GENETICS

A forest is more than its trees: haplotypes and ancestral recombination graphs

Halley Fritze, Nathaniel Pope, Jerome Kelleher, and Peter Ralph

Review Timeline:

Submission Date:	2025-02-25
Editorial Decision:	2025-04-15
Resubmission Received:	2025-05-29
Editorial Decision:	2025-06-23
Revision Received:	2025-07-22
Accepted:	2025-08-12

March 31, 2025

GENETICS-2025-307917
A forest is more than its trees: haplotypes and ancestral recombination graphs

Dear Dr. Ralph:

Three experts in the field have reviewed your manuscript, and I have read it as well. I am pleased to inform you that, with minor revisions, it is potentially suitable for publication in GENETICS. The reviewers have comments and concerns that need to be addressed in a revised manuscript. You can read their reviews at the end of this email.

It is most important that you address the following in your resubmission:

All three reviewers believe that your manuscript is an important contribution to the literature, albeit in some cases to a limited audience. But overall I feel it is a strong contribution. Both reviewers 1 and 2 have fairly minor comments and believe the paper is ready for acceptance. However, Reviewer 3 has a number of minor comments and needs clarification and checking of several equations, statements and pseudocode. I don't believe these requests are unreasonable, and just to be on the safe side, I may send it back to reviewer 3 to check for a second review.

We look forward to receiving your revised manuscript. Please let the editorial office know approximately how long you expect to need for revisions.

Upon resubmission, please include:
1. A clean version of your manuscript;
2. A marked version of your manuscript in which you highlight significant revisions carried out in response to the major points raised by the editor/reviewers (track changes is acceptable if preferred);
3. A detailed response to the editor's/reviewers' comments and to the concerns listed above. Please reference line numbers in this response to aid the editors.

Additionally, please ensure that your resubmission is formatted for GENETICS.
https://academic.oup.com/genetics/pages/general-instructions

Follow this link to submit the revised manuscript: Link Not Available

Sincerely,

Scott Edwards
Associate Editor
GENETICS

Approved by:
Maria Chikina
Senior Editor
GENETICS

Reviewer #1 :

In this paper Fritze and colleagues take a haplotype-centric view of the ancestral recombination graph (ARG) and contrast this view with the more common simplification of the ARG as a sequence of marginal trees. In particular, internal nodes in the ARG represent ancestral haplotypes, but in the process of converting an ARG to a succinct tree sequence, some of these internal nodes are deleted (essentially if one of the lineages that descend from the node recombines and coalesces elsewhere, then that node would become unary and is not represented in the succinct tree sequence). As such, a given node in a succinct tree sequence may not span as many marginal trees as it "should", and this paper develops an algorithm to try to infer where these unary nodes should be added, which the authors refer to as extending haplotypes. To assess how well their algorithm can recover trees, the authors develop novel dissimilarity measures on ARGs, and use these on simulated true and inferred ARGs. Overall, the extension method works quite well on true ARGs, but less well on inferred ARGs. I think that this paper is an important contribution to the ARG literature -- this haplotype perspective is important and generally overlooked, and measures

for assessing how well methods can recover ARGs in terms of this perspective will be useful for future method development. Overall the paper is very nice, well-written, interesting, and clear, but I do have a few comments for the authors in the hopes that they're useful. They're all minor, and the authors should feel free to just use what they find helpful.

Minor Comments:

- I really enjoyed the title of the paper!

- I am a bit confused by the definition of weighted Robinson-Foulds distance on the bottom of page 6. I believe that this is more like a Robinson-Foulds _similarity_, as I believe that as defined it should be maximized when the two ARGs are the same, and smaller for more dissimilar ARGs.

- Is the dissimilarity measure defined in equation (3) used anywhere?

- What is "dissimilarity" as mentioned in the caption of Figure 3?

- I found the discussion around the metric version to be hard to follow and worth expanding. For example, I believe that the edges of the ARG are what's getting embedded into [0, L) x N x N, and the symmetric differences are taken in this space in the obvious way (i.e., without this mapping its unclear what the relative difference of two ARGs is). There's also a mention of "dissimilarity" in this paragraph. I assume that that's the "symmetrized version of non-matched span" that was just defined, but it would be good to make that clear.

- The performance on inferred ARGs makes clear that this is a difficult problem, and that this haplotype view might be useful conceptually and for methods development, but that as of now we should probably not take these ancestral haplotypes at face value in inferred ARGs. It would be helpful if the authors could provide some intuition for what's going wrong in this case. Presumably the topologies are not inferred well, and many nodes can be extended, which often results in some true and false positives? Another use case I could see would be using this approach to simplify inferred ARGs to make various downstream computations faster (a la figure 4). As such, I would be curious to see how the extension algorithm affects the number of edges in inferred ARGs.

- This is obviously a matter of taste, but I would have found the paper a bit easier to follow if some of the points discussed in the discussion were brought up earlier in the paper. For example, the "ignorance and omission in the ARG" section answered questions I had when reading the bits around Figure 1 -- upon initially reading the bit around Figure 1 I was confused by the interpretation that the lack of a node implies that that individual is not represented in other parts of the ARG (as any point along any branch in an ARG represents _some_ individual ancestral to sample), and as a result I found the presented interpretation somewhat unnatural. This was then eventually clarified (very eloquently) in the discussion. Similarly, I would have appreciated the connection to parsimony earlier. It was not completely obvious to me that recovering the unary nodes from the succinct tree sequence might not be possible in all cases, and as such that the algorithm is an attempt at inference, and furthermore, that it's optimizing some reasonable objective. Again, this point was addressed very well, but I had to wait until the discussion.

Reviewer #2 :

The authors present interesting results in a clear and well-written manuscript, providing a deeper understanding of Ancestral Recombination Graphs and developing improved inference methods. The key idea is to take into account the additional information provided by haplotype structure which has rarely been considered before. On the way, the authors also develop a haplotype-aware measure of (dis)agreement between ARGs. In particular, the achieved goal is to infer information about the intermediate (unary) nodes in the local trees and to show how accurate the inference is.

I only have a few minor comments.

p.2 l.53 "Notation and terminology": it would be good to clarify at this point that parents and children in the graph correspond to ancestors and descendants in the population.

p.4 l.10 "in practice we do repeated passes in both directions until no changes can be made": to someone who does not have hands-on experience with the implementation of this algorithm, the reason why the algorithm needs to do several passes in both directions might not be very clear. If at each tree transition all pair of mergeable paths are merged, it would be good to explain why, after one pass in each direction, additional mergeable paths are found. Perhaps explaining when/why additional mergeable paths are found/created could also give further insight on how many passes will be needed?

p.8 l.25 "we simulate ARGs containing full haplotypes using msprime": since msprime can simulate ARGs under various models, please specify here which model is used (SMC, SMC'... ?).

In general, it comes natural to wonder whether the algorithms can be corrected to avoid incorrectly added spans (e.g. p.10 l. 5). It would be worth to mention whether an attempt of correction has been made and to explain the difficulties/limitations involved.

p.12 l.9 "Data availability": the link https:// github.com/ XXX/ TODO should be updated.

Reviewer #3 :

The fundamental idea of this paper is important and the worked-out algorithm appears quite valuable. I want to see it published. However the current version is so difficult to read that it is likely to have a very limited audience, and appears to have some errors.

I do not feel I have understood the entire paper despite several hours' work. Some of my comments may be misunderstandings, but if so, the text I misunderstood could probably use clarification.

(1) The description of Algorithm 1 and Algorithm 2 is confusing. Naming them this way implies that they are separate algorithms but in fact "Algorithm 2" is the definition of a function needed by Algorithm 1.

Algorithm 2 is mentioned in an explanation a full page before it is introduced.

Algorithm 1 seems to be named ExtendHaplotypes and Algorithm 2 seems to be named Mergeable: having two names for each one is not helpful.

"Mergeable" is an unfortunate name for a function which does not return a yes or no answer but instead returns the number of new edges to be added. Also the text needs a comment about why infinity is sometimes returned.

In Algorithm 1 the variable m means the number of new edges to be added. In Algorithm 2 that quantity is called m_e and m means the number of extended haplotypes. This is confusing. Please use the same name for a given concept in all pseudocode.

Comments in the pseudocode would make it easier to follow, as could use of more meaningful variable names (I particularly dislike M, m, M' in the same algorithm).

Capital I is an unfortunate subscript as it is almost indistinguishable from digit 1.

(2) Issues in algorithms.

Algorithm 1 appears to have an error leading to an infinite loop. Variable M is initially set to 0. It is compared to variable m, which is a number of new edges to be added and must be positive or zero. Thus, m will never be lower than M, so the action taken is to set M' to the minimum of m and M, which will always be zero. M is then set to M', that is, to zero. Since this is inside a while loop with terminating condition M == infinity, it does not terminate. (I coded it to check.)

The logic problem is probably around line 10. We have just tested whether m < M and this was false, so m >= M. It is therefore strange to take the minimum of m and M, as this will always be M. However I can't work out the correct code.

The text description of Algorithm 1 (ExtendHaplotypes) says that

paths with the fewest new edges are added first. The pseudocode given does not appear to do that. It appears (bearing in mind that it has at least one bug, so I can't be positive) to be willing to execute a Merge before examining all children, so I think it will Merge the first child with a better score than previous children, regardless of the fact that an even better child may remain to be found.

After considerable work I have been unable to code Algorithm 2. When is a node NULL? Does this mean the node doesn't exist (for example, the parent of the root)?

(3) I recommend that the example partial tree in Figure 1 be extended to a full tree (it only needs one more node) and used in the practical examples in the text.

(4) The concrete example under "An algorithm to extend haplotypes" is difficult to visualize. Can it be related to Figure 1? When I try to do this, though, I have difficulty because in Figure 1 the right-hand tree does not *have* v1 ... vn, it jumps straight from p to c. Is this example not mergeable, or is the description of mergeablility missing any indication that some of the nodes it relies on may not exist?

(5) The direction in which time is measured on the tree needs to be stated as unfortunately the literature has both conventions. Also, the text implies at several points that this can be used for trees without time information, but it cannot as times are used in the algorithm.

Following on from this, in Algorithm 1, why can no node in u be at the same time as a node in v? At first I thought we were assuming that no two nodes, other than samples, are ever at the same time (a fairly common assumption) but then we would not need this rule. Is this just a convenience to avoid putting two nodes on the same branch at the same timepoint? What happens if this algorithm is attempted with inferred ARGs that discretize time, as some inference algorithms do, so that ties become common?

(6) "In what follows we think of T_2 as the true ARG and T_1 as an inferred ARG." This is not strongly enough stated. The measure is asymmetrical with regards to T_1 and T_2, and will give wrong answers if they are reversed. It is also not suitable for comparing two ARGs where neither is the true ARG.

In general the section on comparison of true and inferred ARGs is difficult to follow. Figure 3 helps but appears too late in the paper.

(7) In Figure 3 it is unfortunate to use to use an example where the forward and inverse matches are the same. I would also appreciate a brief explanation of why it's 46 and 47 in the denominator--I figured it out but it's less than obvious. Why do sample nodes count, as they cannot vary among trees? Doesn't this just add bookkeeping effort? It's otherwise a very helpful figure though.

(8) I cannot clearly distinguish the three purplish colors in Figure 4.

(9) The interpretation of Figure 5 is completely unclear to me. I think it needs more explanatory text. I have no idea what point is being made by these diagrams.

(10) I remind the authors to replace the script availability placeholder URL with the real thing!

Associate Editor Comments:

Reviewer 1:

In this paper Fritze and colleagues take a haplotype-centric view of the ancestral recombination graph (ARG) and contrast this view with the more common simplification of the ARG as a sequence of marginal trees. In particular, internal nodes in the ARG represent ancestral haplotypes, but in the process of converting an ARG to a succinct tree sequence, some of these internal nodes are deleted (essentially if one of the lineages that descend from the node recombines and coalesces elsewhere, then that node would become unary and is not represented in the succinct tree sequence). As such, a given node in a succinct tree sequence may not span as many marginal trees as it "should", and this paper develops an algorithm to try to infer where these unary nodes should be added, which the authors refer to as extending haplotypes. To assess how well their algorithm can recover trees, the authors develop novel dissimilarity measures on ARGs, and use these on simulated true and inferred ARGs. Overall, the extension method works quite well on true ARGs, but less well on inferred ARGs. I think that this paper is an important contribution to the ARG literature – this haplotype perspective is important and generally overlooked, and measures for assessing how well methods can recover ARGs in terms of this perspective will be useful for future method development. Overall the paper is very nice, well-written, interesting, and clear, but I do have a few comments for the authors in the hopes that they're useful. They're all minor, and the authors should feel free to just use what they find helpful.

(1.1) *I really enjoyed the title of the paper!*

Reply: Thank you!

(1.2) (p. 7, l. 5) *I am a bit confused by the definition of weighted Robinson-Foulds distance at (p. 7, l. 5).I believe that this is more like a Robinson-Foulds* similarity*, as I believe that as defined it should be maximized when the two ARGs are the same, and smaller for more dissimilar ARGs.*

Reply: Oh dear, that was an embarrassing error (that happily doesn't affect anything else in the paper). This has been corrected.

(1.3) *Is the dissimilarity measure defined in equation* (3) *used anywhere?*

Reply: No, the weighted mean square error defined in (3) is not shown for our later experiments shown in Figure 6, however it is always computed when using our implementation is `tscompare`; we now explicitly say this (p. 4, l. 40).

(1.4) *What is "dissimilarity" as mentioned in the caption of Figure 3?*

Reply: "dissimilarity" was our original language to describe the non-matched spans, and we missed this use of the old name. We have now changed the caption in Figure 3 to reflect that change.

(1.5) (p. 7, l. 27) *I found the discussion around the metric version to be hard to follow and worth expanding. For example, I believe that the edges of the ARG are what's getting embedded into* $[0, L) \times N \times N$*, and the symmetric differences are taken in this space in the obvious way (i.e., without this mapping its unclear what the relative difference of two ARGs is). There's also a mention of "dissimilarity" in this paragraph. I assume that that's the "symmetrized version of non-matched span" that was just defined, but it would be good to make that clear.*

Reply: Thanks again to the reviewer for carefully thinking about what we wrote, which on further examination has turned out to be wrong in several ways (apologies!). Please see the updated version that we believe to be correct and thus hopefully less confusing. (p. 7, l. 27)

(1.6) *The performance on inferred ARGs makes clear that this is a difficult problem, and that this haplotype view might be useful conceptually and for methods development, but that as of now we should probably not take these ancestral haplotypes at face value in inferred ARGs. It would be helpful if the authors could provide some intuition for what's going wrong in this case. Presumably the topologies are not inferred well, and many nodes can be extended, which often results in some true and false positives? Another use case I could see would be using this approach to simplify inferred ARGs to make various downstream computations faster (a la Figure 4). As such, I would be curious to see how the extension algorithm affects the number of edges in inferred ARGs.*

Reply: We agree on all counts – in particular, it would be helpful to dig into why haplotypes are being inferred wrongly; however, this more detailed investigation is beyond the scope of this paper (since it's more an investigation into the

inference methods). More generally: if an inferred ARG is incorrect for certain nodes, our method of extend haplotypes exacerbates this issue as extending an incorrectly inferred node extends this incorrect node span even further; we've added a note to this effect (p. 13, l. 24). We've also added Figure S5 that shows numbers of edges and runtime for the inferred ARGs used in our tests, and additional discussion of this point (p. 13, l. 28). The results are interesting, but detailed investigation is outside the scope of this paper – here we are introducing concepts and tools (and, there's plenty to digest); while further investigation could depend strongly on the inference method used.

(1.7) *This is obviously a matter of taste, but I would have found the paper a bit easier to follow if some of the points discussed in the discussion were brought up earlier in the paper. For example, the "ignorance and omission in the ARG" section (p. 13, l. 34) answered questions I had when reading the bits around Figure 1 – upon initially reading the bit around Figure 1 I was confused by the interpretation that the lack of a node implies that that individual is not represented in other parts of the ARG (as any point along any branch in an ARG represents* some *individual ancestral to sample), and as a result I found the presented interpretation somewhat unnatural. This was then eventually clarified (very eloquently) in the discussion. Similarly, I would have appreciated the connection to parsimony earlier. It was not completely obvious to me that recovering the unary nodes from the succinct tree sequence might not be possible in all cases, and as such that the algorithm is an attempt at inference, and furthermore, that it's optimizing some reasonable objective. Again, this point was addressed very well, but I had to wait until the discussion.*

Reply: This is a good suggestion; rather than break up the flow (for less sophisticated readers) we've inserted a pointer to these parts of the discussion, at (p. 2, l. 46).

Reviewer 2:

> The authors present interesting results in a clear and well-written manuscript, providing a deeper understanding of Ancestral Recombination Graphs and developing improved inference methods. The key idea is to take into account the additional information provided by haplotype structure which has rarely been considered before. On the way, the authors also develop a haplotype-aware measure of (dis)agreement between ARGs. In particular, the achieved goal is to infer information about the intermediate (unary) nodes in the local trees and to show how accurate the inference is.

(2.1) (p. 3, l. 2) *"Notation and terminology": it would be good to clarify at this point that parents and children in the graph correspond to ancestors and descendants in the population.*

Reply: Good observation. We have made this change. (p. 3, l. 2)

(2.2) (p. 4, l. 19) *"in practice we do repeated passes in both directions until no changes can be made": to someone who does not have hands-on experience with the implementation of this algorithm, the reason why the algorithm needs to do several passes in both directions might not be very clear. If at each tree transition all pair of mergeable paths are merged, it would be good to explain why, after one pass in each direction, additional mergeable paths are found. Perhaps explaining when/why additional mergeable paths are found/created could also give further insight on how many passes will be needed?*

Reply: Noted. We have written some follow up for that sentence at your suggestion. The creation of new mergable paths after a number of passes is something we thought about quite a bit, however could not come up with a good theoretical conclusion as to if there was a number of required passes before termination. We instead provide some empirical evidence to the number of required passes later in the section and Table S1. (p. 4, l. 19)

(2.3) (p. 9, l. 7) *"we simulate ARGs containing full haplotypes using msprime": since msprime can simulate ARGs under various models, please specify here which model is used (SMC, SMC'... ?).*

Reply: Good suggestion; we've done this (we used "hudson"). (p. 9, l. 7)

(2.4) *In general, it comes natural to wonder whether the algorithms can be corrected to avoid incorrectly added spans (e.g. (p. 10, l. 11)). It would be worth to mention whether an attempt of correction has been made and to explain the difficulties/limitations involved.*

Reply: Good suggestion; we've added a note about this. (p. 10, l. 20)

(2.5) (p. 13, l. 2) *"Data availability": the link https:// github.com/ XXX/ TODO should be updated.*

Reply: Now updated. (p. 13, l. 2)

Reviewer 3:

> The fundamental idea of this paper is important and the worked-out algorithm appears quite valuable. I want to see it published. However the current version is so difficult to read that it is likely to have a very limited audience, and appears to have some errors.

We're glad you think it's important, and apologies for the errors.

> I do not feel I have understood the entire paper despite several hours' work. Some of my comments may be misunderstandings, but if so, the text I misunderstood could probably use clarification.

As you'll see, we've tried to improve clarity throughout; hopefully it is more digestable now.

(3.1) *The description of Algorithm 1 and Algorithm 2 is confusing. Naming them this way implies that they are separate algorithms but in fact "Algorithm 2" is the definition of a function needed by Algorithm 1.*

Reply: We think it's reasonably standard for one algorithm to make use of another (for instance, many algorithms apply a `sort` at some point). However, it's a good point that referring abstractly to "Algorithm 1" and "Algorithm 2" is a little abstract and hard to remember. So, we've gone back to places where the text just said "Algorithm 1", and where we thought helpful, changed it to something like "extending haplotypes with Algorithm 1". For more background: originally we had both Algorithm 1 and Algorithm 2 within one algorithm environment. However, the `Mergeable` is a conceptually separate operation that by separating it out makes understanding `ExtendHaplotypes` easier. Algorithm 1 is the first algorithm as it is the main construction and the most important of the two.

(3.2) *Algorithm 2 is mentioned in an explanation a full page before it is introduced.*

Reply: This is a consequence of the typesetting: Algorithm 2 is in a floating environment, and the earliest the environment will appear is the next page following the previous figure. Rest assured that Genetics will typeset this appropriately.

(3.3) *Algorithm 1 seems to be named ExtendHaplotypes and Algorithm 2 seems to be named Mergeable: having two names for each one is not helpful.*

Reply: This point overlaps with point 3.1 – see our response above for how we've hopefully improved the situation. We like to think of ExtendHaplotypes and Mergable as the names of the functions contained in Algorithm 1 and Algorithm 2, respectively.

(3.4) *"Mergeable" is an unfortunate name for a function which does not return a yes or no answer but instead returns the number of new edges to be added. Also the text needs a comment about why infinity is sometimes returned.*

Reply: A reasonable point; we've changed the name to `MergeNum`. For clarity, we have also included a note about returning infinity in the description of Algorithm 2.

(3.5) *In Algorithm 1 the variable m means the number of new edges to be added. In Algorithm 2 that quantity is called m_e and m means the number of extended haplotypes. This is confusing. Please use the same name for a given concept in all pseudocode.*

Reply: Good point, this change has been made.

(3.6) *Comments in the pseudocode would make it easier to follow, as could use of more meaningful variable names (I particularly dislike M, m, M' in the same algorithm).*

Reply: Good point. We have included some comments for clarity. However, we're going to respectfully differ on stylistic choice of concise variable names.

(3.7) *Capital I is an unfortunate subscript as it is almost indistinguishable from digit 1.*

Reply: While we agree in general, within the context of our algorithm we are interested paths of inheritance between the edges removed from a local tree (the out-forest) and the edges just added into the next local tree (the in-forest). Because of our language use of out and in, we feel like our choice of O and I is reasonable.

Issues in algorithms.

(3.8) *Algorithm 1 appears to have an error leading to an infinite loop. Variable M is initially set to 0. It is compared to variable m, which is a number of new edges to be added and must be positive or zero. Thus, m will never be lower than M, so the action taken is to set M' to the minimum of m and M, which will always be zero. M is then set to M', that is, to zero. Since this is inside a while loop with terminating condition M == infinity, it does not terminate. (I coded it to check.)*

The logic problem is probably around line 10. We have just tested whether $m < M$ and this was false, so $m \geq M$. It is therefore strange to take the minimum of m and M, as this will always be M. However I can't work out the correct code.

Reply: Good catch – and, many apologies – that is indeed a bug; the correction changes line 7 to "if $m \leq M$" and line 10 to "Set $M' = \min(m_e, M')$.". (To be clear, the bug was in the paper, not the implemented code; if the reviewer is interested they can see our implementation in the tskit library at c/tskit/trees.c:9078 in the current release.)

(3.9) *The text description of Algorithm 1 (ExtendHaplotypes) says that paths with the fewest new edges are added first. The pseudocode given does not appear to do that. It appears (bearing in mind that it has at least one bug, so I can't be positive) to be willing to execute a Merge before examining all children, so I think it will Merge the first child with a better score than previous children, regardless of the fact that an even better child may remain to be found.*

Reply: We apologize again for the bug in the previous point that caused this misunderstanding. Fixed, it should perform as advertised (and, we've added a bit clarifying why it works to the description of Algorithm 1).

(3.10) *After considerable work I have been unable to code Algorithm 2. When is a node NULL? Does this mean the node doesn't exist (for example, the parent of the root)?*

Reply: A node is NULL when it has no parent; this is now explained in the description of the algorithm. Apologies, this is part of the conventions in tskit.

(3.11) *I recommend that the example partial tree in Figure 1 be extended to a full tree (it only needs one more node) and used in the practical examples in the text.*

Reply: Good idea, we now reference the figure in the text, at (p. 3, l. 17) and (p. 4, l. 1). On further consideration we think that adding the rest of the tree here would make the figure too confusing (not so much for the trees in A&C, but more so in the haplotype diagrams B&D).

(3.12) *The concrete example under "An algorithm to extend haplotypes" is difficult to visualize. Can it be related to Figure 1? When I try to do this, though, I have difficulty because in Figure 1 the right-hand tree does not* have *$v_1 \ldots v_n$, it jumps straight from p to c. Is this example not mergeable, or is the description of mergeablility missing any indication that some of the nodes it relies on may not exist?*

Reply: Good point. We have connected our concrete construction in that section to reference Figure 1 to improve clarity. (p. 3, l. 17)

(3.13) *The direction in which time is measured on the tree needs to be stated as unfortunately the literature has both conventions. Also, the text implies at several points that this can be used for trees without time information, but it cannot as times are used in the algorithm.*

Reply: We state how time is measured here (p. 2, l. 53); we've added a reminder at (p. 4, l. 7). We do use time in our algorithm, but mainly as a way to break ties, as is the case with Equation 1. This tie break will default to the first node if the ARG does not contain time information. We've also added a note about undated ARGs (p. 4, l. 12).

(3.14) *Following on from this, in Algorithm 1, why can no node in u be at the same time as a node in v? At first I thought we were assuming that no two nodes, other than samples, are ever at the same time (a fairly common assumption) but then we would not need this rule. Is this just a convenience to avoid putting two nodes on the same branch at the same timepoint? What happens if this algorithm is attempted with inferred ARGs that discretize time, as some inference algorithms do, so that ties become common?*

Reply: This is exactly right: we don't assume that node times are unique, as it's common (e.g., from discrete-time simulators) for this not to be true; so the check is to avoid producing an illegal situation (parent and child at the same time). We now explain this at (p. 4, l. 12).

(3.15) (p. 5, l. 9) *"In what follows we think of T_2 as the true ARG and T_1 as an inferred ARG." This is not strongly enough stated. The measure is asymmetrical with regards to T_1 and T_2, and will give wrong answers if they are reversed. It is also not suitable for comparing two ARGs where neither is the true ARG.*

Reply: We've added extra emphasis to the point that the quantities are not symmetric (p. 6, l. 3); furthermore the additional discussion in the "Metrics on ARGs" section should help reinforce this point. As for reversing $\mathbb{T}_1$ and $\mathbb{T}_2$, we think that while $\overrightarrow{\text{match}}(\mathbb{T}_2, \mathbb{T}_1)$ might not be the most useful quantity, it can still give some valuable information between inferred $\mathbb{T}_1$ and true $\mathbb{T}_2$. We discuss this when comparing inverse matched spans between $(\mathbb{T}_1, \mathbb{T}_2)$ and matched spans between $(\mathbb{T}_2, \mathbb{T}_1)$ (p. 6, l. 27).

(3.16) *In general the section on comparison of true and inferred ARGs is difficult to follow. Figure 3 helps but appears too late in the paper.*

Reply: We have expanded our example to hopefully give more clarity (see additional text in caption). The first reference to Figure 3 occurs at the end of the second paragraph in the section on comparison; it would be hard to include the example earlier (and, we're not sure how helpful it will be until ARF and TPR are defined).

(3.17) *In Figure 3 it is unfortunate to use to use an example where the forward and inverse matches are the same. I would also appreciate a brief explanation of why it's 46 and 47 in the denominator–I figured it out but it's less than obvious. Why do sample nodes count, as they cannot vary among trees? Doesn't this just add bookkeeping effort? It's otherwise a very helpful figure though.*

Reply: We sympathize with the reviewer that working through even a simple example like this is tricky and confusing (but, we also find it very helpful). Good point about the same forward and inverse matches. Note that we did construct this example so that matching for node 4 requires a max to be taken over shared spans of its two possible matches 4 and 9. To show some variety, we have added computation of $\overrightarrow{\text{match}}(\mathbb{T}_2, \mathbb{T}_1)$ to show some additional variety and hopefully give more clarity to the methods themselves (see caption). The samples are included because in practice they do *not* necessarily always match: consider a case where some of the samples are ancestors of others (ancient samples; trios; etcetera).

(3.18) *I cannot clearly distinguish the three purplish colors in Figure 4.*

Reply: Thank you for this feedback. We have changed Figure 4 and the subsequent supplementary figures (Figure S2, Figure S1, Figure S3) to have more color variety.

(3.19) *The interpretation of Figure 5 is completely unclear to me. I think it needs more explanatory text. I have no idea what point is being made by these diagrams.*

Reply: Thanks for the feedback. We have added substantial additional explanation at (p. 10, l. 14) and the caption of Figure 5.

(3.20) *I remind the authors to replace the script availability placeholder URL with the real thing!*

Reply: Noted and now changed, thank you!(p. 13, l. 2)

June 23, 2025
RE: GENETICS-2025-308236

Dear Dr. Ralph:

I am pleased to accept your manuscript titled "A forest is more than its trees: haplotypes and ancestral recombination graphs" for publication in GENETICS, pending minor revision.

Please submit your revision along with a brief description of how you modified the manuscript in response to the reviewers' concerns and suggestions (which can be viewed at the bottom of this email. Most important are the few minor revisions suggested by both reviewers but especially by reviewer 2. These focus on some minor but important issues regarding the presentation and description of the methodology. I expect you should be able to submit a revised manuscript within 30 days. A suitably revised manuscript will be acceptable for publication; I don't expect to send it out for review.

When revising the ms., please make an effort to shorten it, because that almost always improves a manuscript. We urge authors to heed the advice of Strunk and White: "omit needless words"1. Follow this link to submit the revised manuscript: Link Not Available

Thank you for submitting this story to Genetics.

Sincerely,

Scott Edwards
Associate Editor
GENETICS

Approved by:
Maria Chikina
Senior Editor
GENETICS

Reviewer comments:
Reviewer #1 :

The authors have more than sufficiently addressed all of my comments from the previous round of reviews. I congratulate them on the interesting and well-written paper.

One minor typo has arisen in the revision:
p.6 line 8 (track changes version): "gets an different set of quantities" --> maybe "gets a different set of quantities"?

Reviewer #3 :

The clarity of the paper is much improved and the pseudocode now appears to be correct (though to this reviewer, Algorithm 1 is a bit of black magic--I had to code it to determine that it performs as advertised!)

I have only minor corrections.

p. 3 line 21 "the times of the nodes {u_i} and {v_i} are unique"

Could we please establish before this that parent and child cannot be at the same time? I know this is assumed by tskit but it's not a universal assumption, and

without this assumption the line above is really ambiguous.

I also suggest rephrasing as "no node in {u_i} has the same time as a node in {v_i}."

p. 3 and 4

P. 3 around line 21 says we are extending nodes {u_i} to T_k+1. This sounds like it modifies T_k+1 by adding unary nodes to it. Around line 25 says that we are modifying T_k. p. 4 says we are modifying both T_k and T_k+1. Please clarify.

p. 7 "we compute the ARF and TPR of in a simple example" typo

p. 9 "here we require" tripped me up: does it refer to the method used throughout the paper, or the alternative proposed here? Becomes clear a few lines later but could be clarified here.

p. 9 lines 54-55. You simulated "ARGs modified from a 'true' ARG"--modified how? The text doesn't seem to say at all.

The results on simulated data could use a bit of roadmapping: a lot of different data sets are mentioned with no clear organization or goal.

Associate Editor comments:

Reviewer 1:

> The authors have more than sufficiently addressed all of my comments from the previous round of reviews. I congratulate them on the interesting and well-written paper.

We thank you very much for your effort in reviewing our paper!

(1.1) (p. 6, l. 4) *One minor typo has arisen in the revision: "gets an different set of quantities" –> maybe "gets a different set of quantities"?*

Reply: Changed, thank you for pointing out this typo.

Reviewer 3:

> The clarity of the paper is much improved and the pseudocode now appears to be correct (though to this reviewer, Algorithm 1 is a bit of black magic–I had to code it to determine that it performs as advertised!)
>
> I have only minor corrections.

Thank you for your dedicated work on review. We appreciate your input.

(3.1) (p. 3, l. 21) *"the times of the nodes u_i and v_i are unique" Could we please establish before this that parent and child cannot be at the same time? I know this is assumed by* `tskit` *but it's not a universal assumption, and without this assumption the line above is really ambiguous.*

I also suggest rephrasing as "no node in u_i has the same time as a node in v_i."

Reply: Thank you for your comment. We have added the assumption of the parent node and child node having unique times in the previous section where the notation for parent and child nodes are introduced in the previous section, (p. 3, l. 2). We have additionally included your re-phrasing to describe the uniqueness between the sets $\{u_i\}$ and $\{v_j\}$ (p. 3, l. 21).

(3.2) (p. 3, l. 23) *Around [(p. 3, l. 23) it] says we are extending nodes u_i to T_{k+1}. This sounds like it modifies T_{k+1} by adding unary nodes to it. Around [(p. 3, l. 25) it] says that we are modifying T_k. [At (p. 4, l. 29) it] says we are modifying both T_k and T_{k+1}. Please clarify.*

Reply: Thank you for bringing this to our attention. The algorithm only extends edges "into" the next tree; so keeps T_k fixed and changes T_{k+1} (which will then be the "next" T_k). We've made corrections to (p. 4, l. 29) and (p. 3, l. 25).

(3.3) (p. 7, l. 29) *"we compute the ARF and TPR of in a simple example" typo*

Reply: Thanks for pointing out this typo! It is now fixed.

(3.4) (p. 8, l. 18) *"here we require" tripped me up: does it refer to the method used throughout the paper, or the alternative proposed here? Becomes clear a few lines later but could be clarified here.*

Reply: Good point, we have added some extra nouns to help clarify that "here" meant the metric defined above.

(3.5) (p. 9, l. 39) *You simulated "ARGs modified from a 'true' ARG"–modified how? The text doesn't seem to say at all.*

Reply: Thanks for the comment. We perform the modifications explained in the Results subsections *Accuracy with true trees* and *Inferred ARGs*. We refer to Figure 6 now so it's more clear what we're referring to.

(3.6) *The results on simulated data could use a bit of roadmapping: a lot of different data sets are mentioned with no clear organization or goal.*

Reply: Good idea; we've added a roadmap (p. 9, l. 42).

July 29, 2025
RE: GENETICS-2025-308236R1

Dr. Peter Ralph
University of Oregon
Data Science
Fenton Hall, University of Oregon
Eugene, Oregon 97405

Dear Dr. Ralph:

Congratulations, your manuscript titled "A forest is more than its trees: haplotypes and ancestral recombination graphs" is accepted for publication in GENETICS! Many thanks for submitting your research to the journal.

To Proceed to Publication:
1. Format your article according to GENETICS style: https://academic.oup.com/genetics/pages/author-guidelines

2. Ensure that you comply with data and community resource citation guidelines:
https://academic.oup.com/genetics/pages/author-guidelines#section-5-9-2

3. Upload your final files at https://genetics.msubmit.net

4. Add oupsupport@scipris.com and genetics.oup@novatechset.com (or the domains @scipris.com and @novatechset.com) to your email program's "safe senders" list. You will be contacted by both at various points during the production process.

Notes:
- Your currently-accepted manuscript (unedited, as submitted, reviewed, and accepted) will be published at GENETICS and deposited into PubMed as an Advance Access article. Notify sourcefiles@thegsajournals.org before signing your license if you do not wish to publish your article via Advance Access.

- We invite you to submit an original color figure related to your paper for consideration as cover art. Please email your submission to the editorial office or upload it with your final files. You can submit a small-sized image for evaluation, and if selected, the final image must be a TIFF file 2513px wide by 3263px high (8.375 by 10.875 inches; resolution of 600ppi). Please avoid graphs and small type.

- After files are sent to Oxford University Press we use SciPris to manage article licensing and payment. If you do not have a SciPris account, you will receive an email from no-reply@scipris.com to sign up to use Oxford University Press' author portal. After logging in, follow the online instructions to sign your license and arrange any payment due.

If you have any questions or encounter any problems while uploading your accepted manuscript files, please email the editorial office at sourcefiles@thegsajournals.org.

Sincerely,

Scott Edwards
Associate Editor
GENETICS

Approved by:
Maria Chikina
Senior Editor
GENETICS