidistant representations (Semple and Steel 2003), dendrograms, or ultrametric trees such trees are commonly constructed when a molecular clock (Zuckerkandl and Pauling 1962) can be assumed for the evolution of the taxa of interest.

The recently emerged plethora of NGS data sets, particularly those at the subspecies level, have been shown to include many consistent with a molecular clock. These data, together with widely used software packages such as BEAST (Bouckaert et al. 2014), which enable the construction of phylogenetic trees for both molecular and relaxed molecular clocks, highlight the relevance of this concept to contemporary phylogenetic studies. Examples of recent analyses where molecular clocks have enabled a deep level of understanding to be gained include population studies (e.g., determination of plant germplasm genetic diversity; Xiao et al. 2010), paleontological studies (e.g., estimation of divergence dates and the effect of climate change on diversification; Weir and Schluter 2008b), and phylogeographic studies (Confalonieri et al. 1998). Weir and Schluter (2008a) provide more details on these examples and see Hellmuth et al. (2013) on the successful use of so-called symbolic ultrametric trees in orthology detection.

Additional challenges in phylogenetic analysis stem from attempting to combine disparate data sets in a single analysis. In addition to dealing with patchy taxonomic coverage, as some taxa will have been studied more than others (e.g., Philippe et al. 2004; Sanderson et al. 2010; Steel and Sanderson 2010; Roure et al. 2012), constructing such a tree entails finding ways to combine different types, qualities, and quantities of data, as well as addressing the problem of how to combine data sets that might only share very few taxa. The latter is a formidable problem in its own right, and in practice there have been two main approaches to solve it. Supertree approaches aim to combine two or more (potentially very small) phylogenetic trees within a single parental tree on all the taxa, that in some sense displays the evolutionary information contained within the starting trees. Supermatrix approaches aim to combine data sets underlying the tree, either at the character state matrix or at the distance matrix level. Supermatrix approaches are, of course, a particular form of the missing value problem described above, as combining two complete matrices with overlapping taxa will result in one partial matrix with potentially many missing values. These approaches generally use some sort of algorithm (known as an *imputing* scheme) to replace missing values with likely values (see, e.g., Bininda-Emonds 2004; Queiroz and Gatesy 2006 for such schemes in the character state context, and Guénoche and Grandcolas 1999; Makarenkov 2001; Guénoche et al. 2004 in the distance context), rather than leaving them blank. However, although many solutions have now been proposed for the supertree problem (see, e.g., Bininda-Emonds 2004; Brinkmeyer et al. 2013, and the references therein) and the supermatrix problem for character state matrices (see, e.g., Bininda-Emonds 2004), relatively few approaches have been put forward to address the supermatrix problem for distance matrices that do not use an imputing scheme (see, e.g., DeSoete 1984; Gaul and Schader 1994; Guénoche and Grandcolas 1999; Makarenkov 2001; Guénoche et al. 2004; Criscuolo et al. 2006; Criscuolo and Gascuel 2008, for imputing-based solutions).

The various supertree and supermatrix approaches have differing pros and cons. For example, supermatrices have been criticized for the dependence of the generated supermatrix upon the order in which the missing values are inferred, and the potentially heavy influence of even a small imputation error on the tree topology, the latter due to a cascading effect

such an error might have on other inferred missing values (Lapointe and Levasseur 2004). Criticisms of supertrees include not using primary information, combining trees that have potentially evolved under different evolutionary models into a supertree without properly accounting for this disparity, and not properly taking into account the branch-lengths associated with the input trees (see Willson [2004] for an exception to this and Kupczok [2011] for a recent comparison of supertree methods).

In the form of the Lasso approach, we propose a novel method for rooted tree reconstruction from "partial distances" on data sets that are approximately clock-like. Contrary to the methods alluded to above, it is not imputing-based. Also, note that it bears no relation to the "Lasso regularized least squares method" in statistics (Tibshirani 1996), instead taking its name from prior theoretical phylogenetic research (e.g., Dress et al. 2011; Huber and Steel 2014). Lasso is similar in spirit to the supermatrix approach introduced in Misof et al. (2013) and the veto-supertree approach proposed in Scornavacca et al. (2008) in that not every taxon in the combined taxa set is guaranteed to be a leaf in the resulting tree. It essentially works by trying heuristically to detect a treelike signal on as many taxa as possible from the available distances and then reconstructing the "unique" equidistant tree on those taxa.

Like UPGMA, Lasso is an iterative process in the sense that it begins with a distance matrix D on some set X with $n \geq 2$ taxa with a graph G of n isolated vertices, each of which is labeled by a taxon in X. In each iteration step, the distance matrix on a smaller taxa set is recomputed and, in a bottom up manner, an equidistant tree on X is reconstructed. Central to this process is the identification of a subset of taxa that have minimal distance from one another. In contrast to UPGMA, Lasso works on both partial and complete distance matrices, whereas UPGMA can only take a complete distance matrix as input. Furthermore, Lasso replaces the minimal distance taxa subset by a composite vertex that can contain two or more taxa, whereas UPGMA only allows two. Finally, the distances between a newly created composite vertex and any other vertices are calculated by a consensus rather than by using average distance as in the case of UPGMA. These latter two differences ensure that Lasso enjoys several desirable properties such as "consistency," by which we mean that the equidistant tree T returned by Lasso is the unique tree which, for any two taxa x and y for which the distance value $D(x, y)$ is known (and which remains at the end of the relevant Lasso run), is the distance between them in T.

We assessed the performance of Lasso as a tree reconstruction approach from partial distance matrices using simulated data sets and a real biological data set containing 26 intraspecific strains of the wild yeast *Saccharomyces paradoxus* (West et al. 2014). In both studies, and independent of the shape of the topology of the starting equidistant tree considered in our simulation experiments, we found that even with 10% of the distance values missing Lasso was able to successfully reconstruct that tree. In addition, to illustrate Lasso's potential as a supertree approach, we applied it to a wheat accession data set which we obtained by combining molecular marker scores on 411 wheat accessions (generated as part of the GEDIFLUX EU Framework V project; Reeves et al. 2004) with a similar data set of 118 wheat accessions (Sayar-Turet et al. 2011), approximately a quarter of which are also included in the GEDIFLUX data set. We showed the resulting Lasso supertree to be highly congruent with the two input data sets, and furthermore to a greater or similar extent to those supertrees produced by five leading supertree methods. Finally, to enable other researchers to apply Lasso to their own data sets, we implemented the approach within software which, together with an accompanying manual, is freely available for download from https://www.uea.ac.uk/computing/lasso (last accessed February 19, 2015).

Results and Discussion

We refer to Materials and Methods for terminology and notation.

Tree Reconstruction from Simulated Partial Distance Matrices

The results of our missing value simulation study showed that, as expected, the normalized Robinson-Foulds distances between $T'|_Y$ and T increased for all three tree topology types (caterpillar, balanced, and Yule-Harding) with the percentage P_{miss} of missing values. We further discovered that, across all values of P_{miss}, equidistant caterpillar trees were reconstructed most accurately, with a mean normalized Robinson–Foulds distance below 0.1 even when 30% of the distance values were missing.

A potential reason for the superior performance of Lasso on caterpillar trees might lie in the way in which nonleaf vertices are reconstructed. To correctly reconstruct such a vertex v of T', the so-called child-edge graph associated with v must be a complete graph (Huber and Popescu 2013). It is straightforward to show that for a caterpillar tree (i.e., where all but one vertices possess two children with at least one leaf below—see also Semple and Steel [2003]), the likelihood that this condition holds is high, implying that Lasso often correctly reconstructs v. Conversely, a balanced tree has the most number of vertices where both children are leaves. Thus, the likelihood that the child-edge graph associated with such vertices is not a complete graph increases as the number of missing distance values grows, implying that T may be very different from $T'|_Y$. Furthermore, trees generated under the Yule–Harding model tend to be highly balanced (see, e.g., Semple and Steel 2003, Section 2.5). Therefore, it is unsurprising to observe that equidistant Yule–Harding trees and equidistant balanced trees exhibit a similar behavior. Interestingly, equidistant Yule–Harding trees were reconstructed slightly more accurately than equidistant balanced trees overall, presumably due to small departures from a purely balanced state.

Figure 4(*i*) suggests that for low quantities of missing distances (i.e., $P_{\text{miss}} \leq 10\%$), Lasso is very good at exploiting redundancy in a given distance matrix to correctly reconstruct the underlying equidistant tree, independent of the tree type. To better understand how much this observation depended on the starting trees not containing a polytomy, we also

investigated the influence of the maximal vertex out-degree k of such a tree on Lasso's performance. We summarize our results in figure 4(*ii*) in terms of the average percentage P_{leaves} of the elements in X that are also present in the leaf set of the equidistant tree (T,ω) returned by Lasso.

As expected, P_{leaves} is very high (i.e., above 90%) for all values of k if the quantity of missing distances is low (i.e., $P_{miss} \leq 10\%$). This observation is encouraging from a supertree perspective, as the out-degree of a vertex can be comparatively high in such trees. However, with an increasing proportion of missing distances, equidistant trees with a lower maximal out-degree appear to fare better overall. More precisely, in the case $k = 2$ (i.e., no polytomy) an equidistant tree returned by Lasso still contains over 80% of the leaves of T' even if 40% of the distance values are missing. To obtain a similar result for $k = 20$, only approximately 15% of missing distance values can be tolerated. A potential reason for this discrepancy might be that the likelihood of the child-edge graph of a high out-degree vertex not being complete increases quickly with a growing proportion of missing distance values.

Tree Reconstruction from a Yeast Partial Data Set

Applying either UPGMA or Lasso to the complete distance matrix from the yeast study (West et al. 2014) produced a tree very similar to that estimated by its authors (see supplementary fig. S4, Supplementary Material online), suggesting that the tree underlying the data set is indeed equidistant. We further present the equidistant tree returned by LASSO on the simulated partial distance matrix in figure 5. Note that this tree contains all 26 input taxa. Furthermore, its topology is highly similar to that produced, on the full distance matrix, within the original study (West et al. 2014). Most importantly, the groupings of the American, Far Eastern, and European strains are preserved, as is the separation of the United Kingdom- and non-United Kingdom-derived strains within the European group. Furthermore, the putative European/Far Eastern hybrid strains *N*_17 and *N*_45 are located within the tree at positions consistent with such an evolutionary history. Although some minor changes in topology are seen within the European and American groups, the relationships within the Far Eastern group are wholly preserved once 10% of distances have been removed.

As Lasso's ability to reconstruct an equidistant tree from a partial distance matrix depends both on which distances are missing and on which ties are broken at random by the algorithm, we also constructed a consensus tree for the yeast data set. The resulting consensus tree (fig. 6) is again highly congruent with the full distance matrix tree, and differs from figure 5 only in the relationship between the three European strains Q89_8, Q95_3, and S36_7. Noticeably, the support for the bifurcation of the latter two strains is the only one in figure 6 less than 74.

Supertree Reconstruction on Two Overlapping Wheat Data Sets

Next, we analyzed two partially overlapping wheat genetic marker data sets in order to evaluate the potential of Lasso as a supertree reconstruction approach. The equidistant trees, T_A and T_B, resulting from separate Lasso analyses of data sets A and B, were found to be supported by 77,577 (out of 84,255) and 6,844 (out of 6,903) distance values from d_A and d_B, respectively, and are shown in supplementary figures S5 and S6, Supplementary Material online. The trees within these figures also contain branches colored by population group membership, as described in Materials and Methods. For both data sets, we see that accessions belonging to the same population group are largely clustered within the equidistant trees. The large sizes of the two Lassos (encompassing 92.1% and 99.1% of distance values within d_A and d_B, respectively), together with the consistency of population grouping across the equidistant trees, strongly suggest the suitability of the Lasso approach in determining the genetic relationships between accessions within both of these data sets.

In total, the resulting partial distance matrix D contained 90,814 entries (where we exclude entries of the form $D(x, x)$ and only count entries of the form $D(x, y)$ and $D(y, x)$ once) which equates to 28.1% missing values of the potential 126,253 distance values for 503 taxa. We depict supertree S, estimated from D using Lasso, in figure 7 and remark in passing that it contains all 503 input taxa and that the size of the strong lasso returned by Lasso supporting S is 89,642. Put differently, S is the unique equidistant tree that displays correctly 98.7% of the 90,814 distance values for D.

Mantel tests comparing S with the two individual Lasso trees, T_A and T_B, showed a positive correlation of 0.57 and 0.47, respectively, with P-values for both being 0.0009990. These results indicate that S displayed relationships between accessions within the two data sets appropriately, including the overlapping accessions. The results of additional comparisons with five leading supertree approaches, together with characteristics of the algorithms assessed, are shown in table 1.

Notably, the PhySIC and PhySIC_IST algorithms failed to produce an adequate supertree for this data set, resulting in a star tree and the input trees, respectively, and were not considered further. Considering the remaining four approaches, although Lasso does not perform optimally for any of the distance measures, each of which can be thought of as representing a different aspect of tree comparison, it places second for the majority of the eight comparisons. Furthermore, when taking the average of pairs of comparisons over the two subdata sets (e.g., of $D_{RF}(T_A, S|_A)$ and $D_{RF}(T_B, S|_B)$), Lasso is the only one of the four algorithms that does not perform least well for any of the distance measures. It is also noticeable that other algorithms may perform much better on one sub-data set than on the other, whereas Lasso generally performs equally well on both. For example, the BuildWithDistances supertree is most similar to one sub-data set for three of the distance measures (i.e., $D_T(T_A, S|_A)$, $D_{SS}(T_B, S|_B)$, and $D_{AS}(T_B, S|_B)$) while also being least similar to the other sub-data set for the same distance measure (i.e., $D_T(T_B, S|_B)$, $D_{SS}(T_A, S|_A)$, and $D_{AS}(T_A, S|_A)$).

Although we did not compare the speed of the different approaches formally, due in part to our use of the third-party software EPoS (Griebel et al. 2008) to run two of them, we

Table 1. Characteristics and Results of Six Supertree Algorithms which Can Be Used to Construct Trees and Supertrees (or Both) from (Partial) Distances when Applied to the Wheat NBS Data Sets A and B.

Algorithm	Tree Method?	Supertree Method?	Edge-Weighted?	Adequate Supertree?	$D_{RF}(T_A, S_A)$	$D_{RF}(T_B, S_B)$
Lasso	✓	✓	✓	✓	0.311275	0.416309
Modified MinCutSupertree	×	✓	×	✓	0.505590	0.441441
BuildWithDistances*	×	✓	✓	✓	0.049261	0.381974
FlipSupertree*	×	✓	×	✓	0.068627	0.253219
PhySIC	×	✓	×	×	—	—
PhySIC_IST	×	✓	×	×	—	—

Algorithm	$D_T(T_A, S_A)$	$D_T(T_B, S_B)$	$D_{SS}(T_A, S_A)$	$D_{SS}(T_B, S_B)$	$D_{AS}(T_A, S_A)$	$D_{AS}(T_B, S_B)$
Lasso	0.000346	0.002159	0.926605	0.887092	0.473567	0.457534
Modified MinCutSupertree	0.000520	0.001363	0.875586	0.931511	0.454063	0.477384
BuildWithDistances*	0.000297	0.002394	0.992379	0.878096	0.497626	0.452113
FlipSupertree*	0.000539	0.001936	0.981233	0.924485	0.492873	0.470559
PhySIC	—	—	—	—	—	—
PhySIC_IST	—	—	—	—	—	—

Note.—Methods denoted with an asterisk refer to those versions of the relevant algorithm as implemented within the EPoS software (Griebel et al. 2008).

noted that FlipSupertree performed most rapidly on this data set (<10 s), whereas both Lasso and modified MinCutSupertree analyzed it in tens of seconds, with BuildWithDistances taking tens of minutes. In conclusion, Lasso performs as well as (or in some cases better than) five leading supertree approaches on this challenging data set, even though it was not developed specifically for the reconstruction of supertrees. Furthermore, its returned supertree possesses several desirable properties and characteristics (e.g., edge-weights, uniqueness), not all of which could be produced or guaranteed by the alternative approaches.

Concluding Remarks

In this article, we introduce the novel Lasso approach for distance-based equidistant phylogenetic tree reconstruction from partial distance matrices. Furthermore, we illustrate its potential as a supertree reconstruction approach. Computer code for the Lasso algorithm, together with an accompanying manual, has been developed and is freely available from https://www.uea.ac.uk/computing/lasso.

Lasso is similar in spirit to UPGMA but takes as input either partial or complete distance matrices, as opposed to the complete distance matrices required by UPGMA. It aims to reconstruct a unique (in a well-defined sense), equidistant tree by exploiting redundancy in a given distance matrix rather than by trying to estimate missing distance values, as do approaches such as that presented in Criscuolo and Gascuel (2008). Given the growing number and size of high-throughput biological data sets, and the prevalence of accompanying missing data, the availability of an approach that can rapidly handle such data sets is timely. Furthermore, Lasso trees can be exploited as starting trees within a search of tree space, particularly when undergoing analysis by computationally intensive phylogenetic methods such as Maximum Likelihood and Bayesian Inference (Burbrink and Castoe 2009; Bouckaert et al. 2014). Indeed this utility could be viewed as setting the scene for the development of novel likelihood-based approaches to the phylogenetic analysis of partial data sets, exploiting Lasso as a guide tree in order to limit tree parameter space exploration.

We assessed the performance of Lasso as a tree reconstruction approach within a simulation study. We considered three different types of binary equidistant tree and found that, independent of the tree type, Lasso performed strongly when 10% or fewer of distance values were missing. For higher percentages of missing distance values, performance was strongly affected by the equidistant tree type. We observed that the equidistant caterpillar tree was recovered most accurately under this scenario, with the equidistant Yule–Harding and balanced trees faring less well. We also found that Lasso performed well when the equidistant tree underlying a given partial distance did not possess polytomies of too high a degree in the presence of a high proportion of missing values. For example, even with 10% of distance values missing for an underlying equidistant tree with vertices of maximal out-degree 20, Lasso was still able to return a tree on more than 90% of the original taxa.

We also assessed the performance of Lasso on two real biological data sets. In the first of these studies, a yeast data set derived from a whole-genome resequencing study, originally developed and analyzed in West et al. (2014), was successfully analyzed with Lasso, even with 10% of distance values removed at random. Furthermore, we showed that Lasso's performance was robust to the choice of missing values. In the second study, Lasso was used to construct a supertree of two partially overlapping wheat NBS marker data sets (Reeves et al. 2004; Sayar-Turet et al. 2011). Subsequent statistical tests showed both that the Lasso supertree appropriately displayed relationships found within the two original data sets and that it was as congruent with the two input trees, or in some cases more so, as those supertrees constructed by five leading supertree algorithms. Importantly, for both of these data sets Lasso was able to return trees comprising 100% of the starting taxa. Collectively, these studies suggest that Lasso could be a highly useful method for both tree and supertree reconstruction on real data sets.

It is interesting to speculate that the strong performance of Lasso in a supertree context owes part of its success to its ability to reject shared distances that are not highly correlated. Indeed, when we repeated our Mantel tests comparing the equidistant supertree derived from the full combined data set (i.e., not rejecting any shared distance values) to the two separate distance matrices (d_A and d_B), the correlations were found to be 0.47 ($P = 0.0009990$) for both data sets. Although this performance is almost identical for data set *B*, we see that removing certain shared distances leads to a highly improved result for data set *A*, which we earlier noted possessed a lower proportion of distance values within the Lasso. In future, an investigation of methods to combine data sets for supertree construction would be highly valuable, to see whether a further improvement could be obtained.

Additional future work should also focus on understanding the affects of data error upon Lasso tree reconstruction. Of course, the Lasso algorithm makes decisions regarding which distances contribute toward the final strong lasso, which could be viewed as a form of error smoothing. Consequently, some information regarding potential sources of error may be derived directly from algorithm runs. In addition, simulation studies with carefully controlled error models will enable us to assess the accuracy of Lasso in a range of scenarios and parameter spaces.

Further studies might include developing new methods to update the distance matrix *D* (i.e., Reduction step (ii)) during a Lasso iteration. For example, it might be interesting to relax the consistency requirement slightly by saying that the distances in the computed tree will only be approximately equal to the distances given in the partial distance matrix for each outputted cord. Such a relaxed result could be obtained, for example, by removing outliers using a standard statistical method and then computing the mean. The advantage of this distance would be that Lasso would remain robust to noisy data, potentially alleviating some of the error effects that we plan to investigate. Other distance matrix updating methods might enable a greater number of taxa to be included in the returned strong lasso for some data sets, although this might come at the expense of a greater computation time. Also it might be interesting to investigate the Lasso approach in a "relaxed" molecular clock framework (Drummond et al. 2006).

Finally, we are currently developing NGS data sets for a large number of yeast strains, where polyploidy and recombination are known complications within genetic analyses. Consequently, a study on the performance of Lasso on SNP data sets derived from our NGS reads, based on a range of distance measures, would be highly interesting. We could, for example, compare the accuracy of Lasso when applied to (partial) distances such as those introduced recently in Joly et al. (2015) for SNP data sets with older methods developed for allele frequences (Cavalli-Sforza and Edwards 1967).

In summary, we propose the Lasso approach, and accompanying computer software, as a key new method within the molecular phylogenetics toolkit. Given its demonstrated potential both in tree reconstruction in the face of missing data, and in supertree reconstruction, we believe that it can play an important role in analyzing the next generation of biological data sets.

Materials and Methods

To help explain the inner workings of Lasso, we first introduce some relevant terminology and then present an outline of the Lasso approach. For the convenience of the reader, we also present a worked example.

Basic Terminology and Assumptions

We assume throughout this section that X is a set containing at least two taxa. Furthermore, a rooted tree T whose leaf set is X is a "phylogenetic tree" *on* X. In such a tree T, no vertex connects only two edges except for a distinguished vertex which we call the "root" and denote by ρ_T.

An "edge weighting" ω of T is a map that assigns a positive real number to every edge of T. The pair (T,ω) is then called an edge-weighted (phylogenetic) tree on X. Following Semple and Steel (2003), we call an edge-weighted tree on X equidistant if 1) the distance (i.e., the sum of edge-weights on a path) between the root and every leaf is the same, and 2) for all taxa x in X, the distance between a vertex u and x is larger than the distance between a vertex v and x whenever u is encountered before v on the path from the root ρ_T to x. For convenience, we will sometimes refer to the underlying graph of an equidistant tree as its topology.

To illustrate these definitions, consider the tree T depicted in figure 1(*ii*), a phylogenetic tree on the taxa set $X = \{a, \ldots, f\}$. If the edge-weighting ω assigns the value 1 to every edge of T, then (T,ω) is rendered an equidistant tree on X. Consequently, for example, the distance between a and d is 4.

Next, suppose $\mathcal{L}$ is a *set of cords* on X, that is, a subset of all possible pairs of taxa in X. It is often helpful to view such a set $\mathcal{L}$ as a graph $\Gamma(\mathcal{L})$, which we alternatively term $\Gamma(\mathcal{L})^w$ when an edge-weighting w has been assigned. In such a graph, which we term a "graph of cords," taxa x and y in X are only joined by an edge if $xy \in \mathcal{L}$. Figure 1(*i*) illustrates such a representation for the set $\mathcal{L} = \{bd, bf, df, cd, ab, ef\}$.

Furthermore, we say that $\mathcal{L}$ is a *strong lasso* for T if $\mathcal{L}$ uniquely determines both the topology and an equidistant edge-weighting of T (we refer the reader to the Appendix for a precise definition of this concept). For example, for $X = \{a, \ldots, f\}$ and $\mathcal{L} = \{bd, bf, df, cd, ab, ef\}$, the latter depicted as graph $\Gamma(\mathcal{L})$ in figure 1(*i*), it follows from Huber and Popescu (2013) that $\mathcal{L}$ is a strong lasso for the tree T depicted in figure 1(*ii*). However, once we remove the cord df from $\mathcal{L}$ then the resulting cord set $\mathcal{L}'$ is no longer a strong lasso for T. This result follows as both T and the tree T' depicted in figure 1(*iii*) agree on the distances between the taxa pairs in $\mathcal{L}'$, thus violating the uniqueness property required of a strong lasso.

The Lasso Algorithm

Here, we present an outline of the Lasso algorithm (see the Appendix for precise definitions and pseudocode and the

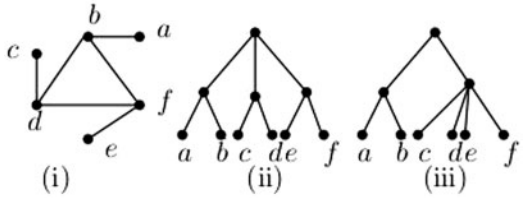

Fig. 1. The graph $\Gamma(\mathcal{L})$ in (*i*) represents the set of cords $\mathcal{L} = \{bd, bf, df, cd, ab, ef\}$ on $X = \{a, \ldots, f\}$. Consider two trees, T in (*ii*) and T' in (*iii*), to which the edge-weightings ω and ω' both assign weight 1 to each edge, respectively. $\mathcal{L}$ is a strong lasso for T. However, $\mathcal{L}' = \{bd, bf, cd, ab, ef\}$ is not a strong lasso for T, as the distance induced on T by any two elements of a cord in $\mathcal{L}'$ is identical to that induced on T'.

supplementary material, Supplementary Material online, for a more formal presentation of Lasso). As Lasso employs a similar strategy to UPGMA, we begin by giving a brief outline of the latter method.

UPGMA is a hierarchical clustering method on some taxa set X (of size $n \geq 2$) for which a complete distance matrix D is known. It begins with a graph G of n isolated vertices, each labeled by a taxon in X. Through a series of $n-1$ iterations, a phylogenetic tree that represents D is grown in a bottom up fashion. Within each iteration, a reduction step and a construction step are carried out. In the reduction step, the two closest elements x and y are replaced by a composite element $\nu_{x,y}$ and an average measure of distance is used to determine the distance from a composite vertex to the other vertices in the distance matrix, thereby obtaining a new distance matrix D' on the reduced taxa set X'. In the construction step, an equidistant tree whose root is labeled by $\nu_{x,y}$ is created by adding new edges from x and y to $\nu_{x,y}$ (noting that one or both of x and y might themselves be the roots of equidistant trees constructed in previous iterations). UPGMA is known to return the tree underlying a given distance matrix if the (complete) distance matrix used as input obeys the molecular clock assumption and the underlying equidistant tree does not possess polytomies (Durbin et al. 1998).

The Lasso approach is also an iterative approach consisting of a reduction step and a construction step. Furthermore, these steps both serve a similar purpose to their counterparts in UPGMA. In contrast to UPGMA, Lasso takes as input a partial distance matrix on X (which can of course also be complete). The Lasso algorithm proceeds as follows.

Method Outline

Given a partial distance matrix D on some taxa set X, Lasso heuristically finds a subset $Y(\subseteq X)$ of taxa as large as possible such that the equidistant tree it returns is uniquely determined, with regards to topology and edge-weighting, by the available distances on Y. To achieve this, Lasso employs the following recursive strategy.

Construction step:

i) One or more sets of cords on X that satisfy certain properties, and which we term *cliques*, are identified and stored in the set $\mathcal{C}$ (i.e., $\mathcal{C} = \{K_1, \ldots, K_r\}$, for some positive integer r, where each element K_i is a set of cords on X). These properties are as follows:

a) The graph $\Gamma(K_i)$ of the cords in a given K_i is connected and, furthermore, any two of its vertices are joined by an edge (i.e., the set of cords K_i is a *clique*).

b) All cords in a given K_i that satisfies property (a) possess the current smallest edge-weight m according to the distance matrix D (i.e., the set of cords K_i is a *clique* with minimal distance).

For instance, for the worked example discussed below, m = 2 and the set $\mathcal{C}$ comprises the sets $K_1 = \{bc, bd, cd\}$ and $K_2 = \{ab\}$.

ii) If $\mathcal{C}$ comprises only cliques that contain a single vertex, the algorithm terminates and the current equidistant tree(s) on taxa Y and the respective strong lasso(s) are saved. Otherwise, an optimal clique K is chosen from the elements in $\mathcal{C}$. When $\mathcal{C}$ contains more than one clique, the largest is chosen heuristically or, if two or more possess the largest size, one is chosen at random. We term K the "suitable clique" for the current iteration. For example, in the worked example alluded to above, $K = K_1 = \{bc, bd, cd\}$ as it is the larger of the two identified cliques.

iii) An equidistant tree (U,ω) is grown from K. First, each of the vertices v of K is joined through a new edge to a new root vertex u, thereby obtaining the tree U. The vertices in K belong to one of two categories in that they either represent those vertices present in the starting taxa set or that they represent composite vertices w from previous iterations, which will have been the roots of equidistant trees T_w constructed within those iterations. Let v_1 represent a vertex of the former category and v_2 a vertex of the latter. Next, the "root height" h_v of each vertex v in K is determined. For each vertex v_1 in K, where a tree with v_1 as the root can only consist of v_1 itself, $h_{v1} = 0$. For each vertex v_2 in K, $h_{v_2} = D(x, y)/2$, where x and y are leaves of T_{v_2} such that v_2 lies on the path joining x and y. Finally, we assign edge-weights to all edges in U. Each edge in U that was an edge in a T_{v_2} from a previous iteration is assigned its previous weight in that tree. All new edges, connecting vertex v to the root vertex u, are assigned the weight $(m/2) - h_v$.

iv) Move on to the Reduction step.

Reduction step:

i) All taxa that are vertices of K are replaced with a new, composite element u. An updated taxa set X' is found by removing all taxa in K and adding u (i.e., the size of the taxa set may reduce by more than one in a single step).

ii) An updated distance matrix D' is calculated for the new X'. For any two taxa x and y that are not vertices in K, we keep $D'(x, y) = D(x, y)$ and set $D'(u, u) = 0$. Next, for each vertex x not in K we define $D^*(u, x)$ to be the mode of all distances $D(v, x)$, where v is any

vertex in K and ignoring any distance $D(v, x) = m$ which effectively removes x from X' (this latter condition ensures that the set of cords returned by Lasso is correct for the equidistant tree grown by it). Where we have a tie for the modal distance, one is chosen at random. Finally, we set $D'(u, x) = D^*(u, x)$.

iii) Set $X = X'$ and $D = D'$ and return to the Construction step.

Figure 2 illustrates the concepts of a "clique" and a "suitable clique." Note that the important property (Construction step (i)b) that the cords in a suitable clique are mutually closest to one another makes this step analogous to the Construction step of UPGMA, where the two closest vertices are selected. Within Reduction step (ii), other choices for D^* are conceivable, such as defining $D^*(u, x)$ to be the distance in which one has most confidence over all distances $D(v, x)$. A further alternative is to take the mean over all distances $D(v, x)$, as for average linkage. However, it should be noted that the latter case cannot be guaranteed to give rise to a strong lasso for the returned equidistant tree as the mean need not be one of the original distance values. Finally, note that the heuristic nature of the algorithm (and in particular the potential for random decisions in Construction step (ii) and Reduction step (ii)), means it will not always return the same equidistant tree for a given data set (e.g., the number of leaves may differ between consecutive runs). Consequently, the algorithm is currently set to run a default ten times and to return the equidistant tree (plus underpinning strong lasso) with the largest number of leaves, again breaking ties randomly.

An Example

To illustrate the Lasso approach, imagine that $X = \{a, \ldots, e\}$ is a taxa set and that D is a partial distance matrix on X given in terms of the edge-weights of the graph $\Gamma(\mathcal{L})^w$ presented in figure 3(*i*). The set $\mathcal{C}$ computed in Construction step (i) of the first iteration contains two cliques, with cord sets $\{ab\}$ and $\{bc, bd, cd\}$, respectively, with each cord in each cord set possessing the minimal edge-weight $m = 2$. The larger of the two cliques is shown in figure 3(*ii*), and is thus chosen as the suitable clique K. As the trees with roots b, c, and d, respectively, are the vertices b, c, and d themselves, h_b, h_c, and h_d all equal zero. The equidistant tree (U, ω) grown by Lasso in Construction step (iii) is then that depicted in figure 3(*iii*). Here, the weights of the new edges joining b, c, and d to root u are each assigned the value $(m/2) - h_v = (2/2) - 0 = 1$. In the Reduction step, the set X is updated to $X' = \{u, a, e\}$. In updating D to D' the cord ab is effectively removed, as it possesses the edge-weight 2 (i.e., m), thereby leaving $D'(u, a) = D^*(u, a) = 4$. A random choice between edge-weights 6 and 8 is made for the distance $D^*(u, e)$. In this example, we choose to set $D'(u, e) = D^*(u, e) = 6$, effectively deleting the cord de. The distance D' on this new set X' is represented in terms of the graph $\Gamma(\mathcal{L}')^w$ displayed in figure 3(*iv*). This completes the first iteration and we return to the Construction step.

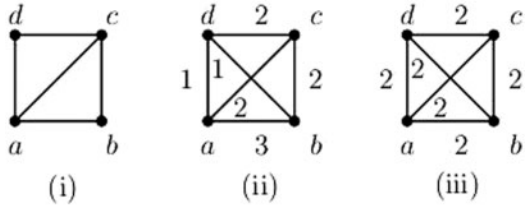

Fig. 2. (*i*) The graph $\Gamma(\mathcal{L})$ of $\mathcal{L} = \{ab, ac, ad, bc, cd\}$ is not a clique as the edge joining vertices b and c are missing. Conversely, the graph $\Gamma(\mathcal{L}')$ of $\mathcal{L}' = \{ab, ac, ad, bc, bd, cd\}$ is a clique. However, its weighted version $\Gamma(\mathcal{L}')^w$ with edge-weighting w as indicated in (*ii*) is not a suitable clique as not all edges have the same (minimum) weight. (*iii*) The graph $\Gamma(\mathcal{L}')^w$ of $\mathcal{L}'$ with w as indicated is a suitable clique as all pairs of edges are joined by cords and all edge weights are the same and have minimal weight.

As the set $\mathcal{C}$ in the second iteration consists of precisely one set of cords (i.e., the set $\{ua\}$), we choose this set to be K. We next grow an equidistant tree (U', ω') by creating new edges from u and from a to a new root vertex u', noting that $m = 4$ within this iteration. We then calculate that $h_a = 0$ and $h_u = D(b, c)/2 = 1$. Consequently, we assign the weight $(m/2) - h_a = (4/2) - 0 = 2$ to the edge connecting a to u' and $(m/2) - h_u = (4/2) - 1 = 1$ to the edge connecting u to u'. The resulting tree (U', ω') is shown in figure 3(v). We complete the second iteration by updating X to the set $\{u', e\}$ and the distance matrix D to $D(u', e) = 6$, as depicted in figure 3(vi), and return to the Construction step.

In the third iteration, the set $\mathcal{C}$ again consists of a single set of cords (i.e., the set $\{u'e\}$) which we choose to be K. We next grow an equidistant tree (U'', ω'') by creating new edges from u' and from e to a new root vertex u'', noting that $m = 6$ within this iteration. We calculate that $h_e = 0$ and $h_{u'} = D(a, d)/2 = 2$ and we assign weight $(m/2) - h_e = (6/2) - 0 = 3$ to the edge connecting e to u'' and $(m/2) - h_{u'} = (6/2) - 2 = 1$ to the edge connecting u' to u''. The resulting tree (U'', ω'') is shown in figure 3(vii). We complete the third iteration by updating X to the set $\{u''\}$ and the distance matrix D to $D(u'', u'') = 0$ and return again to the Construction step.

As the set $\mathcal{C}$ in the fourth iteration contains only one clique and that clique has one vertex, Lasso terminates and saves the tree (U'', ω'') which is strongly lassoed by the available distances on $Y = X = \{a, b, c, d, e\}$. We depict those distances in terms of the graph $\Gamma(\mathcal{L}_Y)^{w''}$ in figure 3(viii) and start the next run of Lasso. As the only decision taken at random was a choice between cords $\{ce\}$ and $\{de\}$ within the Reduction step of the first iteration, it follows that the only alternative Lasso tree to that shown in figure 3(vii) would possess an identical topology but the edge-weights of cords $\{u''e\}$ and $\{u''u'\}$ would instead be 4 and 2, respectively.

Lasso's Performance in the Face of Missing Data

Missing data are a key problem in biological research. Although Lasso enjoys theoretical features that may help to mitigate this problem (see Appendix for details), its success in dealing with missing data in practical situations has yet to be tested formally. Therefore, to assess the performance of Lasso

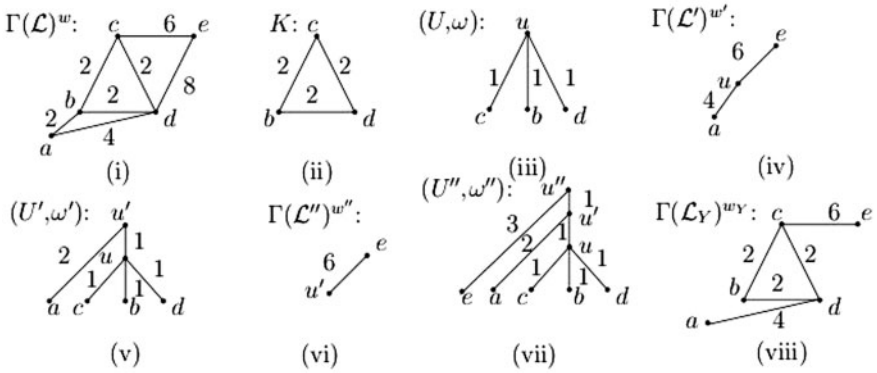

FIG. 3. The graph $\Gamma(\mathcal{L})^w$ in (*i*) represents the weighted set of cords $\mathcal{L} = \{ab, ad, bc, bd, cd, ce, de\}$ on $X = \{a, \ldots, e\}$. The suitable clique K identified in the first iteration of LASSO upon the corresponding partial distance matrix D, and consisting of the cords $\{bc, bd, cd\}$, is shown in (*ii*). For clarity, we have also included edge-weights. From K, an equidistant tree (U, ω) (*iii*) is subsequently grown and the taxa set and associated partial distance matrix are reduced to those represented by graph $\Gamma(\mathcal{L}')^{w'}$ in (*iv*). The equidistant tree (U', ω') grown in the second iteration of LASSO is shown in (*v*), with the reduced taxa set and partial distance matrix represented by graph $\Gamma(\mathcal{L}'')^{w''}$ in (*vi*). The third iteration of LASSO results in the equidistant tree (U'', ω'') shown in (*vii*) which is the final tree and thus returned by LASSO. Finally, $\Gamma(\mathcal{L}_Y)^{w_Y}$ in (*viii*) depicts the strong lasso returned for D rendering the tree shown in (*vii*) the unique tree that correctly represents the distance indicated in (*viii*).

in reconstructing trees from partial distance matrices, while controlling key aspects of the input data, we carried out a simulation study similar in spirit to that presented in Criscuolo and Gascuel (2008). Furthermore, we applied LASSO to a yeast data set (West et al. 2014) recently developed from a whole-genome resequencing study (Liti et al. 2009), in order to gauge its performance on a real biological data set.

Tree Reconstruction from Simulated Partial Distance Matrices

To understand how the topology of an equidistant tree affects our ability to reconstruct it from a partial distance matrix, we generated three distinct binary tree types: Balanced trees, caterpillar trees, and trees generated using the Yule–Harding model. Partial data set simulation followed a three-step process. In step 1, we implemented the approach described in Semple and Steel (2003, Section 2.5) to produce unweighted trees of the required topology type. In step 2, we turned each of the resulting trees into an equidistant tree. For balanced trees, this meant assigning weight 1 to all edges. For caterpillar trees, we instead assigned the difference in height between two adjacent vertices to the weight of the joining edge. The Yule–Harding tree case was slightly more complex and proceeded as follows. Starting with a Yule–Harding tree T, we first assigned to every vertex v of T its height in T, that is the number $h(v)$ of edges on a longest path from v to a leaf of T below v, where we put $h(v) = 0$ in cases where v was a leaf. For e an edge of T joining two vertices u and v, we then assigned $|h(u) - h(v)|$ as weight to e. In the final step, an incomplete distance matrix was generated for each tree. This was carried out by randomly removing a given percentage P_{miss} of entries from the (complete) distance matrix induced from each tree, while ensuring that the graph $\Gamma(\mathcal{L})$ of the associated set of cords remained connected.

Using this process we generated 2,500 incomplete distance matrices for each of the three equidistant tree types, 500 for each of the P_{miss} values of 1%, 5%, 10%, 20%, and 30%. Across all simulations, we took the size of the leaf sets to be 128. We then used the resulting 7,500 partial distance matrices as input to LASSO. Each equidistant tree found by LASSO was then compared with the respective equidistant tree (T', ω') used to generate the underlying input matrix. More precisely, for Y denoting the leaf set of a tree (T,ω) returned by LASSO, we computed the Robinson–Foulds distance (Robinson and Foulds 1981) $D_{RF}(T, T'|_Y)$ between T and the restriction $T'|_Y$ of T' to Y. This amounted to counting the number of clusters induced by $T'|_Y$ but not by T and vice versa. We then normalized these distances by dividing them by the maximal Robinson–Foulds distance between two trees on X, $2(n - 2)$, where n denotes the number of elements in X. For each of the 15 equidistant tree type and percentage P_{miss} combinations, we then calculated the mean of the relevant 500 normalized Robinson–Foulds distances, with the resulting mean values depicted in figure 4(*i*). We also refer the reader to supplementary tables S1–S3, Supplementary Material online, for some simple statistical measures on the supporting strong lassos.

Finally, we investigated the influence of a tree's maximal vertex out-degree k on LASSO's performance, where by vertex out-degree we mean the number of edges starting at that vertex. Specifically, we generated 500 random equidistant trees (T', ω'), 125 for each of the maximum vertex out-degree values of $k = 2$, 5, 10, and 20, as described in Algorithm 1 (supplementary material, Supplementary Material online). For all 500 trees, the number of taxa in X was 100.

Tree Reconstruction from a Yeast Partial Data Set

To test LASSO on a real biological data set, again as a tree reconstruction approach in the face of missing data, we applied it to a distance matrix generated for the analysis of several intraspecific strains of yeast. In West et al. (2014), the authors identified both fully and partially resolved SNPs (i.e., SNPs and pSNPs) within the ribosomal DNA (rDNA) tandem arrays of 26 strains of the wild yeast *Saccharomyces paradoxus*. Within this study, a distance matrix was constructed from the resulting allele frequency data set using

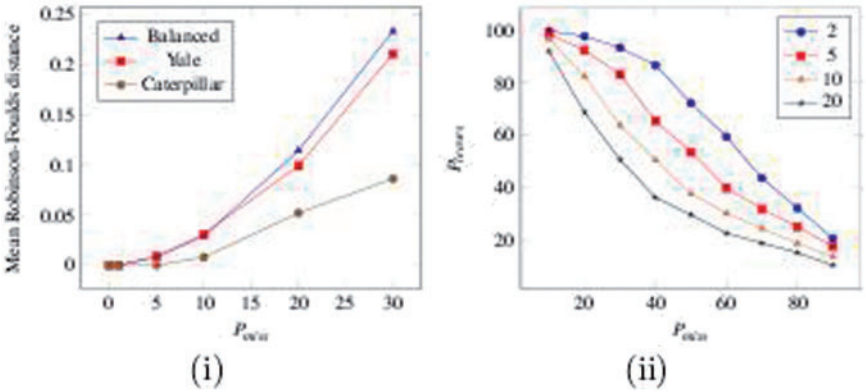

FIG. 4. (*i*) For all three equidistant tree types with 128 leaves, we plot the mean normalized Robinson–Foulds distances between T and $T'|_Y$, over data sets with varying proportions of missing values. (*ii*) For T', a tree with 100 leaves and maximum out-degree $k = 2$, 5, 10, and 20, we depict the proportion of X which forms the leaf set of T.

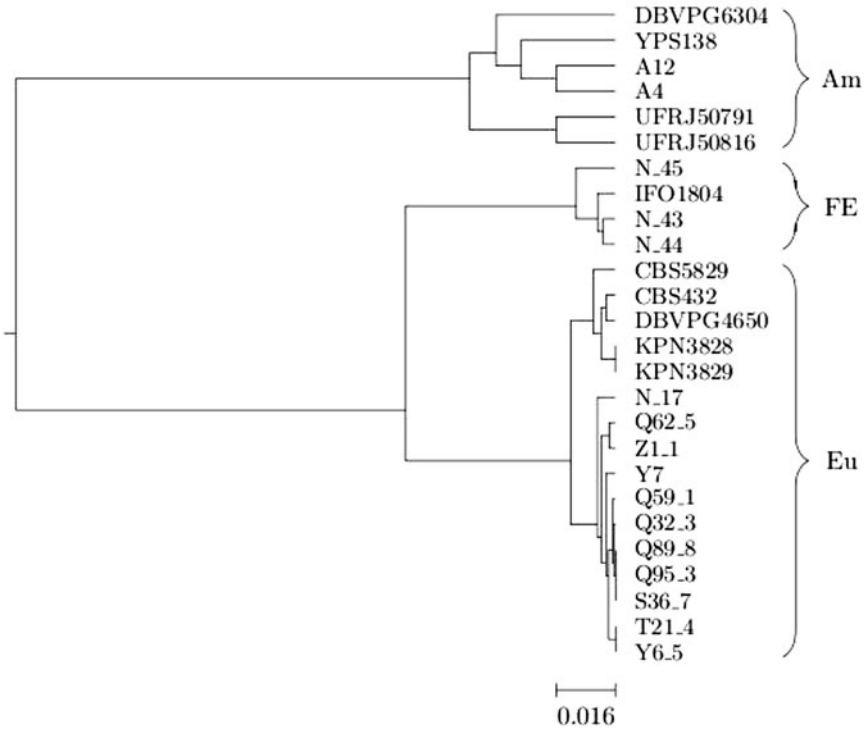

FIG. 5. An equidistant tree returned by LASSO from the yeast data set with 10% of the distances removed at random. The 16 European strains are denoted by the label "Eu," the 4 Far Eastern strains by "FE," and the 6 American strains by "Am." The uppermost five European strains (CBS5829 to KPN3829), together with N_17, derive from outside the United Kingdom, with the remaining ten European strains having been isolated within the United Kingdom.

the Cavalli-Sforza and Edwards Chord distance measure (Cavalli-Sforza and Edwards 1967) and a phylogenetic tree was estimated using Neighbor Joining. The tree was rooted by analyzing rDNA variation in S288c, the type strain of the closely related baker's yeast *Saccharomyces cerevisiae*.

We first applied both UPGMA and LASSO to the complete distance matrix from this study. From the distance matrix D induced by the LASSO-tree we then randomly removed 10% of the distance values ensuring that 1) whenever we removed for two strains x and y the distance value $D(x, y)$ we also removed the distance value $D(y, x)$, 2) values of the form $D(x, x)$ for strain x did not count toward the removed 10%, and 3) the graph $\Gamma(\mathcal{L})$ remained connected (where $\mathcal{L}$ denotes the pairs of taxa between which distance values are available).

Next, we constructed a consensus tree for the yeast data set, using an approach similar to bootstrapping. Within this approach we counted the number of times clusters induced by nonleaf (and nonroot) vertices were displayed on equidistant trees returned by LASSO. More precisely, we generated 100 partial distance matrices with 10% missing values chosen at random, as described above. We then ran LASSO on each

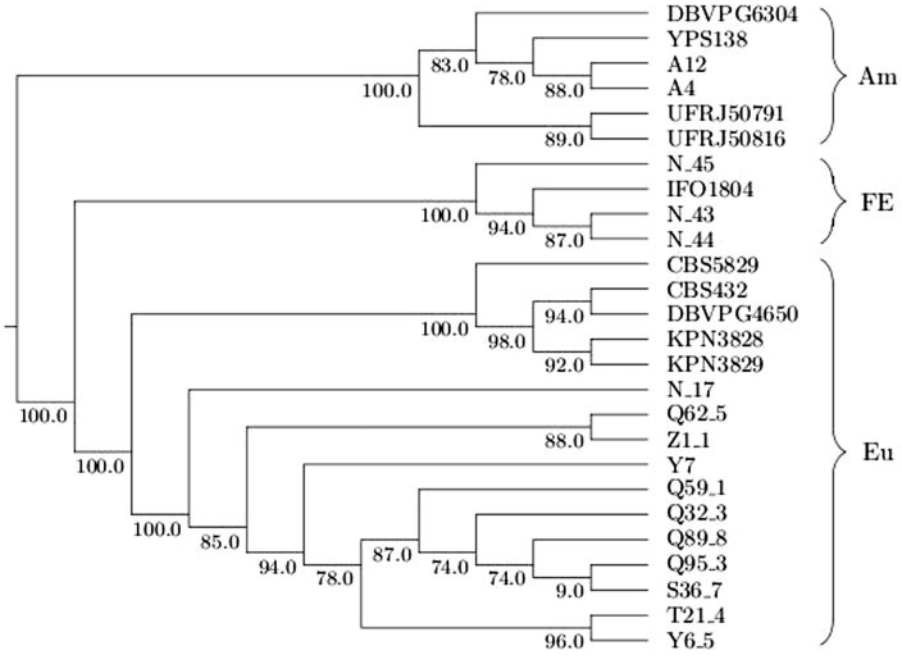

Fig. 6. Consensus tree built from 100 runs of Lasso on matrices with 10% of the distance values removed at random. The number next to a vertex shows the number of times the cluster induced by that vertex appeared in the input of Consense. The length of an edge is of no relevance.

partial distance matrix, resulting in a total of 100 equidistant trees each supported by, on average, a strong lasso with 205 cords (out of the possible 293). The resulting trees were then used as input to the Consense program (Felsenstein 2004) with default settings to build a consensus tree using the "majority rule (extended)" option.

Lasso as a Supertree Reconstruction Method

We further assessed the potential of Lasso as a supertree reconstruction approach by combining two partially overlapping wheat data sets developed in distinct studies (Reeves et al. 2004; Sayar-Turet et al. 2011). The first data set (to which we will refer as data set *A*) consists of 57 NBS (nucleotide binding site) markers scored over 411 accessions, a subset of a data set developed within the GEDIFLUX EU Framework V project (Reeves et al. 2004) to assess genetic diversity over time across four major crops, including wheat. The second data set (to which we will refer as data set *B*) consists of 71 NBS markers scored over 118 accessions, a subset of a study comparing genetic diversity within and between winter wheat accessions from Turkey, Kazakhstan, and Europe (Sayar-Turet et al. 2011), the latter comprising a small group of the GEDIFLUX wheat accessions. Consequently, the two data sets share a common set of 26 European winter wheat accessions, comprising 6.3% and 22.0% of their accessions, respectively. We will refer to this shared data set as data set *C*. Equidistant trees were estimated for data sets *A* and *B* separately, using the Modified Rogers measure (Reif et al. 2005) to calculate a (complete) distance matrix, followed by tree construction with Lasso. For convenience, we denote the two distance matrices as d_A and d_B, where the index indicates the data set to which they refer.

We then assessed the individual Lasso trees according to population group data for the two data sets. In the original analysis of data set *B* (Sayar-Turet et al. 2011), the model-based clustering method STRUCTURE (Pritchard et al. 2000) was carried out to estimate the number of founder populations underlying the data set and the genetic contribution of each population to each accession. We colored branches of the Lasso tree (see supplementary fig. S6, Supplementary Material online) such that the color of each branch corresponded to the main population group to which the relevant accession belonged. For data set *A*, we conducted our own population structure analysis, here using the ADMIXTURE method (Alexander et al. 2009) with default parameter values. ADMIXTURE uses an identical genetic model to STRUCTURE, but a different computational approach to optimize population parameters, rendering it considerably faster to run. See supplementary table S7, Supplementary Material online, for the membership (Q matrix) of the three population groups determined by ADMIXTURE for each accession in data set *A*, together with an indication of the group that is inferred to have contributed most to each accession's genetic material. The Lasso tree for data set *A*, colored according to these groups, is shown in supplementary figure S5, Supplementary Material online.

We next obtained a (partial) distance matrix *D* on the combined data set of $411 + 118 - 26 = 503$ accessions, proceeding as follows. If *x* and *y* were accessions such that one of them was contained in data set $A \backslash C$ (i.e., in *A* but not in *C*) and the other in *A* then we put $D(x, y) = d_A(x, y)$. Similarly, if one of them was contained in data set $B \backslash C$ and the other in *B* then we put $D(x, y) = d_B(x, y)$. For the remaining case that both accessions were contained in the overlap we took the mean, that is, we put $D(x, y) = (d_A(x, y) + d_B(x, y))/2$. To

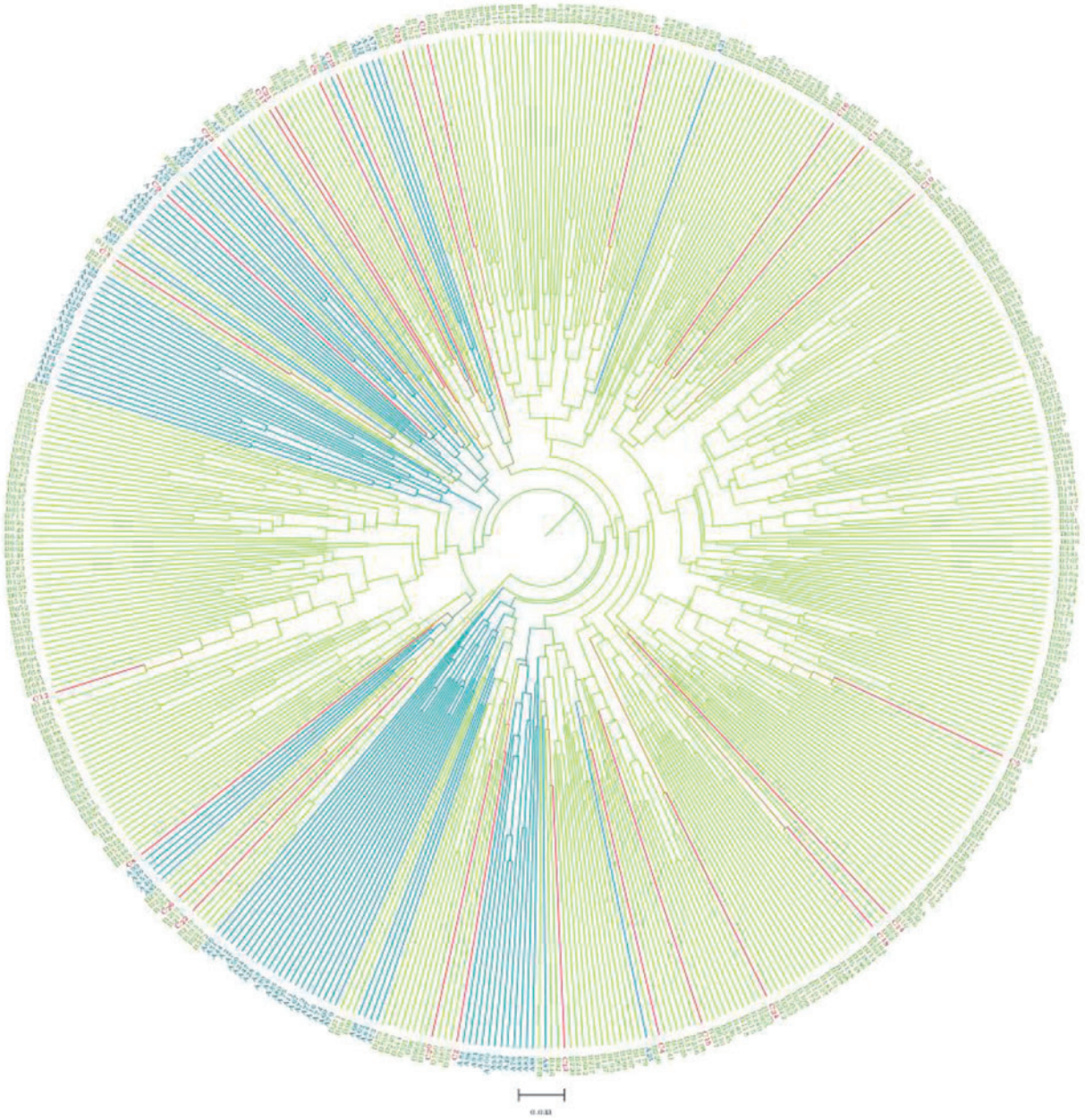

FIG. 7. The equidistant supertree built by LASSO for the two wheat data sets. Accessions from the GEDIFLUX data set (*A*) are indicated by green branches, those from the Turkish data set (*B*) by blue branches, with the 26 accessions found in both data sets (*C*) indicated by red branches. Note that the shared accessions are spread across the supertree and that the tree contains all 503 input taxa.

mitigate against the fact that, for some accessions in *C*, the distance values $d_A(x,y)$ and $d_B(x,y)$ correlated more strongly than for others we used the ratio $d_A(x, y)/d_B(x, y)$ to identify outliers, which we subsequently removed from the analysis. For this, we calculated the empirical distribution of these ratios and defined a distance value to be an outlier if it was more than one interquartile range above the upper quartile or one interquartile range below the lower quartile. We then used the resulting (partial) distance matrix as input to LASSO, thereby obtaining a supertree *S* on the combined data set.

Finally, we assessed the consistency of supertree *S* with the two original distances matrices, d_A and d_B, and we compared the similarity of its topology to the two individual LASSO trees T_A and T_B against those of supertrees constructed using five leading alternative algorithms: Modified MINCUTSUPERTREE (Page 2002), BUILDWITHDISTANCES (Willson 2004), FLIPSUPERTREE (Griebel et al. 2008), PHYSIC (Ranwez et al. 2007), and PHYSIC_IST (Scornavacca et al. 2008). For the former, we performed Mantel tests between d_A and d_B with the corresponding distance values displayed by *S*. For the latter, we used either the distance matrices d_A and d_B or the equidistant trees T_A and T_B generated from them by LASSO, as appropriate, to construct a supertree using the five algorithms listed above. We then restricted each supertree *S* to each of the data sets *A* and *B*, resulting in the trees $S|_A$ and $S|_B$, respectively. For each supertree algorithm, we then measured the distance between $S|_A$ and T_A and between $S|_B$ and T_B using four different measures: Robinson–Foulds (RF), the triplet distance

(*T*), Simple Scoring (SS), and Adjustable Scoring (AS) (as implemented within the EPoS software).

Supplementary Material

Supplementary material, tables S1–S7, and figures S1–S6 are available at *Molecular Biology and Evolution* online (http://www.mbe.oxfordjournals.org/).

Acknowledgments

The authors gratefully acknowledge Lesley Boyd (NIAB, Cambridge, UK) and Muge Sayar-Turet (Bogazici University, Bebek-Istanbul, Turkey) for providing a wheat data set used within the supertree evaluation, and Andrei-Alin Popescu for his help with the ADMIXTURE analysis and testing of the Lasso software. They thank the referees for their highly constructive comments and suggestions. The NCYC is a BBSRC supported National Capability; I.N.R. and J.D. acknowledge the BBSRC for both NCYC (National Capability grant BBS/E/F/00044440) and IFR (Institute Strategic Programme grant BBS/E/F/00044471) funding.

Appendix

In this section, we first present some relevant definitions and then formally define the notion of a suitable clique. Subsequent to this, we provide a pseudocode version of Lasso and, finally, establish some theoretical properties enjoyed by Lasso.

Preliminaries

Suppose X is a set of at least two taxa and let T denote a (rooted) phylogenetic tree on X. If T' is a further rooted phylogenetic tree on X then, following (Huber and Popescu 2013), we say that T and T' are equivalent if they are isomorphic in the usual graph theoretical sense and the underlying map is the identity on X that maps the root of T to the root of T'. An edge-weighting of T is a map ω that assigns to every edge of T a positive weight. An edge-weighted rooted phylogenetic tree on X is a pair (T,ω) such that T is a rooted phylogenetic tree on X and ω is an edge-weighting for it. For any two taxa x and y, we define the distance $D_{(T,\omega)}(x, y)$ between x and y induced by (T,ω) to be the sum of edge-weights on the path joining x and y.

Let $\mathcal{L}$ denote a set of cords on X. Following Huber and Popescu (2013), we call two equidistant trees (T_1, ω_1) and (T_2, ω_2) on X "$\mathcal{L}$-isometric" if $D_{(T_1,\omega_1)}(x, y) = D_{(T_2,\omega_2)}(x, y)$ holds for all cords $xy \in \mathcal{L}$. Suppose that T is a rooted phylogenetic tree on X and $\mathcal{L}$ is a set of cords on X. Then, we say that $\mathcal{L}$ is a strong lasso for T if for every rooted phylogenetic tree T' and any equidistant edge-weightings ω of T and ω' of T', respectively, we have that T and T' are equivalent and that $\omega = \omega'$ holds whenever (T,ω) and (T', ω') are $\mathcal{L}$-isometric. We denote the set of cords induced by a partial distance matrix D on X by $\mathcal{L}_D$ and denote the graph representing the cords in $\mathcal{L}_D$ by $\Gamma(\mathcal{L}_D)$.

Suitable Cliques

Central to Lasso is finding for a given set X of taxa and a partial distance matrix a suitable clique in the graph $\Gamma(\mathcal{L}_D)^w$, where w is an edge-weighting and D was constructed in the previous iteration (or is the input distance matrix if within the first iteration). To be able to define this special type of clique, assume that m is the minimal edge-weight of the graph $\Gamma(\mathcal{L}_D)^w$ and let C denote a subgraph of $\Gamma(\mathcal{L}_D)$ obtained by first deleting all edges with weight not m (assuming that C is chosen so that it contains an edge with weight m) from $\Gamma(\mathcal{L}_D)^w$ and then ignoring the edge-weights rendering C an unweighted graph.

Exploiting the fact that the vertices of any clique of C can be thought of as equidistant trees whose leaf sets are contained in X or are in fact a taxon of X, we say that a clique in C is suitable in C if, over all cliques of C, the number of taxa of X it contains is as large as possible. Similarly, we say that a clique is suitable in $\Gamma(\mathcal{L}_D)$ if, over all subgraphs of $\Gamma(\mathcal{L}_D)$ obtained as described above, the number of taxa of X it contains is as large as possible.

As the problem of finding such a clique further requires solving the problem of whether a clique of a pregiven size or more exists in a graph, and this is a well known NP-complete problem (Garey and Johnson 1979), we use a heuristic for this. More precisely, we start with a randomly chosen edge e in C. Note that e is clearly a clique. Denoting that clique by C_e, and its vertices by x and y, we check for all remaining vertices z of C if they are adjacent with every vertex of C_e or not. In the former case we update C_e by adding z to its vertex set and all edges of the form $\{c, z\}$ to its edge set where c is a vertex in C_e, and in the latter case we discard z. We continue in this fashion until we cannot enlarge C_e any further, in which case we stop and save the found clique. To mitigate against a poor choice of C, we repeat this process k-times (ignoring edges that are chosen more than once) where k is a parameter that is currently set to 10. The clique that, over all found cliques, has the largest number of leaves is the clique that we take as the suitable clique.

The Lasso Algorithm in Pseudocode Form

Algorithm 1 A pseudocode version of Lasso

Input: Partial distance matrix D on X.
Output: A subset $\mathcal{L}'$ of cords of $\mathcal{L}_D$ and an equidistant tree (T,ω) on $Y = \cup_{xy\in\mathcal{L}'} xy$ that is strongly lassoed by it such that Y and $\mathcal{L}'$ are as large as possible and $D_{(T,\omega)}(x, y) = D(x, y)$ holds for all $xy \in \mathcal{L}'$.

0. Put $X' = X$ and $D' = D$
1. For $m := \min_{xy\in\mathcal{L}_D} D'(x, y)$, delete all edges with weight not m from $\Gamma(\mathcal{L}_{D'})^w$ and ignore all edge-weights to obtain an unweighted graph.
2. Using the above heuristic compute the suitable cliques of the generated graph and collect them in the set $\mathcal{C}$.

3. Choose a suitable clique K in $\mathcal{C}$ that has at least two vertices. If no such clique exists terminate and save the tree(s) and the found set(s) of cords that strongly lasso them, respectively.
4. Join the vertices of K through an edge to a new vertex u to obtain the tree U. Define the equidistant edge-weight ω for U using the edge weightings of the equidistant trees under consideration and their root height (see Materials and Methods for details).
5. Update the set X' by deleting all vertices in K and adding the vertex u. Using the definition of D^* (see Materials and Methods for details), update D' to a new distance matrix on X'.
6. Return to step 1.

We remark that steps (1)–(3) correspond to Construction step (i) and (ii) in the main text and that the definition of D' might further reduce X'. Also, to mitigate against poor choices, we currently run the algorithm ten times for a given input data set and return the equidistant tree with the highest number of leaves (plus its underpinning strong lasso) over all those runs. Finally, for efficiency reasons, we have replaced step 2 in our implementation of Lasso by first randomly choosing a connected component of the graph generated in step 1 and then finding a suitable clique in that component.

Theoretical Properties of Lasso

In this section, we present theoretical properties enjoyed by Lasso. As before, let Y, $\mathcal{L}_Y$, and (T_Y, ω_Y) denote the output of Lasso for a partial distance matrix D on a taxa set X.

Theorem 1

Suppose D is a partial distance matrix on X. Then $\mathcal{L}_Y$ is a strong lasso for T_Y. Furthermore, if there exists an equidistant tree (T,ω) such that $D_{(T,\omega)}(x, y) = D(x, y)$ for all $xy \in \mathcal{L}_D$ and $\mathcal{L}_D$ is a topological lasso for T then $\mathcal{L}_Y$ is a strong lasso for T and the equidistant tree returned by Lasso is (T,ω). In particular, if D is a complete distance matrix on X then the equidistant tree returned by Lasso is (T,ω).

Proof

By construction, for any interior vertex v of T_Y the child-edge graph of v is a clique (Huber and Popescu 2013). Thus, by Huber and Popescu (2013, Theorem 7.1), $\mathcal{L}_Y$ is a topological lasso for T_Y. By Huber and Popescu (2013, Corollary 7.3) it follows that $\mathcal{L}_Y$ is a strong lasso for T_Y.

For the remainder, assume that there exists an equidistant tree (T,ω) such that $D_{(T,\omega)}(x, y) = D(x, y)$ for all $xy \in \mathcal{L}_D$. Assume first $\mathcal{L}_D$ is a topological lasso for T. In view of the remark following the presentation of the pseudocode version of the Lasso algorithm, let S denote a connected component of the graph generated in step 1 of that presentation and let K denote a suitable clique in S. Then, as (T,ω) is an equidistant tree and $D_{(T,\omega)}(x, y) = D(x, y)$ holds for all $x, y \in X$ it follows that every vertex in S is also a vertex in K. Thus, S is itself a suitable clique. Furthermore, as $\mathcal{L}_D$ is a topological lasso for T, $\Gamma(\mathcal{L}_D)$ is connected (Huber and Kettleborough, submitted) and, so, there must exist an edge in $\Gamma(\mathcal{L}_D)$ joining a vertex contained in S with a vertex not contained in S. In combination, this implies that every such edge that joins a vertex in S with the same vertex not contained in S must have the same weight in $\Gamma(\mathcal{L}_D)^w$. Consequently, no vertex is discarded in the computation of the new set X' and no distance value is removed when taking the mode to recompute the distance matrix. This implies that $\mathcal{L}_Y$ is also a topological lasso for T and thus a strong lasso for it. As $D_{(T,\omega)}(x, y) = D(x, y) = D_{(T_Y,\omega_Y)}(x, y)$ holds for all $xy \in \mathcal{L}_Y$, it follows that T and T_Y must be equivalent and $\omega_Y = \omega$. Thus, the equidistant tree returned by Lasso is (T,ω).

Now assume that D is a complete distance matrix on X. Then, $\mathcal{L}_D$ is in particular a strong lasso for T and therefore also a topological lasso for T. By the above, the equidistant tree returned by Lasso is (T,ω).

References

Alexander D, Novembre J, Lange K. 2009. Fast model-based estimation of ancestry in unrelated individuals. *Genome Res.* 19:16551664.

Bininda-Emonds ORP. 2004. Phylogenetic Supertrees: combining information to reveal the tree of life. Dordrecht, The Netherlands: Kluwer Academic Publishers.

Bouckaert R, Heled J, Kühnert D, Vaughan T, Wu C-H, Xie D, Suchard MA, Rambaut A, Drummond AJ. 2014. BEAST 2: a software platform for Bayesian evolutionary analysis. *PLoS Comput Biol.* 10:e1003537.

Brinkmeyer M, Griebel T, Boecker S. 2013. Flipcut supertrees: towards matrix representation accuracy in polynomial time. *Algorithmica* 67: 142–160.

Burbrink F, Castoe T. 2009. Molecular phylogeography of snakes. In: Mullin Stephen J, Seigel Richard A, editors. Snakes: ecology and conservation. Ithaca, NY, United States: Cornell University Press. 38–77.

Cavalli-Sforza LL, Edwards AW. 1967. Phylogenetic analysis. Models and estimation procedures. *Am J Hum Genet.* 19(3 Pt 1):233–257.

Confalonieri V, Sequeira A, Todaro L, Vilardi J. 1998. Mitochondrial DNA and phylogeography of the grasshopper trimerotropis pallidipennis in relation to clinical distribution of chromosome polymorphisms. *Heredity* 81:444–452.

Criscuolo A, Berry V, Douzery E, Gascuel O. 2006. Sdm: a fast distance-based approach for (super)tree building in phylogenomics. *Syst Biol.* 55:740–755.

Criscuolo A, Gascuel O. 2008. Fast NJ-like algorithms to deal with incomplete distance matrices. *BMC Bioinformatics* 9(1):166.

DeSoete G. 1984. Ultrametric tree representations of incomplete dissimilarity data. *J Classif.* 1(1):235–242.

Dress AWM, Huber KT, Steel M. 2011. "Lassoing" a phylogenetic tree I: basic properties, shellings, and covers. *J Math Biol.* 1–29.

Drummond A, Ho S, Phillips M, Rambaut A. 2006. Relaxed phylogenetics and dating with confidence. *PLoS Biol.* 4:e88.

Durbin R, Eddy S, Krogh A, Mitchison G. 1998. Biological sequence analysis: probabilistic models of proteins and nucleic acids. Cambridge (United Kingdom): Cambridge University Press.

Felsenstein J. 2004. PHYLIP (Phylogeny Inference Package) version 3.6. Distributed by the author. Seattle (WA): Department of Genome Sciences, University of Washington.

Garey MR, Johnson DS. 1979. Computers and intractability: a guide to the theory of NP-completeness. New York: W. H. Freeman & Co.

Gascuel O. 1997. BIONJ: an improved version of the NJ algorithm based on a simple model of sequence data. *Mol Biol Evol.* 14:685–695.
Gaul W, Schader M. 1994. Pyramidal classification based on incomplete dissimilarity data. *J Classif* 11(2):171–193.
Griebel T, Brinkmeyer M, Bocker S. 2008. EPoS: a modular software framework for phylogenetic analysis. *Bioinformatics* 24(20):2399–2400.
Guénoche A, Grandcolas S. 1999. Approximations par arbre d'une distance partielle. *Math Inf Sci Hum* 37(146):51–64.
Guénoche A, Leclerc B, Makarenkov V. 2004. On the extension of a partial metric to a tree metric. *Discrete Math* 276(1):229–248.
Hellmuth M, Hernandez-Rosales M, Huber KT, Moulton V, Stadler PF. 2013. Orthology relations, symbolic ultrametrics, and co-graphs. *J Math Biol.* 66:399–420.
Huang H, Knowles L. 2015. Unforseen consequences of excluding missing data from Next-Generation sequences: simulation study of RAD sequences. *Syst Biol.* Advance Access published July 4, 2014, doi:10.1093/sysbio/syu046.
Huber KT, Popescu A-A. 2013. Lassoing and corralling rooted phylogenetic trees. *Bull Math Biol.* 75(3):444–465.
Huber KT, Steel M. 2014. Reconstructing fully-resolved trees from triplet cover distances. *Electron J Comb* 21:P2.15.
Joly S, David B, Lockhart PJ. 2015. Flexible methods for estimating genetic distances from nucleotide data. *Methods Ecol Evol.* Advance Access published February 16, 2015, doi:10.1111/2041-210X.12343.
Kupczok A. 2011. Consequences of different null models on the tree shape bias of supertree methods. *Syst Biol.* 60:218–225.
Lapointe FJ, Levasseur C. 2004. Everything you always wanted to know about the average consensus and more. In: Bininda-Emonds ORP, editor. Phylogenetic Supertrees: combining information to reveal the tree of life. Dordrecht, The Netherlands: Kluwer Academic Publishers. p. 87–105.
Liti G, Carter DM, Moses AM, Warringer J, Parts L, James SA, Davey RP, Roberts IN, Burt A, Koufopanou V, et al. 2009. Population genomics of domestic and wild yeasts. *Nature* 458(7236):337–341.
Makarenkov V. 2001. Une nouvelle mthode efficace pour la reconstruction des arbres additifs partir des matrices de distances incompltes. Proceedings of the 8-imes Rencontres de la Socit Francophone de Classification. Pointe-à-Pitre, Guadeloupe: Universite de Antilles-Guyane. p. 238–244.
Misof B, Meyer B, Reumont VBM, Kück P, Misof K, Meusemann K. 2013. Selecting informative subsets of sparse supermatrics increases the chance to find correct trees. *BMC Bioinformatics* 14:348.
Page R. 2002. Modified mincut supertrees. In: Proceedings of ProdWorkshop on Algorithms in Bioinformatics (WABI '02), Springer Lecture Notes in Computer Science. Vol. 2452. p. 537–552.
Philippe H, Snell EA, Bapteste E, Lopez P, Holland PW, Casane D. 2004. Phylogenomics of eukaryotes: impact of missing data on large alignments. *Mol Biol Evol.* 21(9):1740–1752.
Pritchard J, Stephens M, Donnelly P. 2000. Inference of population structure using multilocus genotype data. *Genetics* 155:945–959.
Queiroz A, Gatesy J. 2006. The supermatrix approach to systematics. *Trends Ecol Evol.* 22:34–41.
Ranwez V, Berry V, Criscuolo A, Fabre P-H, Guillemot S, Scornavacca C, Douzery EJP. 2007. Physic: a veto supertree method with desirable properties. *Syst Biol.* 56(5):798–817.
Reeves JC, Chiapparino E, Donini P, Ganal M, Guiard J, Hamrit S, Heckenberger M, Huang X-Q, VanKaauwen M, Kochieva E, et al.. September 8-11, 2004. Changes over time in the genetic diversity of four major European crops: a report from the GEDIFLUX Framework 5 project. In: Vollmann J, Grausgruber H, Rueckenbauer P, editors. Proceedings of the XVIIth EUCARPIA General Congress; 8–11 September 2004; Vienna, Austria: University of Natural Resources and Applied Life Sciences. p. 3–7.
Reif JC, Melchinger AE, Frisch M. 2005. Genetical and mathematical properties of similarity coefficients applied in plant breeding and seed bank management. *Crop Sci.* 45:1–7.
Robinson D, Foulds LR. 1981. Comparison of phylogenetic trees. *Math Biosci* 53(1):131–147.
Roure B, Baurain D, Philippe H. 2012. Impact of missing data on phylogenies inferred from empirical phylogenomic data sets. *Mol Biol Evol.* 30(1):197–214.
Saitou N, Nei M. 1987. The Neighbor-Joining method: a new method for reconstructing phylogenetic trees. *Mol Biol Evol.* 4(4):406–425.
Sanderson M, McMahon M, Steel M. 2010. Phylogenomics with incomplete taxon coverage: the limits to inference. *BMC Evol Biol.* 10(1):155.
Sayar-Turet M, Dreisigacker S, Braun H-J, Hede A, MacCormack R, Boyd L. 2011. Genetic variation within and between winter wheat genotypes from Turkey, Kazakhstan and Europe as determined by nbs-profiling. *Genome* 54:419–430.
Scornavacca C, Berry V, Lefort V, Douzery E, Ranwez V. 2008. Physic_ist: cleaning source trees to infer more informative supertrees. *BMC Bioinformatics* 9:413.
Semple C, Steel M. 2003. Phylogenetics. Oxford Lecture Series in Mathematics and its Applications. Oxford, United Kingdom: Oxford University Press.
Sokal RR. 1958. A statistical method for evaluating systematic relationships. *Univ Kansas Sci Bull.* 38:1409–1438.
Steel M, Sanderson MJ. 2010. Characterizing phylogenetically decisive taxon coverage. *Appl Math Lett.* 23:82–86.
Tibshirani R. 1996. Regression shrinkage and selection via the lasso. *J R Stat Soc Series B Stat Methodol* 58(1):267–288.
Weir JT, Schluter D. 2008a. Calibrating the avian molecular clock. *Mol Ecol* 17:2321–2328.
Weir JT, Schluter D. 2008b. Ice sheets promote species in boreal birds. *Proc R Soc Lond B Biol Sci.* 271:1881–1997.
West C, James SA, Davey RP, Dicks J, Roberts IN. 2014. Ribosomal DNA sequence heterogeneity reflects intraspecies phylogenies and predicts genome structure in two contrasting yeast species. *Syst Biol.* 63(4):543–554.
Willson SJ. 2004. Constructing rooted supertrees using distances. *Bull Math Biol.* 66:1755–1783.
Xiao Y, Liu W, Dai Y, Fu C, Bian Y. 2010. Using SSRmarkers to evaluate the genetic diversity of *lentinula edodes* natural germplasm in China. *World J Microbiol Biotechnol* 26:527–536.
Zuckerkandl E, Pauling LB. 1962. Molecular disease, evolution, and genic heterogeneity. In: Kasha M, Pullman B, editors. Horizons in biochemistry. New York: Academic Press. p. 189–225.