HexSplit: An Improved Computational Approach for Detecting Partial Gene Transfer

Shreya Seshadri[1,3], Sumaira Zaman[1,4], and Mukul S. Bansal[1,2]

[1] School of Computing, University of Connecticut, Storrs, CT 06269, USA
[2] Institute for Systems Genomics, University of Connecticut, Storrs, CT 06269, USA
mukul.bansal@uconn.edu

Abstract. Horizontal transfer of genetic material plays a critical role in microbial evolution. Such horizontal transfer events can result in the transfer of multiple complete genes, single genes, or partial gene fragments. However, current approaches for microbial gene tree reconstruction often overlook partial gene transfer (PGT), leading to errors in reconstructed gene trees and affecting downstream evolutionary analyses.

In this work, we address the problem of reliably identifying whether a given gene family has been affected by sufficient PGT to impact gene tree reconstruction. Our work builds upon an existing method, called *trippd*, which uses sequence alignment tri-partitions to detect the presence of phylogenetically disruptive PGT within gene families. Our improved method, *HexSplit*, extends the trippd algorithm to identify additional PGT cases that may not be detected by trippd. Our analysis using simulated data, accounting for various evolutionary parameters, reveals that the proposed extension improves the sensitivity of the algorithm (seen through decreases in the false negative rate of PGT detection) without significantly increasing false positive detection. We also apply HexSplit to two real biological datasets and demonstrate how it can be used in biological analyses. HexSplit is open-source and freely available from `https://github.com/shreyaaseshadri/hexsplit`.

Keywords: Partial Gene Transfer Detection · Gene Tree Reconstruction · Microbial Evolution

1 Introduction

Horizontal transfer of genetic material plays a very important role in microbial evolution and adaptation. While most studies of horizontal transfer implicitly assume that such transfer occurs at the level of single complete genes, it is well known that horizontal transfers can occur at different scales, including simultaneous transfer of multiple genes [3, 7, 10, 14] and of partial gene fragments [1, 4, 5, 8, 24, 25]. Partial gene transfer (PGT), in particular, remains underappreciated and is often overlooked even though it occurs frequently. For example, it has been estimated that 10 to 20% of genes in microbial genomes may be affected by PGT [25], and that PGT can affect over 30% of

[3] Currently elsewhere. Work performed while author was an undergraduate student.
[4] Currently at Lawrence Berkeley National Lab, California, USA.

prokaryotic gene families when the taxa under consideration are closely related [23]. Furthermore, previous experimental studies have shown that PGT can significantly impact gene tree reconstruction [16, 20, 23]. Yet, most biological studies that rely on gene tree analysis ignore the possible impact of PGT.

In the context of gene tree reconstruction, a key problem of practical importance is that of reliably detecting the presence of *phylogenetically disruptive* PGT in a given gene family sequence alignment, i.e., identifying gene families that have a sufficiently high level of PGT to negatively affect gene tree reconstruction. A large number of computational methods have previously been developed for the problem of identifying recombination breakpoints in sequence alignments, see for example the many methods discussed in [9, 15] as well as the many methods implemented in the popular RDP software package [12, 13]. Many of these methods can, in principle, be used to detect the presence of PGT in gene family alignments. However, since nearly all these methods focus on breakpoint detection, they are designed to be highly sensitive to potential breakpoints and therefore show high false-positive rates of PGT detection. For example, when we tested two of the most sophisticated methods PhyML-Multi [2] and GARD [11] using simulated gene family sequence alignments without any PGT, we found that they showed false-positive PGT detection rates of 65% [23] and 50% (unpublished result), respectively. Another sophisticated method that is designed for PGT detection is T-Rex [1], which uses a sliding window technique to construct window trees along a sequence alignment and compares them to given a species tree to infer possible transfer events. However, that method assumes that all phylogenetic discordance is due to replacing transfer/homologous recombination and thus cannot be applied to gene families with a history of gene duplication or additive transfer. In addition to the above limitations, existing methods do not focus specifically on the problem of identifying *phylogenetically disruptive* PGT. Identifying only phylogenetically disruptive PGT is extremely important in the context of gene tree reconstruction since any gene families flagged as having PGT must either be discarded or be analyzed separately using manual curation or more sophisticated computational approaches.

Recently, a new method called trippd [23] was proposed to reliably detect phylogentically disruptive PGT. Specifically, trippd seeks to identify gene families with sufficient PGT to impact gene tree reconstruction and does so using a simple and efficient approach based on analyzing sequence alignment tri-partitions. Experimental evaluation of trippd showed that it has a very low false-positive rate of PGT detection, while still maintaining low false-negative rates comparable to PhyML-Multi. However, the accuracy of trippd was shown to degrade significantly when PGT occurs in the interior of the gene family sequence alignment (i.e., not occurring at the ends of the sequence alignment) and also when the gene family is affected by multiple distinct PGT events [23].

In this work, we build upon the trippd method to address the above limitations and further improve overall accuracy and robustness. Specifically, we (i) introduce a new method called *HexSplit* that extends the trippd method and makes it far more robust to the location of PGT events in the sequence alignment and to the presence of multiple PGTs, (ii) thoroughly evaluate both trippd and HexSplit on a wide variety of simulated datasets representing several different evolutionary scenarios, including new datasets representing evolutionary conditions on which trippd had not pre-

viously been evaluated, and (iii) demonstrate the impact of HexSplit in practice by applying it to two biological microbial gene family datasets. Our experiments show that, across nearly all tested conditions, HexSplit provides a substantial reduction in false-negative detection rate with only a slight increase in the false-positive detection rate, yielding a clear improvement in overall PGT detection accuracy (measured using F_1 score) over trippd. Finally, since the available prototype implementation of trippd was difficult to use and only provided as a series of separate scripts, we now also provide an easy-to-use push-button implementation that allows users to easily run both HexSplit and trippd. This open-source implementation is freely available from: `https://github.com/shreyaaseshadri/hexsplit`.

2 Description of HexSplit

We first describe the key ideas used in trippd and then describe HexSplit.

Trippd uses a fast and efficient alignment-based test for detecting the presence of significant partial gene transfer while avoiding explicit breakpoint inference. It splits the given multiple sequence alignment into three equal windows, builds a maximum likelihood tree for each window, and identifies the pair of windows whose trees are most similar and the pair whose trees are most dissimilar. It then generates bootstrap replicate trees for each of the three windows and constructs distributions of Robinson–Foulds (RF) distances [19] between these replicates for the most similar pair of windows and the most dissimilar pair of windows. A distribution overlap test is then used to compare the two resulting distributions to decide if the observed tree-discordance could be explained by phylogenetic reconstruction uncertainty/error or if the results are indicative of partial gene transfer. Importantly, trippd does not require prior knowledge of the number or locations of transferred gene fragments and is designed to identify gene families with PGTs large enough to affect gene tree inference.

Trippd was shown to have a very low false-positive rate of PGT detection while maintaining sensitivity comparable to more complex breakpoint detection methods such as PhyML-Multi [23]. However, it was also observed that trippd struggles to identify PGT cases accurately when the PGT region is located towards the interior of the sequence alignment and shows lower accuracy when the input alignment is affected by multiple PGTs. HexSplit builds upon trippd's simple yet effective approach to address these deficiencies.

Key idea behind HexSplit. The key methodological improvement of HexSplit is that it considers not just the set of three windows (i.e., alignment tri-partition) used by trippd, but also an additional set of three windows offset by one-sixth of the alignment length. This is shown in Figure 1. This modification is based on the observation that the PGT detection accuracy of trippd is negatively impacted when, either due to the location of the PGT region within the sequence alignment or due to the presence of multiple PGTs, the distance between the most similar and most dissimilar pairs of window trees is reduced. By using an additional set of three offset windows, we increase our chances of finding pairs of windows with clearly divergent evolutionary histories irrespective of the location of PGT within the alignment. At the same time, this approach ensures that we do not significantly increase the false-positive detection rate. Thus, HexSplit simply

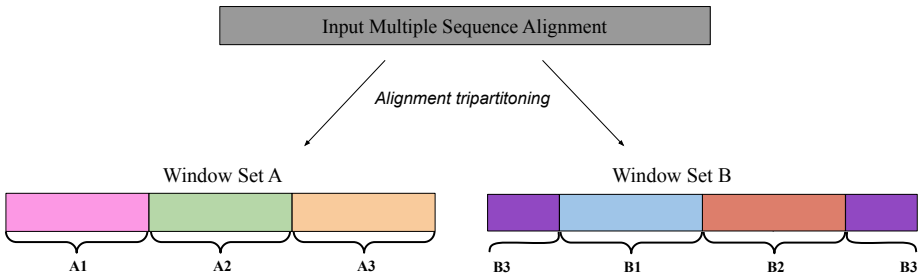

Fig. 1: **Alignment tri-partitions used by HexSplit.** Window Set A consists of three equal-size windows labeled A1, A2, and A3. Window Set B consists of three equal-size windows labeled B1, B2, and B3 offset by one-sixth of the alignment length.

executes trippd on each of these two sets of tri-partitions; if either of these two executions detect PGT then we conclude that phylogenetically disruptive PGT is present, and if neither of the two executions detect PGT then we conclude that phylogenetically disruptive PGT is not present. As we show later, this relatively simple modification results in a marked improvement in PGT detection accuracy across nearly all tested evolutionary conditions.

Methodological details. A detailed step-by-step description of HexSplit appears below. A graphical overview of key steps is shown in Figure 2.

1. **Alignment tri-partitioning.** The given gene family alignment is partitioned into 2 sets containing three windows each. The first set of windows, referred to as Set A, splits the alignment into three equal (or roughly equal) parts in the same way as trippd. The second set of windows, referred to as Set B, also split the alignment into three equal parts but are offset by one-sixth of the alignment length. This is depicted in Figure 1.
2. For each of the two window sets A and B, independently do the following:
   (a) **ML window tree reconstruction**. Use RAxML [21] to compute a maximum likelihood tree for each of the three windows.
   (b) **Identifying most similar and most dissimilar pairs of windows.** Compute the RF distance between all three pairs of ML window trees. Identify the pairs with smallest RF distance, denoted ww_{min}, and largest RF distance, denoted ww_{max}. Note that if $ww_{min} = ww_{max}$ then PGT is assumed to be absent and the following steps are not executed for this set of windows.
   (c) **Bootstrap replicates for each window.** Compute several (100 by default) bootstrap replicates for each of the three windows using RAxML. Denote these as $\{w_i^1, \ldots, w_i^b\}$ for window-i, where $1 \leq i \leq 3$ and b denotes the number of bootstrap replicates per window.
   (d) **Computing distributions of RF-scores.** Given the bootstrap replicates for any two windows i and j, define $D(i, j)$, to be the distribution of RF distances $RF(w_k^i, w_k^j)$, where $k \in \{1, ..., b\}$. Compute $D(ww_{min})$ and $D(ww_{max})$.

(e) **Distribution overlap test.** Apply a simple test to determine if differences between $D(ww_{min})$ and $D(ww_{max})$ are likely due to PGT or not. Specifically, compute the overlap between the distributions $D(ww_{min})$ and $D(ww_{max})$ and check if the percentage of overlap is more than a certain threshold. An overlap smaller than or equal to the threshold percentage indicates presence of PGT. Overlap greater than the threshold indicates lack of PGT.

3. **Final determination of PGT presence or absence.** If either of the two sets of windows trigger PGT detection, report that phylogenetically disruptive PGT is present. Otherwise, report that phylogenetically disruptive PGT is absent.

Threshold selection for distribution overlap test: trippd uses a default threshold of 50% for the distribution overlap test [23]. This threshold was found to generally work well across a wide range of evolutionary scenarios. In HexSplit, we use the same 50% threshold as default.

Computational efficiency: HexSplit essentially runs trippd twice, once on each of the two window sets. Thus, HexSplit requires only twice as much time as trippd to analyze a given sequence alignment. trippd itself is highly efficient. It's most computationally intensive step is the computation of RAxML ML trees and bootstrap replicates. Since each of the three windows includes only one-thirds of the columns in the sequence alignment, trippd's computational requirements are equivalent to executing RAxML once, with bootstrap replicates, on the full input alignment.

3 Description of Datasets

We evaluated the effectiveness and accuracy of HexSplit using both simulated and real biological datasets. Simulated datasets have known ground truths and can therefore be used to assess the PGT detection accuracy (false positive and false negative detection rates) across datasets representing various evolutionary scenarios. The biological datasets were used to assess the potential real-world impact of using HexSplit on real gene family datasets.

3.1 Simulated datasets

We used the simulated datasets used previously to benchmark the performance of trippd [23]. We further augmented these existing datasets with additional simulated datasets to assess robustness across additional evolutionary parameters not considered in that original study. In all, 17 simulated datasets were used. Next, we describe these datasets.

Existing simulated datasets. These datasets consist of 11 collections of gene family alignments, with each collection representing a different evolutionary scenario. Each of these 11 datasets consists of 100 individual gene family alignments simulated using the same evolutionary parameters. Briefly, each gene family alignment consists of sequence alignment columns for sequences evolving vertically down a gene tree, appended to sequence alignment columns for sequences that evolve down the same gene tree but are affected by one or more horizontal transfer events. All alignments are 1000bp in length (which is roughly the average size of microbial genes) and the average number

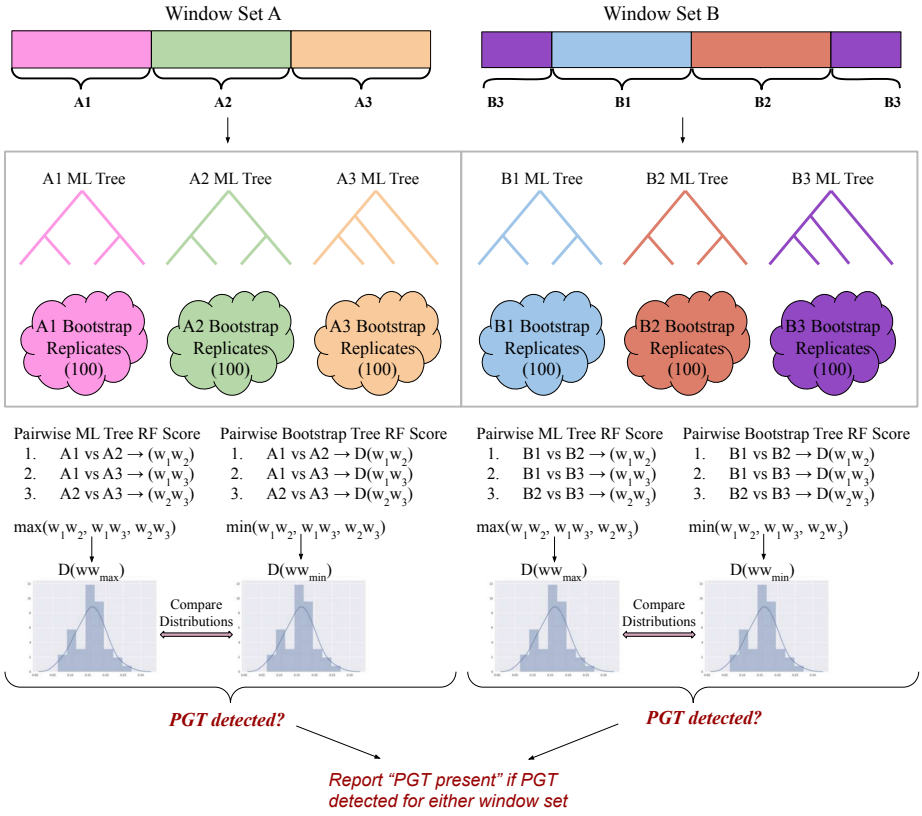

Fig. 2: **Overview of HexSplit.** Key steps of the HexSplit PGT detection approach are shown, ordered from top to bottom. These include the creation of two separate tripartitions (Set A and Set B) of the input alignment, construction of a maximum likelihood tree and 100 bootstrap replicates for each of the six windows, determination of the most similar and most dissimilar pairs of windows for each of the two sets, computation of the distribution of Robinson-Foulds distances between the bootstrap replicate trees for the identified window pairs, application of the distribution overlap test to determine if PGT is present for Set A and for Set B, and using those results to make a final determination about the presence or absence of PGT in the input sequence alignment.

of sequences in an alignment is 98.2. Within each of these 11 datasets, 75 sequence alignments are affected by PGT (i.e., their PGT regions were each affected by at least one horizontal transfer event) and 25 are not affected by PGT (i.e., no horizontal transfer events in their PGT regions), allowing for the inference of both false-positive and false-negative detection rates. Further details on how the datasets were generated appear in [23]. These 11 datasets can be divided into five categories:

1. **Default dataset.** This dataset consists of 100 gene family alignments with the following characteristics: The alignment consists of a single PGT region (consisting of

contiguous columns of the alignment possibly affected by PGT) which spans 40% (i.e., 400bp) of the total alignment length and is located at the beginning of the sequence alignment. Both the PGT and non-PGT regions of the alignment evolve at the same mutation rate of 0.5 substitutions per site per unit branch length. As with all datasets, 75 of the 100 gene family alignments are affected by PGT events and the remaining 25 are not. The datasets below are derived by varying key parameters one at a time.

2. **PGT location datasets.** This category consists of three datasets with alternative locations of the PGT region within the alignment, corresponding to PGT region offsets of 34, 84, and 134 bp compared to the default dataset. We note that larger offsets are unnecessary given the tripartition-based approach used by trippd.
3. **Substitution rate datasets.** This category includes two datasets with alternative rates of sequence evolution corresponding to 0.1 and 5 substitutions per site per unit branch length.
4. **Multi-PGT dataset.** One dataset with two separate PGT regions within each alignment. These PGT regions span 40% and 20% of the total gene family alignment length, with the 20% region appended to the 40% region.
5. **PGT region length datasets.** This category has four datasets corresponding to alternative PGT region lengths of 20%, 30%, 50%, and 60% of the total gene family alignment length.

Additional simulated datasets. In general, PGT and non-PGT regions can (and likely do) evolve at different rates. Such rate variation can complicate PGT inference and lead to elevated rates of false positive detection. Since the existing datasets do not account for this possibility, we simulated 6 additional datasets where the PGT and non-PGT regions evolve at different rates.

6. **Higher PGT rate datasets.** We have three datasets for which the non-PGT and PGT region mutation rates are (0.25, 0.5), (0.5, 1.0), and (0.5, 2.0).
7. **Higher non-PGT rate datasets.** We have three datasets for which the non-PGT and PGT region mutation rates are (0.5, 0.25), (1.0, 0.5), and (2.0, 0.5).

3.2 Biological datasets

In addition to the simulated datasets, we used two published biological datasets to gauge the real-world impact of using HexSplit. Our first dataset is a broadly sampled dataset containing over 4700 gene families from 100 species (11 eukaryotic, 12 archaeal, and 67 bacterial) [6]. Since these taxa are distantly related, lower prevalence of disruptive PGT is expected. The second dataset consists of 8277 gene families from 103 *Aeromonas* strains [10, 18], representing closely related taxa with higher anticipated rates of homologous recombination and PGT. To maximize PGT detection accuracy, we filtered the gene families in these two datasets by removing all gene families that had fewer than 40 genes or alignments shorter than 150 amino acids or 450 nt. Since neither trippd nor HexSplit is currently set up to properly handle large gaps (see discussion on future work in the last section), we also removed gene families with large gaps. After filtering, 802 gene families from the broadly sampled dataset and 466 from the *Aeromonas* dataset were retained for PGT analysis.

4 Results

To demonstrate the improvement of HexSplit over trippd, we applied both methods to all our simulated and biological datasets.

4.1 Results on simulated datsets

Default dataset results. On the default dataset, we find that HexSplit improves upon the false-negative PGT detection rate of trippd from 0.20 to 0.16 while still maintaining a false-positive rate of 0. Thus, HexSplit strictly improves upon the accuracy of trippd on this dataset, reducing trippd's false-negative rate by 20% and matching trippd's false-positive rate. These results demonstrate that HexSplit is able to find additional cases of PGT without a significant increase in the rate of false-positive detection, i.e., that it can achieve improved sensitivity without a significant reduction in specificity. Our results for the other categories of datasets, given below, support this initial finding.

Impact of evolutionary parameters on detection accuracy. Figure 3 shows the false-negative and false-positive detection rates for trippd and HexSplit for all of the other categories of simulated datasets (each corresponding to a key evolutionary parameter). For the PGT location datasets, we find that HexSplit shows a drastically reduced false-negative rate (Figure 3), especially for offsets of 84bp and 134bp, without any significant increase in the false-positive rate. HexSplit also shows a drastic improvement on the Multi-PGT dataset, with the false-negataive rate reduced from 0.43 to 0.21 (Figure 3). The false-positive rate increases from 0.07 to 0.14, but this is more than offset by the very large reduction in the false-negative detection rate. Improvements are more modest for the other categories of datasets (plots labeled PGT region length, Higher PGT rate, Higher non-PGT rate, and Substitution rate in Figure 3). These datasets all show consistent reductions in the false-negative rate, but also often show comparable increases in the false-positive detection rate. Importantly, the rates of false-positive detection remain low overall and are not much higher than those of trippd across nearly all cases. Overall, these benchmarking results show that HexSplit can greatly improve PGT detection accuracy in some evolutionary scenarios while providing better or comparable accuracy to trippd in nearly all other scenarios.

Overall accuracy in terms of F_1 score. Table 1 shows the corresponding F_1 score for each of the six plots shown in Figure 3. F_1 scores account for the trade-off between false-positive and false-negative detection rates and therefore present a unified view of the overall accuracy of each method, with higher F_1 score representing greater overall accuracy. As the table shows, HexSplit improves upon the F_1 score of trippd across five of the six categories of simulated datasets. across all evolutionary parameters tested. The largest improvements in accuracy are seen for the PGT location datasets, where the F_1 score increases from 0.83 to 0.89, and for the Multi-PGT dataset, where the the F_1 score increases from 0.72 to 0.87. As observed previously, these results show that HexSplit is far more sensitive at detecting PGTs when they occur in the interior of the sequence alignment as well as when there are multiple distinct PGT regions within the same alignment. Importantly, as the Table 1 shows, the increased sensitivity

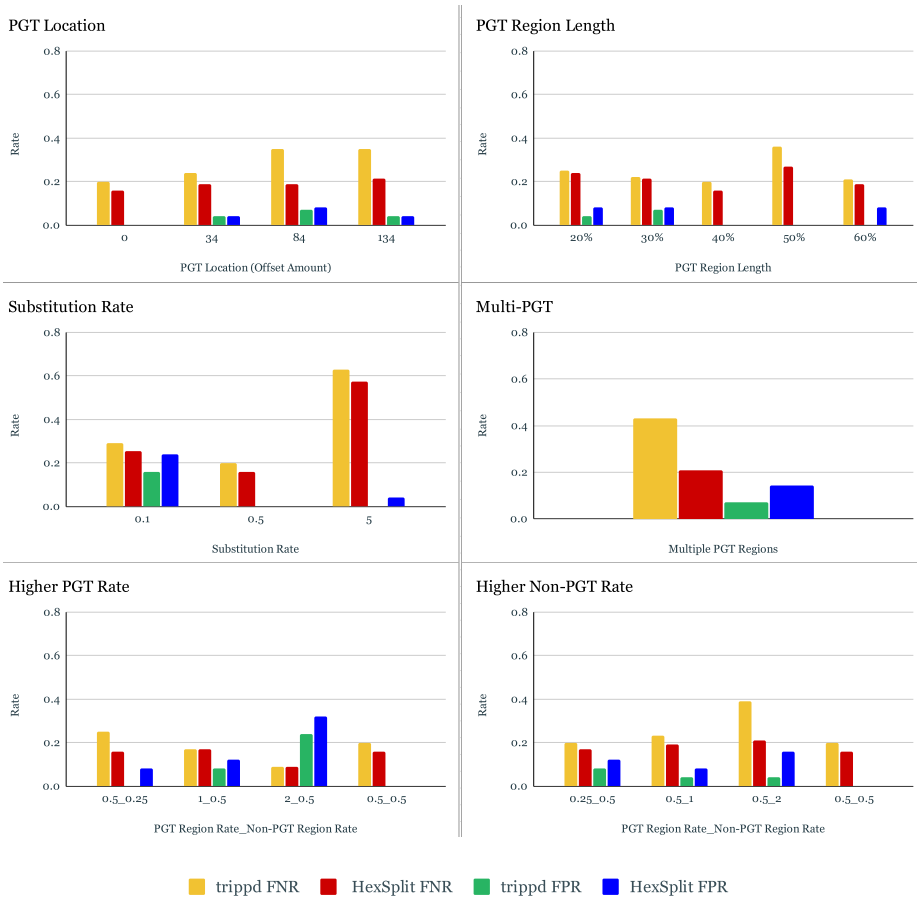

Fig. 3: **PGT detection accuracy for different simulated datasets.** False-positive and false-negative detection rates are shown for HexSplit and trippd. Each plot shows the effect of varying a specific evolutionary parameter (each corresponding to a different category of simulated datasets). Results for the default dataset are included in all plots except for the Multi-PGT plot. All results are averaged over 100 gene families.

Table 1: Average F_1 scores (PGT detection accuracy) of trippd and HexSplit for different categories of simulated datasets corresponding to the results shown in Figure 3.

Dataset category	**trippd**	**HexSplit**
PGT Location	0.83	0.89
PGT region length	0.85	0.87
Substitution rate	0.82	0.77
Multi-PGT	0.72	0.87
Higher PGT rate	0.89	0.90
Higher non-PGT rate	0.85	0.88

of HexSplit over trippd does not come at the expense of substantially worsened false-positive detection. An exception is the substitution rate dataset category, where the F_1 score of HexSplit is less than that of trippd (0.77 versus 0.82). This suggests that very low or very high sequence mutation rates, which both lead the reduced phylogenetic signal, can sometimes mislead HexSplit into incorrectly inferring the presence of PGT.

4.2 Results on biological datasets

To demonstrate the utility of HexSplit for identifying phylogenetically disruptive PGT in microbial gene families, we applied HexSplit and trippd to the two biological datasets described previously. For the 802 gene families in the broadly sampled dataset, HexSplit identified 97 gene families as having PGT (12.1%) while trippd detected 62 cases. Thus, detects HexSplit detects about 50% more cases of PGT than trippd. Still, both methods identify relatively small fractions of these gene families as being affected by PGT; this is expected given the large evolutionary distances between the taxa in this datasets.

For the *Aeromonas* dataset, HexSplit identified 209 out of the 466 gene families (44.8%) as having PGT compared to 151 families (32.4%) by trippd. Thus, HexSplit is again substantially more effective at identifying PGT than trippd. Both trippd and HexSplit identify a large fraction of these gene families as having PGT; this is consistent with the observation that, since the taxa included in this dataset are very closely related, there should be greater abundance of horizontal gene transfer and PGT. These results should be interpreted with caution since additional factors, such as alignment error, can inflate these numbers in real datasets. Still, this analysis hints at the prevalence and widespread phylogenetic impact of PGT in real datasets.

5 Discussion and Conclusion

In this work, we introduced an improved method, HexSplit, for the detection of phylogenetically disruptive PGT. Using a thorough experimental study, we demonstrated the substantial improvement in detection accuracy offered by HexSplit over trippd, the previous version of this method. Specifically, our experiments show that, across most of

the evaluated evolutionary conditions, HexSplit increases the sensitivity of PGT detection without significantly decreasing precision, resulting in higher F_1 scores than trippd. We also demonstrate HexSplit's impact in practice by applying it to two large microbial gene family datasets.

We envision HexSplit or its derivatives being used as part of automated gene tree reconstruction pipelines that reconstruct gene trees for thousands of gene families. Such automated pipelines are frequently used as part of phylogenomic analysis, orthology inference, protein function assignment, etc. Gene families identified by HexSplit as having phylogenetically disruptive PGT can then be either discarded, analyzed separately using more sophisticated alignment and/or recombination-aware phylogeny construction methods such as PhyML-Multi [2], or manually curated, resulting in improved accuracy of downstream inferences. To that end, there are at least three aspects of HexSplit that can be improved further. First, it may be possible to substantially speed up HexSplit without sacrificing accuracy by using faster maximum likelihood phylogeny construction methods such as FastTree 2 [17] instead of RAxML. Second, applicability can be improved by explicitly identifying and accounting for gene sequences with large gaps in the gene family alignment. In particular, it is possible that, due to large gaps, some of the genes in a gene family have very little or no sequence information for one or more HexSplit windows. Such cases could be handled by identifying such sequences, removing them from the window trees, and using a missing-data-aware phylogenetic distance measure, such as RF(+) [22], for comparing the window trees. Finally, it should be possible to dynamically adjust the threshold for the distribution overlap test depending on the specific characteristics, such as alignment length, bootstrap support values, etc., of the given gene family alignment. Our preliminary experiments (results not shown) suggest that adjusting this threshold upwards or downwards from the default 50% can significantly improve PGT detection accuracy, especially for more complex evolutionary scenarios such as when the alignment is affected by multiple PGT events.

Acknowledgments: We thank Sofia Furman, a high-school student who worked with MSB during Summer 2025, for helping to assess the accuracy of GARD for PGT detection. We also thank Siavash Mirarab for helpful discussions.

Funding: This work was supported in part by NSF award IIS 1553421 to MSB.

References

1. Boc, A., Makarenkov, V.: Towards an accurate identification of mosaic genes and partial horizontal gene transfers. Nucleic Acids Res. **39**(21), e144 (Nov 2011)
2. Boussau, B., Guéguen, L., Gouy, M.: A mixture model and a hidden markov model to simultaneously detect recombination breakpoints and reconstruct phylogenies. Evol. Bioinform. Online **5**, 67–79 (Jun 2009)
3. Brinkmann, H., Göker, M., Koblízek, M., Wagner-Döbler, I., Petersen, J.: Horizontal operon transfer, plasmids, and the evolution of photosynthesis in Rhodobacteraceae. The ISME Journal **12**, 1994 – 2010 (2018)
4. Chan, C.X., Beiko, R.G., Darling, A.E., Ragan, M.A.: Lateral transfer of genes and gene fragments in prokaryotes. Genome Biology and Evolution **1**, 429–438 (2009)
5. Chan, C.X., Darling, A.E., Beiko, R.G., Ragan, M.A.: Are protein domains modules of lateral genetic transfer? PLoS ONE **4**(2), e4524 (02 2009)

6. David, L.A., Alm, E.J.: Rapid evolutionary innovation during an archaean genetic expansion. Nature **469**(7328), 93–96 (Jan 2011)
7. Dunning, L.T., Olofsson, J.K., Parisod, C., Choudhury, R.R., Moreno-Villena, J.J., Yang, Y., Dionora, J., Quick, W.P., Park, M., Bennetzen, J.L., Besnard, G., Nosil, P., Osborne, C.P., Christin, P.A.: Lateral transfers of large dna fragments spread functional genes among grasses. Proceedings of the National Academy of Sciences **116**(10), 4416–4425 (2019)
8. Hollingshead, S.K., Becker, R., Briles, D.E.: Diversity of PspA: Mosaic Genes and Evidence for Past Recombination in *Streptococcus pneumoniae*. Infection and Immunity **68**(10), 5889–5900 (2000)
9. Jaya, F.R., Brito, B.P., Darling, A.E.: Evaluation of recombination detection methods for viral sequencing. Virus Evolution **9**(2), vead066 (11 2023)
10. Kloub, L., Gosselin, S., Fullmer, M., Graf, J., Gogarten, J.P., Bansal, M.S.: Systematic Detection of Large-Scale Multigene Horizontal Transfer in Prokaryotes. Molecular Biology and Evolution **38**(6), 2639—-2659 (2021)
11. Kosakovsky Pond, S.L., Posada, D., Gravenor, M.B., Woelk, C.H., Frost, S.D.: Gard: a genetic algorithm for recombination detection. Bioinformatics **22**(24), 3096–3098 (11 2006)
12. Martin, D., Rybicki, E.: RDP: detection of recombination amongst aligned sequences. Bioinformatics **16**(6), 562–563 (06 2000)
13. Martin, D.P., Varsani, A., Roumagnac, P., Botha, G., Maslamoney, S., Schwab, T., Kelz, Z., Kumar, V., Murrell, B.: RDP5: a computer program for analyzing recombination in, and removing signals of recombination from, nucleotide sequence datasets. Virus Evolution **7**(1), veaa087 (04 2020)
14. Petersen, J., Wagner-Dobler, I.: Plasmid transfer in the ocean – a case study from the roseobacter group. Frontiers in Microbiology **8**, 1350 (2017)
15. Posada, D., Crandall, K.A.: Evaluation of methods for detecting recombination from DNA sequences: computer simulations. Proc. Natl. Acad. Sci. U. S. A. **98**(24), 13757–13762 (Nov 2001)
16. Posada, D., Crandall, K.: The effect of recombination on the accuracy of phylogeny estimation. Journal of Molecular Evolution **54**, 396–402 (2002)
17. Price, M.N., Dehal, P.S., Arkin, A.P.: FastTree 2 – Approximately Maximum-Likelihood Trees for Large Alignments. PLOS ONE **5**(3), 1–10 (03 2010)
18. Rangel, L.T., Marden, J., Colston, S., Setubal, J.C., Graf, J., Gogarten, J.P.: Identification and characterization of putative Aeromonas spp. T3SS effectors. PLOS ONE **14**(6), 1–20 (06 2019)
19. Robinson, D.F., Foulds, L.R.: Comparison of phylogenetic trees. Math. Biosci. **53**(1), 131–147 (Feb 1981)
20. Ruths, D., Nakhleh, L.: Recombination and phylogeny: effects and detection. International journal of bioinformatics research and applications **1**(2), 202–212 (2005)
21. Stamatakis, A.: RAxML version 8: a tool for phylogenetic analysis and post-analysis of large phylogenies. Bioinformatics **30**(9), 1312–1313 (May 2014)
22. Yao, K., Bansal, M.S.: Optimal Completion and Comparison of Incomplete Phylogenetic Trees Under Robinson-Foulds Distance. In: Gawrychowski, P., Starikovskaya, T. (eds.) 32nd Annual Symposium on Combinatorial Pattern Matching (CPM 2021). Leibniz International Proceedings in Informatics (LIPIcs), vol. 191, pp. 25:1–25:23. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl, Germany (2021)
23. Zaman, S., Bansal, M.S.: On partial gene transfer and its impact on gene tree reconstruction. In: Comparative Genomics. pp. 168–186. Springer International Publishing (2022)
24. Zhaxybayeva, O., Lapierre, P., Gogarten, J.P.: Genome mosaicism and organismal lineages. Trends in Genetics **20**(5), 254–260 (2004)
25. Zheng, Y., Roberts, R.J., Kasif, S.: Segmentally variable genes:a new perspective on adaptation. PLOS Biology **2**(4), E81 (04 2004)