Nucleic Acids Research, 2025, **53**, gkaf266
https://doi.org/10.1093/nar/gkaf266
Methods

DUPLEXDISCOVERER: a computational method for the analysis of experimental duplex RNA–RNA interaction data

Egor Semenchenko[1,2], Volodymyr Tsybulskyi[1,2], Irmtraud M. Meyer[1,2,3,*]

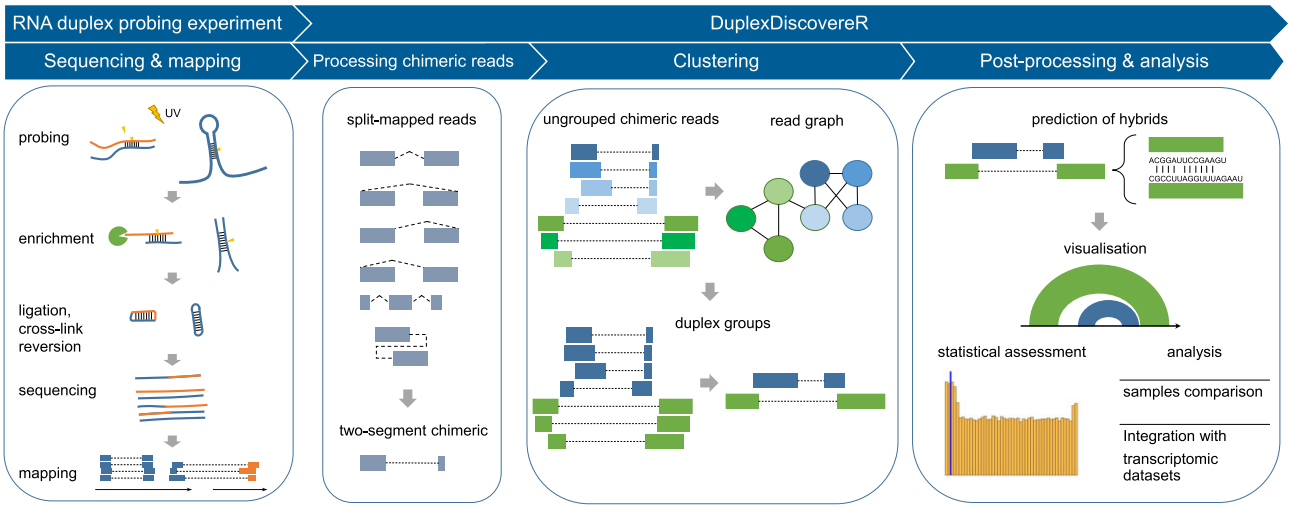
[1]Laboratory of bioinformatics of RNA Structure and Transcriptome Regulation, Berlin Institute for Medical Systems Biology, Max Delbrück Center for Molecular Medicine, Robert-Rössle-Str. 10, 13125 Berlin, Germany
[2]Department of Biology, Chemistry and Pharmacy, Institute of Chemistry and Biochemistry, Thielallee 63, Freie Universität Berlin, 14195 Berlin, Germany
[3]Department of Mathematics and Computer Science, Institute of Computer Science, Takustraße 9, Freie Universität Berlin, 14195 Berlin, Germany

*To whom correspondence should be addressed. Email: irmtraud.meyer@cantab.net

Abstract

For a few years, it has been possible to experimentally probe the universe of *cis* and *trans* RNA–RNA interactions in a transcriptome-wide manner. These experiments give rise to so-called duplex data, i.e. short reads generated via high-throughput sequencing that each encode information on a *cis* or *trans* RNA–RNA interaction. These raw duplex data require complex, subsequent computational analyses in order to be interpreted as solid evidence for actual *cis* and *trans* RNA–RNA interactions. While several methods have already been proposed to tackle this challenge, almost all of them lack one or more desirable feature—computational efficiency, ability to readily alter the main processing steps and parameter values, *p*-value estimation for predictions, and interoperability with the common bioinformatics tools for transcriptomics. To overcome these challenges, we present DUPLEXDISCOVERER—a computational method and R package that allows for the efficient, adjustable, and conceptually coherent analysis of duplex data. DUPLEXDISCOVERER is readily adaptable to analysing data from different experimental protocols and its results seamlessly integrate with the most commonly used bioinformatics tools for transcriptomics in R. Most importantly, DUPLEXDISCOVERER generates predictions that are of superior or comparable quality to those of the existing methods while significantly improving time and memory efficiency.

Graphical abstract



Introduction

Overview

It has long been acknowledged that RNA transcripts have a much wider range of functions than merely acting as messengers between a genome and its encoded proteins [1, 2]. Moreover, many genomes—including our human one—seem to encode many more non-coding genes (RNA genes) than protein-coding ones. The functional roles of the former remain mostly unknown [3]. All transcripts—whether protein-coding or not—can exert some of their functional role(s) in *cis* via their so-called RNA structure(s) [4, 5] as well as in *trans* via direct RNA–RNA interactions between transcripts [6, 7] or

Received: January 29, 2024. **Revised:** March 7, 2025. **Editorial Decision:** March 11, 2025. **Accepted:** March 31, 2025
© The Author(s) 2025. Published by Oxford University Press on behalf of Nucleic Acids Research.
This is an Open Access article distributed under the terms of the Creative Commons Attribution-NonCommercial License (https://creativecommons.org/licenses/by-nc/4.0/), which permits non-commercial re-use, distribution, and reproduction in any medium, provided the original work is properly cited. For commercial re-use, please contact reprints@oup.com for reprints and translation rights for reprints. All other permissions can be obtained through our RightsLink service via the Permissions link on the article page on our site—for further information please contact journals.permissions@oup.com.

with other molecules in the cellular environment, e.g. ligands, proteins [8, 9]. In contrast to networks of protein–protein interactions, the universe of cellular *trans* RNA–RNA interactions between two (or more) transcripts remains largely unexplored [10, 11]. We are also only beginning to get a glimpse of all the functional RNA structures that regulate how genes *in vivo* are expressed in a cell-specific manner. Overall, the universe of functional *cis* and *trans* RNA–RNA interactions *in vivo* thus remains mostly unchartered territory.

To change the above status quo, we require experimental and computational methods that are able to tackle the many inherent challenges that the investigation of *cis* and *trans* RNA–RNA interactions *in vivo* poses [12, 13]. Since 2016, we have the first methods that enable the transcriptome-wide probing of *cis* and *trans* RNA–RNA interactions *in vivo* in a high-throughput manner. These methods utilize psoralen or a psoralen-derived chemical for the covalent cross-linking of double-stranded regions of RNA (which are double-stranded because they interact either in *cis* or in *trans*) followed by proximity ligation and high-throughput sequencing of the resulting short reads, e.g. *SPLASH* [14], *PARIS* [15], and *LIGR-seq* [16].

Another class of related methods—*CLASH* [17], *MARIO* [18], *RIL-seq* [19], and *RIC-seq* [20]—studies *cis* and *trans* RNA–RNA interactions that involve RNA-binding proteins (RBPs). Both classes of methods aim to identify RNA duplexes, i.e. regions of *cis* or *trans* interacting double-stranded RNA.

The latter family of methods for investigating RBP-mediated RNA–RNA interactions currently has a higher efficiency than the first class of methods as the binding of a protein allows for a pull-down step and thus a more efficient enrichment in the corresponding experimental protocols while the former methods offer a potentially more unrestricted view of the RNA–RNA interactome. We will refer to both classes of methods as RNA duplex probing methods in the following.

Challenges of the existing RNA–RNA interaction probing methods

One major challenge of all RNA duplex probing methods is the complexity of their experimental protocols and their overall low efficiency [21]. Several methodological steps have recently been improved in the *PARIS2* protocol [22]. The cross-linking chemistry has been enhanced by using a different cross-linking compound. In addition, a new extraction protocol has been developed which allows for the enrichment of cross-linked fragments. Despite these improvements, proximity ligation remains a persistent bottleneck, causing all currently existing RNA duplex probing methods to deliver sparse and noisy data. Therefore, the subsequent bioinformatics analysis of the raw experiment data is of utmost importance for extracting evidence for functionally relevant *cis* and *trans* interactions. This is a key pre-requisite for any subsequent biological interpretation and functional analysis of any RNA duplex probing experiment.

Bioinformatics analysis of the RNA duplex data

RNA duplex probing methods produce data in the form of short chimeric reads, i.e. RNA-seq sequencing reads, consisting of two or more segments, where each segment (also referred to as read arm) corresponds to a sub-sequence of a transcript which forms an RNA duplex. These reads are also called duplex reads. An overview of the key steps of any RNA duplex data analysis is shown in Fig. 1, using our method DuplexDiscoverer as an example.

The first step involves the mapping of the reads and—importantly—the correct identification of the chimeric reads within those. Any mistakes or omissions made in this first step cannot be fixed in the subsequent bioinformatics analysis. This first step is thus of utmost importance as well as a key challenge due to the potentially large choice of alignment tools and the myriad of corresponding potential parameter settings. Technically, chimeric reads are mapped similarly to non-chimeric reads split at splice junctions. The length of a duplex read, however, is typically comparatively short and does not exceed 50 nt. Finding the correct split alignments can thus pose a challenge to any mapping software. Moreover, reads may not only require the correct mapping of their respective fragments, but any fragment may—at least in principle—span a splice junction, further complicating the alignment procedure. Overall, it is thus fair to conclude that the complexities of correctly mapping the rather short fragments of a chimeric read to a much larger mapping space pose a tremendous challenge.

Once duplex data have been mapped, alignments can be classified according to their type, see Fig. 1. Chimeric alignments containing two segments are usually considered as the primary type of duplex reads and subjected to the main analysis, whereas more complex arrangements are ignored or analysed separately.

In the subsequent steps of the bioinformatics analysis pipeline, the duplex reads are clustered into so-called ‘duplex groups’ (DGs) based on the overlap of their alignments. DGs thus—hopefully—define boundaries within which the RNA duplexes are located. The number of reads contained in a DG is assumed to be proportional to the relative abundance of the corresponding RNA duplex. Even though grouping into DGs might be considered a trivial procedure, its implementation is not straightforward and may consume a significant amount of computational resources and time if not undertaken in an algorithmically well-defined and conceptually carefully chosen way.

In the final step, the detected duplex RNA–RNA interactions are assigned a confidence score and can then be ranked accordingly. Each ranking may choose to assess a particular feature of the detected duplex. Commonly used features include e.g. alignment coverage, random ligation probabilities, or computationally estimated free energy of the duplex hybridization. Due to a lack of reliable positive and negative control reference datasets, arbitrary cut-offs are typically used by existing methods to distinguish between genuine and spuriously detected RNA duplexes i.e. between true and false positives.

Existing methods for the bioinformatics analysis of the RNA duplex data

Currently, any analysis of duplex data requires state-of-the-art software. All published RNA duplex probing studies establish and propose a corresponding, dedicated computational analysis pipeline alongside their particular experimental protocol for generating the raw RNA duplex data. Otherwise, the corresponding raw duplex reads could not have been analysed by the corresponding studies. Each experimental protocol currently comes with its own computational analysis pipeline. While this current state of affairs is understandable, the differences between these computational analysis pipelines make

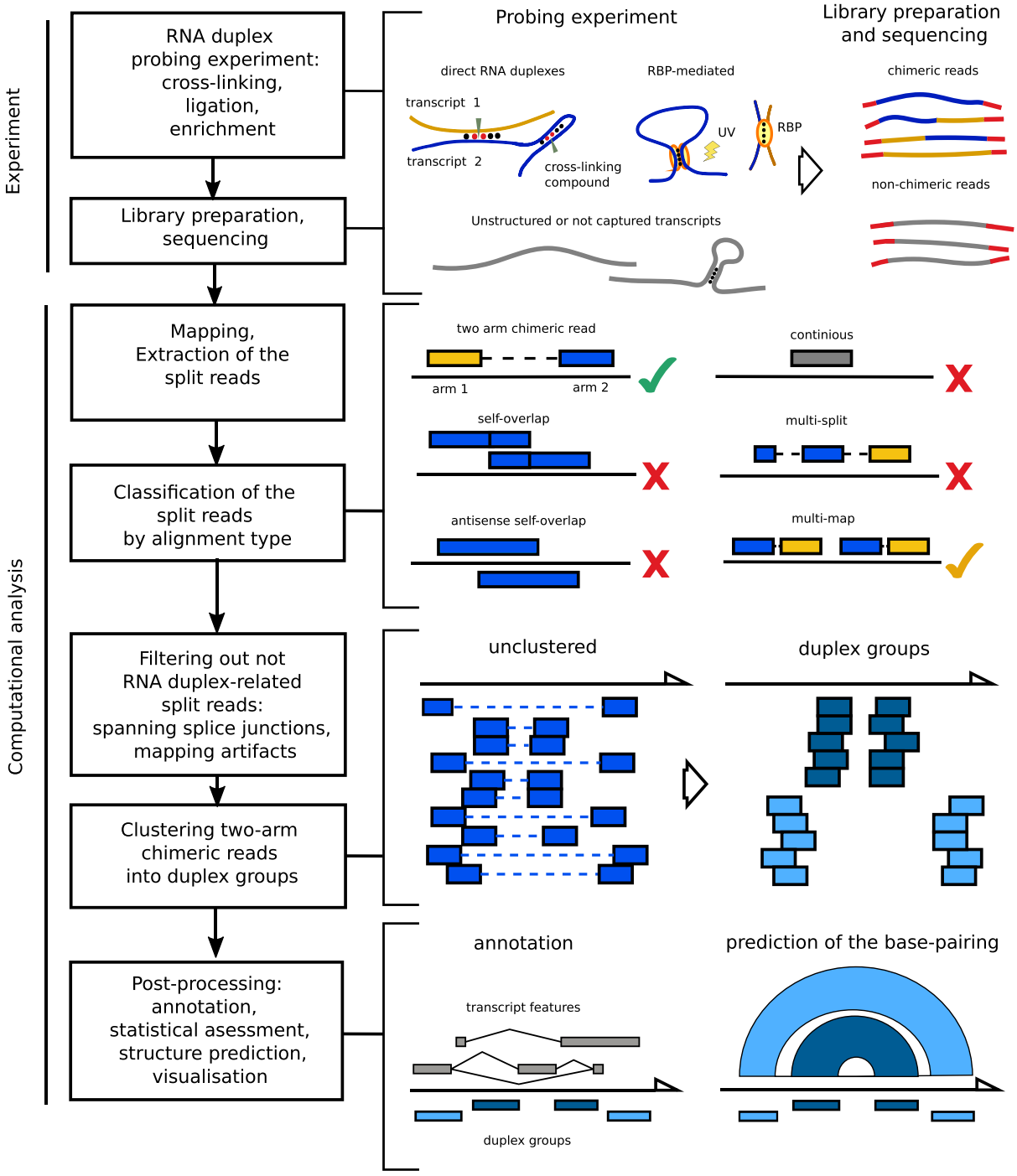


Figure 1. Overview of the RNA duplex probing data analysis with DUPLEXDISCOVERER. Experimental part: depending on the experimental protocol, the RNA duplexes are captured through reversible UV-dependent cross-linking with probing compound or through cross-linking to the bound protein. After the protocol-specific fragmentation, ligation, and enrichment steps, sequencing libraries are prepared, where RNA duplexes are recorded as chimeric inserts. Computational part. Classification: reads are categorized by the type of alignment. Only two-arm split reads are retained for further analysis. Multi-mapped reads can be omitted or analysed assuming them as unique. Filtering: two-arm split reads are compared against a splice junction database and disregarded if the chimeric junctions coincide with a splice junction. Reads with unexpectedly short chimeric junctions are filtered out as mapping artefacts. Clustering: chimeric reads are clustered by the amount of overlap between the alignments and collapsed into 'duplex groups' (DGs). Post-processing: DGs are annotated with transcriptome or genome features. Base-pairing and hybridization energies are calculated. The probabilities of the DGs being the result of the random ligation are estimated based on the corresponding abundances of the genes or transcripts.

the replication and principled comparison of results difficult to impossible. Moreover, any user wishing to refine or adjust selected steps of any computational analysis pipeline is faced with the daunting task of having to delve deeply into the source code of the software that is typically not laid out in a modular way.

In order to make the computational handling of RNA duplex data more uniform and efficient, several computational pipelines have already been proposed [23–25]. We discuss their main features below and summarize their differences in Table 1.

RNAnue

RNAnue [24] is a computational analysis pipeline that takes the raw reads of an RNA duplex probing experiment as input and outputs clustered DGs. The main steps of RNAnue follow the outline described above, with the addition of the steps for trimming raw reads. The authors of RNAnue claim that they obtain a higher yield (1–1.5) of chimeric reads compared to the respective original publications [14–16]. The increase in sensitivity is attributed to the use of the mapping method Segemehl [26], which has been shown to find more correct split mappings. In addition to the statistical evaluation and the use of hybridization energies, RNAnue offers a so-called complementarity score as an additional way of filtering DGs.

RNAnue can be considered a solid choice for the analysis of duplex data. The main obstacle to its practical use, however, is its implementation, which is detached from the commonly used toolset for RNA-seq analysis. Also, RNAnue is implemented as a `C++` package with multiple dependencies that requires compilation and packaging by experienced users. In addition, the memory benchmarks for RNAnue imply that the package has to be built for a high-performance computing (HPC) environment which many users may only have restricted access to, further limiting its use in practice. For our work with duplex data, neither we nor our HPC IT specialists were able to successfully install and test this tool due to numerous dependencies and packaging issues. See Supplementary Table S1 for additional details.

ChiRA

ChiRA [23] has been developed as a complete analysis and visualization framework for duplex data. ChiRA differs significantly from other approaches and includes several steps that are exclusive to this method as it relies on transcriptome rather than genome alignments. It aims to solve the problem of mapping the chimeric reads by first collecting as many ambiguous read alignments as reasonable and resolving them probabilistically later.

ChiRA's post-mapping algorithm involves several steps. Briefly, ChiRA considers each part of the split alignment of a duplex read separately and—for clustering of DGs—builds a so-called dataset of common read loci (CRL), i.e. groups of mapping sites that are aggregated if they share a certain number of multi-mapped reads. ChiRA uses an expectation-maximization quantification algorithm to estimate the expression of CRLs and simultaneously selects the most likely CRL and transcript attributed to the RNA duplex.

The complexity of the ChiRA pipeline and the need to resolve ambiguities introduced by the transcriptome mapping can be viewed as a disadvantage compared to other computational analysis pipelines. In particular, quantifying the expression for loci of dynamic size is a procedure for which it can be difficult to find the right intuition about the algorithm's parameters or the interpretation of the notion of 'locus expression' in terms of TPM units. ChiRA is implemented as a set of Python scripts and as a complete GALAXY-based analysis and visualization pipeline [31]. ChiRA was developed and tested on the published CLASH data sets and supports the use of split references, one containing one type of RNA microRNA (miRNAs) and another containing the rest of the transcriptome, making ChiRA best suited for the analysis of CLASH duplex reads where such a dedicated split reference will highlight any interactions involved with miRNAs.

CRSSANT

Developed by one of the authors of *PARIS*, CRSSANT [25] provides a complete RNA duplex data processing pipeline using a state-of-the-art alignment scoring approach and a thorough classification and clustering methodology. CRSSANT proposes a considerable overhaul of the alignment procedure by disabling the STAR gap score penalties and relaxing other chimeric alignment scoring settings. These changes aim to increase the number of chimeric reads that can be detected and to recover the short alignments that share loci with more reliable ones. CRSSANT uses an extended classification scheme which allows the user to keep track of all mapped reads, chimeric or otherwise. A special feature – implemented for the first time in CRSSANT – is the categorization of reads mapped with more than one split into the separate `gapm` group. This allows for the possibility of investigating more complex arrangements of RNA duplexes. The clustering of chimeric reads is implemented by building multiple gene–gene networks from overlapping reads and searching for DGs using cliques or a spectral community detection algorithm. CRSSANT does not, however, provide any hybridization scores or any statistical evaluation of reported interactions in terms of estimated reliability values or p-values. The computed DGs can, however, at least be filtered either by coverage or by the minimum number of reads supporting the DG.

The main drawback of CRSSANT is its significant requirement for computing time and memory. The complete analysis of a single sample of an RNA duplex probing experiment takes days, even with multi-threading enabled, see Fig. 2. Furthermore, the clustering of reads is based on the existing annotation and can thus only find DGs that overlap with already known genes. This limits the usefulness of CRSSANT for detecting RNA duplexes formed by interaction with novel transcripts that are currently missing from the annotation, e.g. small non-coding RNAs. CRSSANT is implemented as a set of Python scripts that can be executed step by step. Note that it also requires the prior computation of some auxiliary files such as genome coverage tracks and gene annotations.

RNAContacts

RNAContacts [30] is a generic method, mainly designed for processing of the RIC-seq RBP-dependent RNA duplex probing data [20]. It is implemented as a Snakemake-based pipeline which can map and identify paired-end chimeric reads, cluster them by ligation points and calculate DG abundance [32]. It incorporates a two-step mapping procedure, which takes the RNA-seq library matching the sample to refine the splice-

Table 1. Key differences between the computational methods for analysing RNA duplex data DuplexDiscoverer is compatible with multiple input types and is the only package implemented to support interoperability with existing tools and formats of Bioconductor

Method	Mapping	Adapted for	Clustering	Annotation-dependent	Filtering	Visualization	Implemented in	Library type
RNAnue [24]	Segemehl [26]	Any	Read overlaps and transcript annotation	No	Binomial test, complementarity	IGV, .sam file	C++17 package	SE
ChiRA [23]	CLAN [27], bwa-mem [28]	CLASH	Common read loci	Yes	TPM, hybridization energy	Galaxy-tools	Python scripts	SE
CRSSANT [25]	STAR[29]	PARIS	Read overlaps	Yes	Coverage	IGV (bedpe, sam)	Python scripts	SE
RNAContacts [30]	STAR	RIC-seq	Chimeric junctions	Yes	None	UCSC tracks	Snakemake	PE
DuplexDiscoverer	STAR, any	Any	Read overlaps	No	Hybridization energy, Binomial test	Gviz track, IGV (bedpe, sam)	R package	Both

Abbreviations: TPM: transcript-per-million, SE/PE: single/pair-end RNA sequencing reads.

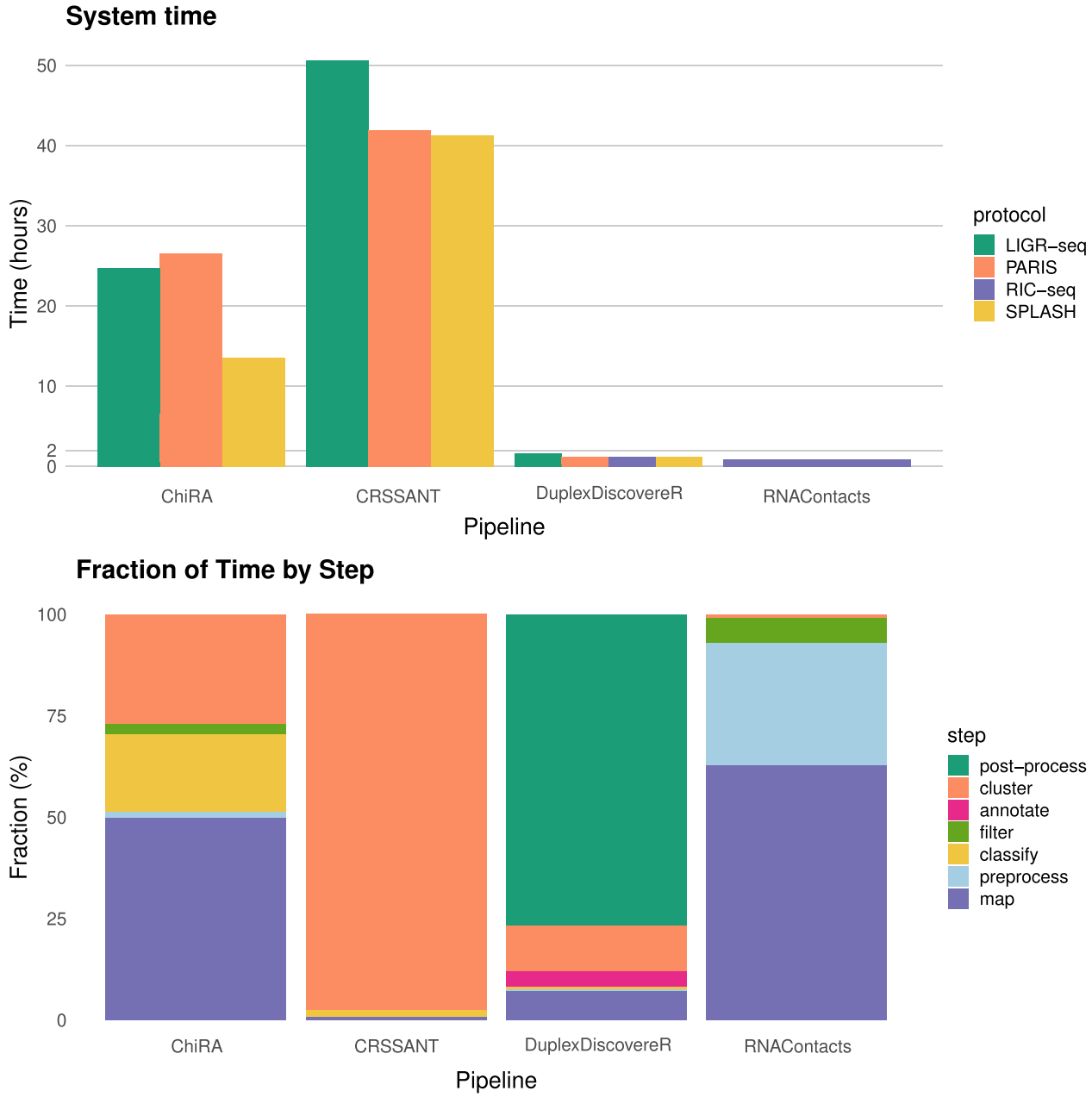


Figure 2. Compute time requirements of DuplexDiscoverer compared to that of other methods, averaged for $n = 2$ replicates of *LIGR-seq* , *PARIS* HEK, *SPLASH* ES, and *RIC-seq*–ribosomal RNA (rRNA) depleted HeLa cells RNA duplex data. As the separate analysis steps routines differ between the methods, the subroutines were annotated by their conceptual similarity. The post-processing step (calculation of hybridization energies) is not implemented in CRSSANT and RNAContacts and was omitted by us in ChiRA due to the excessive runtime requirements.

junction (SJ) database before mapping the RIC-seq probing data. RNAContacts does not embed any particular DG filtering strategy other than read coverage, leaving the calculation of the necessary metrics for the user.

Motivation for developing a new tool for analysis of the RNA duplex data

The main barrier to the detection and functional analysis of *cis* and *trans* RNA–RNA interactions in transcriptomes *in vivo* is the poor usability, the conceptual disadvantages and the high time and memory requirements of all existing computational pipelines for analysing duplex data. Given the relatively new and conceptually challenging nature of these data, it is reasonable for the user to try running any of the existing pipelines with different pipeline settings—for example—to adjust the parameters that affect the DG clustering or the filtering options. This may be particularly necessary in the case of CRSSANT, where the spectral clustering algorithm requires the user to specify values for the eigenvalue threshold as well as minimum chimeric read overlap cutoffs. The typical execution time of tens of hours for processing a single sample of RNA duplex data, however, makes the optimization of different parameter settings computationally prohibitive.

Moreover, every duplex analysis pipeline encodes a particular alignment tool with a specific set of parameter values. While the limitation of the computational analysis pipeline to one or a few alignment tools is practically unavoidable, it remains desirable to give the user full control over the alignment rules. As we mentioned earlier and as we will show in the results section below, the correct mapping and extraction of chimeric reads is a key step that requires particular attention. It is also the step that is most likely to require adaptation to a particular experimental protocol.

Currently, every computational analysis pipeline is rigid in terms of its ability to adapt the mapping step. RNAnue is only compatible with Segemehl, ChiRA can be used with CLAN or bwa-mem, and CRSSANT relies on its own Python scripts, which can only parse the sam-files generated by STAR [29] in a specific configuration and does not allow for any multimaps in duplex alignments. Once changes have been made to the mapping step, the compatibility with the existing downstream steps is no longer guaranteed. In addition, neither RNA duplex analysis pipeline offers the ability to feed previously aligned or pre-existing duplex read data as input into the pipeline to identify DGs, calculate hybridization energies or perform a statistical evaluation. These significant conceptual design limitations discourage any customization of these analysis pipelines, prevent the use of intermediate results, and leave the potential user bound to the explicit or implicit assumptions made by the authors of the computation pipeline.

Our new method—DuplexDiscoverer—is a computational pipeline for analysing duplex reads that aims to overcome all of the above challenges in the sense that it aims for (i) reasonable time and memory requirements, (ii) flexibility to change the strategy for mapping chimeric reads, (iii) high modularity in the pipeline structure, allowing the user to easily modify and add key analysis steps, and (iv) built-in visualization capabilities. DuplexDiscoverer is implemented as an R package. It facilitates the incorporation of additional layers of information into the computational data analysis, enables comparisons between samples and integrates seamlessly with the R packages widely used for transcriptome analyses.

Materials and methods

In this section, we describe the key procedures employed by DuplexDiscoverer and the data we used for analysis. Please refer to Fig. 1 for an overview of the key steps of our analysis.

Duplex-probing datasets analysed with DuplexDiscoverer

In order to investigate the merits of DuplexDiscoverer, we analysed publicly available data generated by the original RNA duplex probing methods—*LIGR-seq* [16], *SPLASH* [14], and *PARIS* [15]. In addition, we also processed two samples of RIC-seq in order to assess the functionality of DuplexDiscoverer to handle paired-end data. The full list of samples analysed and the pre-processing procedures we applied can be found in Supplementary Table S2.

Simulated RNA duplex probing datasets analysed with DuplexDiscoverer

To benchmark DUPLEXDISCOVERER against existing methods, we used four samples `SRR6811718, SRR6811722, SRR6811723, and SRR6811728` from a publicly available dataset of 100 bp single-ended RNA-seq from HeLa cells [33] to generate an artificial dataset of two-arm chimeric reads. To test the performance as a function of the read length, we generated samples with variable length of chimeric segments (referred to as arms) of 20, 30, 40, and 50 nt. Each simulated sample features a specific combination of arm lengths and comprises artificial DGs of 5–15 chimeric reads and a single RNA-seq library serving as background. To minimize background-induced variability and benchmark the detection of simulated DGs, we used `SRR6811722` as the sole background library. For the samples used for benchmarking, we added equal quantities of 30.000 *cis-* (both arms map to the same gene) and *trans-* (arms map to different genes) DGs. Key characteristics of the simulated samples are shown in Supplementary Table S3. Additionally, to control SJ filtering in DUPLEXDISCOVERER and CRSSANT, we extracted 100 000 non-chimeric SJ spanning reads from background libraries and added them to simulated datasets (see Supplementary Tables S4 and S5 for the breakdown of detected read types in the background and simulated controls). For further details on artificial DG simulation, refer to Supplementary Note S1.

Mapping of the split-reads

To make DuplexDiscoverer conceptually modular and independent from any specific alignment tool as well as any RNA-seq-based, library-specific read pre-processing, we separated the mapping of the chimeric reads from the rest of the analysis pipeline. DuplexDiscoverer analysis proceeds after the mapping is performed. We choose STAR [29] as the default alignment tool, as it is widely used for concep-

tually similar problems, e.g. for detecting fusion genes from RNA-seq reads. DuplexDiscoverer accepts as input the `Chimeric.junction.out` file of STAR, which contains the one-line records for each split alignment. When running STAR, we utilize parameter *–alignIntronMax 10*, which forces nearly all split-mapped reads to be reported as chimeric and adds the novel splice junctions to the output. The full list of DuplexDiscoverer's default mapping parameters can be found in Supplementary Table S6. Splice junction reads are distinguished from RNA duplex splits and filtered out in subsequent analysis steps.

If the user's choice is to use a different tool than STAR for mapping, the input for DuplexDiscoverer can be also provided in terms of an input file in bedpe-format specified by bedtools [34]. In that case, split reads mapped with another alignment tool need to be extracted from the sam-file and provided to DuplexDiscoverer as a bedpe-formatted input file, where the first ten mandatory fields can be followed by any number of user-defined columns. The paired-end reads are supported, and the chimeric output can be used as-is if STAR is employed for mapping. For a bedpe-format input file, the following convention in the CIGAR string is adapted: 'Lp' should be used for the gap between the read mates alignments separated by L nucleotides. Note that the value of L can be negative if these mates overlap. We use GENCODE.v44 as the reference transcriptome to generate the genome indexes. DuplexDiscoverer can also be used after the reads are mapped to the transcriptome. It is, however, left up to the user to decide how to process ambiguous chimeric read-to-transcript relations.

Classification and filtering of the alignments

After the split reads are imported into DuplexDiscoverer, each pair of split-read alignments is categorized into one or multiple of the following categories:

- multi-split, if the read is aligned to more than two locations
- multi-map, if the read is mapped ambiguously
- two-arm, if the read is aligned in split-aligned to two locations and does not belong to any of the categories below:
  - self-overlap, if the read is split-aligned to two locations and those locations overlap on the same strand
  - antisense-overlap, if the read is split-aligned to two locations and those locations overlap on opposite strands
  - small-gap, if the read is split-aligned to two locations and the chimeric junction is shorter than the user-defined threshold (10 by default)
- non-chimeric, if the read is continuously mapped or erroneously placed into the chimeric output.

If the CIGAR strings are not provided in the output, multi-split alignments cannot be identified, and any part of the split alignment is considered as the entire chimeric arm. Every category can be saved for separate treatment, i.e. the re-mapping of the multi-split alignments to the transcriptome. For our analysis in the downstream steps, only the two-arm category was used. Two-arm alignments on the same strand and chromosome are filtered by the minimum chimeric junction length and proximity to the splice junction. The minimum gap between the two arms was set to 10 nucleotides. To differentiate between chimeric and splice junctions or a circular RNA, both the start and end sites of the chimeric junction are required to be outside of the +/− 15 nt regions flanking the start and end coordinates of exon–exon junctions. Both the minimum chimeric junction length and the threshold for calling the splice junction can be readily changed in the analysis pipeline.

Clustering local alignments into DGs

To merge the two mapped parts of a duplex read to the DGs, DuplexDiscoverer uses an approach based on graph clustering. Each mapped part of the two chimeric arms is treated as a node of a graph. If two duplex reads overlap in both mapped parts, an edge between the corresponding nodes is created. Each edge is weighted by an overlap score, similar to the ones used by RNAnue and CRSSANT. The definitions of the overlaps and shifts between the chimeric reads can be found in Supplementary Fig. S1. Before finding the communities in the graph, the minimum overlap/span ratio threshold with a default of 0.3 is applied to prune the edges corresponding to poorly overlapping reads. Decreasing this parameter can increase the number of reads in DGs at the cost of longer and less well-defined RNA duplex loci. This default value for the overlap/span was chosen to optimize the global identification of reliably detected DGs. If the goal of the user is, however, to evaluate DG read support for the individual RNA helices at higher resolution, this value can be increased, resulting in an increased number of DGs with reduced read count.

To call the DGs from the chimeric reads, we use an iterative clustering strategy, see Supplementary Fig. S2. In the first step (collapsing), we collapse all reads which have identical alignment coordinates and mapping scores into single entities. In graph-based clustering, these reads will correspond to the nodes connecting the same set of neighbours because they will overlap the same reads. Collapsing therefore reduces the connectivity of the resulting graph, without losing any information.

In the second step (iterative merging), further collapsing is performed. We define the graph based only on the reads, which overlap and are shifted by a small deviation of +/−2 nt relative to each other in both mapped parts (arms). We use the Louvain community detection algorithm from the R igraph package [35] to find the clusters of reads. Each cluster is then merged into a single DG with redefined boundaries. This step is repeated multiple times—a maximum of five iterations or until there are no duplex alignments (DGs and individual reads) which are shifted by only a small amount. In the last step called 'duplex group finding', the graph based on all the overlapping duplexes is formed, and the same community detection algorithm is applied to find the final set of DGs.

Annotation of the DGs

We use genes as the basic units of annotation for the interaction clusters. Each interaction can, however, be assigned to multiple features and overlayed with multiple tracks. Thus, if the sample-specific transcriptome is known, each interaction can be readily assigned to an expressed transcript(s) or the respective exon(s). Generally, any custom track provided as an R GenomicRanges [36] object can serve as a reference annotation with some layer of information. The extraction of gene and transcript annotations from NCBI and GENCODE

GTF files is supported. By default, DG arms are annotated with gene names: DGs with arms corresponding to different genes are classified as *trans*, while those with both arms matching the same gene are classified as *cis*. When transcript-level annotation is used, *trans* and *cis* specifically refer to inter- and intra-transcript RNA–RNA interactions, respectively.

Calculating *p*-values and hybridization scores

One of the known sources of spurious signals in high-throughput duplex data is due to random ligations. Chimeric reads can be present in the duplex data as the result of two molecules being occasionally ligated due to their spatial proximity or their high abundance in the cell. To account for the latter scenario, a common approach used elsewhere [15, 19, 24] is to model the probability of such a ligation event with the binomial distribution. This model assumes that the probability of two non-interacting molecules originating from two different genes being ligated is proportional to the relative expression levels of their respective genes. In mathematical terms, it corresponds to the independent draws of transcripts '*a*' and '*b*' with the following probability density.

$$P(a,b) \propto \begin{cases} 2 \cdot P(a) \cdot P(b) & \text{if } a:b \text{ is observed and } a \neq b \\ P(a) \cdot P(b) & \text{if } a:b \text{ is observed and } a = b \\ 0 & \text{else} \end{cases} \quad (1)$$

$$P(a) = \frac{\text{N reads(a)}}{\text{total N reads}} \quad (2)$$

For each DG, the binomial test is applied to estimate the *p*-value. By default, we apply the Benjamini–Hochberg (BH) correction and retain DGs with a significance level <0.1 for subsequent analysis. For the gene-level counts, the quantification of STAR or `FeatureCounts` is supported. In case the DGs are unambiguously annotated with their transcript identities (i.e. if the transcript quantification is performed by conventional computational analysis of an RNA-seq library in parallel), the random ligation model can be used by switching from gene to transcript counts. To predict the hybridization energy, we use the RNAduplex method from the Vienna RNA [37], which is invoked as an external software package.

Implementation

DuplexDiscoverer is implemented as an R package. For storing objects for duplex data, we utilize the functionality of the `InteractionSet` library [38], originally designed for the analysis of the HiC data.

Analysis steps such as the import of split-read data, preprocessing, classification, clustering, annotation, and visualization are implemented as separate functions, allowing users to readily specify any number of intermediate steps for additional filtering, sub-setting or visualizing the data. Post-processing steps such as the calculation of hybridization energies, the estimation of *p*-values or the re-clustering of the reads into more dense or more sparse DGs can be done on any subset of data, significantly facilitating the detailed data analysis and alleviating the necessity to re-run the entire analysis pipeline whenever a specific change is introduced.

Output and visualization

DuplexDiscoverer supports the output in bedpe- and sam-formats. We implemented a custom annotation track based on the Gviz-engine [39] which enables the plotting of the interactions as a dedicated annotation track for a defined genomic region.

In addition, we provide functions which conveniently display two non-overlapping regions defined by the respective genomic loci involved in the interaction, allowing a user to visualize genome-wise distant transcript features.

Comparisons between samples

Comparisons between the DGs are not entirely straightforward for two reasons. First, many-to-many and one-to-many relations are natural when comparing ranges. One DG may match multiple DGs in another group, which can result in multiple accounts of a single observation. Second, the sizes of the DGs can vary if they are produced by different protocols or if different computational post-processing methods were used. We overcome these problems by using the following procedure. We first assemble the non-redundant super-set of RNA duplexes by gathering all samples which are to be compared, see Supplementary Fig. S3. DGs in this set are then merged into a single representative DG if they have >0.4 overlap/span ratio in each arm.

Samples are then compared to the super-set and each one-to-many hit between the super-set and the tested sample is counted only once. In this manner, we avoid over-inflating the amount of overlap between samples and preserve the ability to find imperfect matches between DGs of proximal loci, yet different sizes. The comparison procedure is implemented as a part of the DuplexDiscoverer package and can be called for an arbitrary number of samples. This facilitates the analysis of replicate reproducibility as well as the identification of the RNA–RNA interactions reliably observed by several protocols.

Benchmarking of DuplexDiscoverer and other methods

To assess the accuracy and time-memory performance of DuplexDiscoverer and other methods, we used the datasets emulating raw data of RNA duplex probing. If the simulated chimeric read is split and mapped to the correct pair of loci, it is considered a true positive (TP). Reads that are simulated as chimeric but do not appear in the chimeric output and reads that were not simulated but appeared as chimeric are considered false negatives (FNs) and false positives (FPs), respectively. Sensitivity and positive predictive value (PPV) were calculated as follows: For simulating the accuracy of the DG detection, rather than individual chimeric reads, we counted DG reported by the methods as TP, if it contained at least two correctly identified chimeric reads, and as FN, if zero reads of the simulated DG were identified.

$$\text{Sensitivity} = \frac{\text{TP}}{\text{TP} + \text{FN}} \quad (3)$$

$$\text{PPV} = \frac{\text{TP}}{\text{TP} + \text{FP}} \quad (4)$$

Specificity was not calculated as true negatives would represent the number of non-chimeric reads that would not make it into the chimeric output, which can easily be inflated by adding more non-chimeric reads to the simulated dataset.

We used CPU clock time and memory [resident set size (RSS)] metrics implemented in Snakemake [32] to calculate the time requirements of ChiRA and CRSSANT and

R *proc.time()* and *gc()* for DuplexDiscoverer. Time was measured in minutes and normalized by read count. Either method was run on the HPC cluster node with the following resources available: `Intel Xeon E5-2667 v4 8 cores/16 threads Max RAM: 256 GB Storage: DDN-SFA14K`.

We compared DuplexDiscoverer to ChiRA and CRSSANT. For RNAContacts, we evaluated only its mapping configuration (see Supplementary Fig. S4), as this pipeline does not support the single-end RNA-seq samples and therefore not suitable for direct comparisons with other methods or immediate processing of the *SPLASH*, *PARIS*, or *LIGR-seq* data. We were unable to install the RNAnue due to multiple packaging issues.

Hybridization energies for DuplexDiscoverer and CRSSANT were computed with RNAduplex. For ChiRA we used IntaRNA [40] with default parameters, by a separate call after completion of the main pipeline. Hybridization energies were not calculated for RNAContacts because this pipeline does not report the DG strand in the output.

Filtering thresholds used for analysis of results of DuplexDiscoverer and other methods

Different methods support different and incompatible filtering strategies, i.e. ChiRA supports filtering by RNA duplex hybridization energy, abundance in TPM and confidence scores, CRSSANT results can only be filtered by the geometric mean of the coverage between the DG arms `covfrac` or the number of supporting reads, and only DuplexDiscoverer estimates *p*-values. For the analysis of data from RNA duplex probing experiments and comparisons between results produced by methods, we chose to apply compatible filtering thresholds as they would be used to derive biological insights. For ChiRA , we selected only DGs with confidencescore = 1 and TPM > 10. For CRSSANT and DuplexDiscoverer, we imposed a hard threshold of at least five reads in the DG. For DuplexDiscoverer , we kept DGs with BH *p*-values < .1.

Results and discussion

Benchmarking time and memory requirements

We measured the compute time required by DuplexDiscoverer and compared it to that of other tools which can process single-end RNA-seq libraries. DuplexDiscoverer completes its analysis substantially faster, reducing the computing time for the complete analysis four-fold and DG clustering time to almost 20-fold compared to ChiRA and CRSSANT, with mapping and hybridization being the most time-consuming routines, see Fig. 2.

For CRSSANT DuplexDiscoverer and ChiRAtools, the peak memory is used during the clustering of the chimeric reads, see Table 2. For RNAContacts , the most memory-consuming part is mapping. We attribute the higher speed and the lower memory usage of DuplexDiscoverer in comparison to CRSSANT and ChiRA to our optimized read aggregation strategy. The collapsing and iterative merging procedure of DuplexDiscoverer reduces the complexity of the graph formed by all overlapping reads and thereby substantially decreases the compute time and memory required for finding the DGs. We found that *LIGR-seq* produces far more highly abundant RNA duplexes than other protocols see Supplementary Fig. S5. Applying DuplexDiscoverer to the duplex reads produced by this protocol is thus particularly beneficial for the identification of DGs. Clustering of 16 million duplex reads of *LIGR-seq* into DGs using DuplexDiscoverer takes <10 min with an optimized read aggregation strategy, compared to the 2 h when the iterative merging is disabled.

Benchmarking DG detection accuracy

The main challenge in evaluating the performance of RNA-RNA duplex identification methods is the lack of a 'gold standard' transcriptome-scale dataset that includes not only a set of known RNA secondary structures (i.e. *cis* RNA–RNA interactions) but also of known RNA–RNA interactions (i.e. *trans* RNA–RNA interactions). Specialized RNA databases such as snoDB [41], mirTarBase [42], and large-scale resources such as RNAInter [43] contain a substantial number of data sets for different organisms. In most cases, information on RNA–RNA interactions is available in the form of gene or transcript names of the interactors, however, without the detailed positions of the interacting loci. The identification of these RNA duplexes, however, is the purpose of the RNA duplex probing methods. Furthermore, cross-checking the names of interacting molecules from databases with those detected in duplex probing experiments would not evaluate the performance of the computational pipeline because each duplex probing method has a preference for certain types of RNA–RNA interactions. Therefore, observing or detecting the RNA–RNA interactions known in a database cannot be used to assess the performance of any computational method for detecting RNA duplexes.

To assess the accuracy of DuplexDiscoverer and compare it to the other methods, we tested it on simulated data that approximate the sequenced RNA-seq libraries produced in the duplex probing experiments, see the above section for more details on how this dataset was made. Several datasets of different chimeric read lengths were evaluated for calculating the PPV and sensitivity, see 'Materials and methods' section. DuplexDiscoverer has demonstrated substantially better performance on the simulated data, with the best combinations of PPV and sensitivity values obtained for the datasets with the maximum simulated chimeric arm length of 50 nt, see Fig. 3 and Supplementary Figs S6 and S7 for the *cis* and *trans* subsets and per-sample comparisons. This benchmark has shown that ChiRA is overall inferior to DuplexDiscoverer and CRSSANT, particularly in terms of PPV due to reporting a high number of false positive DGs formed by the background, non-chimeric 'normal' reads.

To assess the extent to which the accuracy of DG detection depends on the initial choice of the mapping parameters, we ran DuplexDiscoverer using several mapping schemes on the simulated data set, see Supplementary Table S6 and Supplementary Fig. S4, including mapping configurations used in RNA duplex detection tools RNAContacts, CRSSANT, and ARRIBA [44] a computational tool for identifying fusion transcripts from RNA-seq data. We find that the accuracy of RNA duplex detection in DuplexDiscoverer depends mainly on the accuracy of the alignment, see Supplementary Fig. S4. If the simulated read is correctly mapped, it was found in 96% of cases within the designed DG or as part of a larger DG formed by multiple overlapping DGs. Surprisingly, we achieve high values of PPV and sensitivity by mapping reads with the CRSSANT parameters followed

Table 2. Memory requirements—maximum RSS of DuplexDiscoverer in comparison to the other methods, averaged for $n = 3$ replicates of *PARIS*, $n = 2$ of *LIGR-seq*, *SPLASH* ES cells, and *RIC-seq*–rRNA depleted HeLa cells

Method	Step in the pipeline	N threads	Experiment	Peak RAM usage, Gb
DuplexDiscovereR	Clustering	1 (main pipeline)	LIGR-seq AMT+	17.0
	Mapping (STAR)	16 (mapping)	PARIS HEK	14
			SPLASH ES	14.5
			RIC-seq HeLa	13.1
CRSSANT	Clustering; crssnt.py	8	PARIS HEK	28.5
			LIGR-seq AMT+	52.2
			SPLASH ES	35.8
ChiRA	Clustering; chira_quantify.py	8	PARIS HEK	23.7
			LIGR-seq AMT+	46.4
			SPLASH ES	21
RNAContacts	Mapping; align.pass2 (STAR)	1–16 (not adjustable)	RIC-seq HeLa	13.1

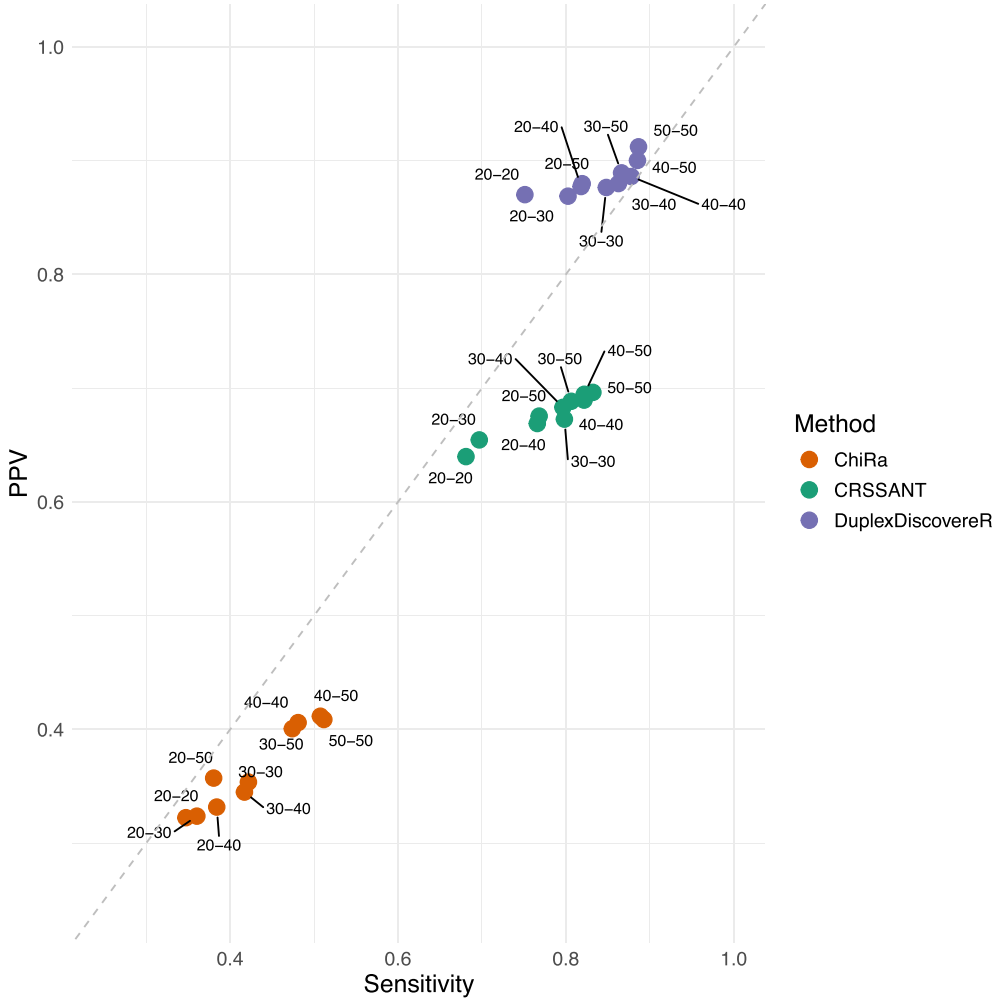


Figure 3. Benchmarking the predictive performance of DuplexDiscoverer, ChiRA, and CRSSANT on the simulated data containing artificially created DGs. The length of the chimeric parts (arms) of the reads is variable between samples, with the total length of the artificial chimeric read equal to twice the chimeric arm length.

by DuplexDiscoverer analysis downstream rather than running CRSSANT, suggesting that correctly mapped chimeric reads may be omitted by this pipeline.

DuplexDiscoverer reliably detects *cis* and *trans* RNA–RNA interactions

Due to persistent efficiency bottlenecks in RNA duplex probing experiments, the results produced by these experiments are still sparse, making it challenging to build a probabilistic model that distinguishes the biological variability in *cis* and *trans* RNA–RNA interactions from the noise introduced by the experimental protocols. To account for a known source of spurious signal in RNA duplex data—random ligation—DuplexDiscoverer uses the model proposed in [16].

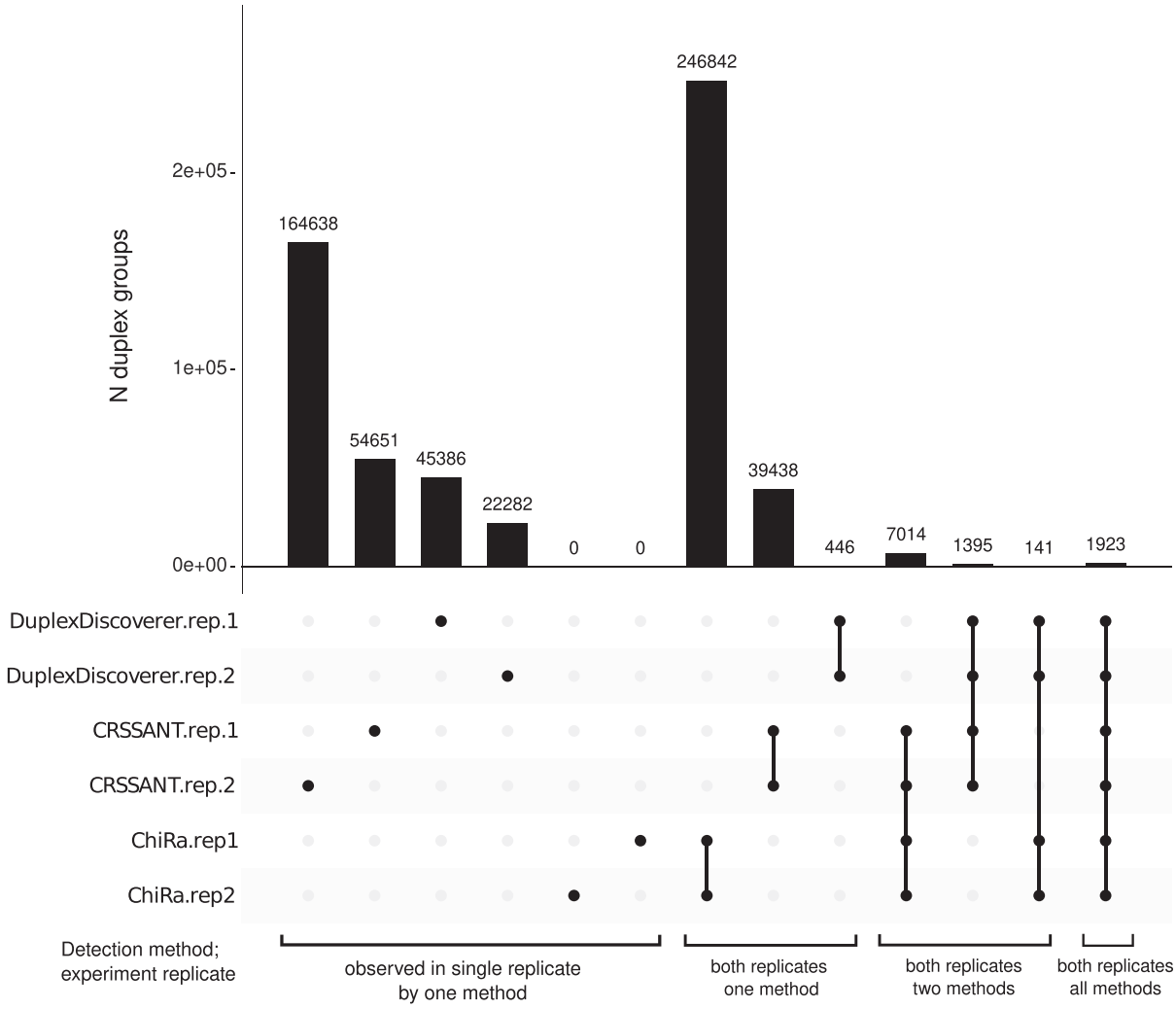


Figure 4. Comparisons of three RNA duplex detection methods for the two replicates of *SPLASH* duplex probing data performed on human embryonic stem cells. After accounting for random ligation events, only a small fraction of reads reported exclusively by DuplexDiscoverer is observed in both replicates.

Another approach to identifying the most reliably detected *cis* and *trans* RNA–RNA interactions is to analyse the reproducibility of observations by comparing biological replicates. Due to the absence of reference RNA interactome datasets, a per-replicate analysis is currently the only way to narrow down the credible observations in the duplex data.

We have observed that selecting DGs predicted by DuplexDiscoverer by the combination of replicate support and *p*-values corresponds to the lower hybridization energy—see Supplementary Fig. S8, indicating that the random ligation model used within DuplexDiscoverer provides the reasonable approximation to account for technical noise in the probing experiments, allowing the user to adopt the most reliable results for downstream analysis.

To compare the results of different methods, we analysed the duplex data of several series of RNA duplex probing experiments with DuplexDiscoverer, ChiRA, and CRSSANT. We do not detect significant differences in the distributions of the hybridization energies between results produced by three different pipelines, while the distributions of DG lengths produced by either tool are moderately different—see Supplementary Fig. S5.

The reproducibility analysis applied to all pipelines highlights substantial differences between methods. Figure 4 shows the comparison between the predictions for the two replicates of the *SPLASH* experiment, see 'Materials and methods' section for details on how the results were filtered and Supplementary Figs S9 and S10 for other experimental protocols and Supplementary Fig. S11 for the extended display of all intersection categories. In the instance of the *SPLASH* data, which has the longer chimeric reads and the least mapping uncertainty, most of the RNA duplexes predicted by DuplexDiscoverer in both biological replicates are also supported by the other methods. For *PARIS*, *LIGR-seq*, the level of cross-replicate support is lower with the tendency of higher 'single-replicate - multiple methods' overlap between DuplexDiscoverer and CRSSANT. *PARIS2* and *RIC-seq* have the lowest level of replicate reproducibility for DGs found by a single method as well as by a combination of DuplexDiscoverer with CRSSANT and RNAContacts, respectively.

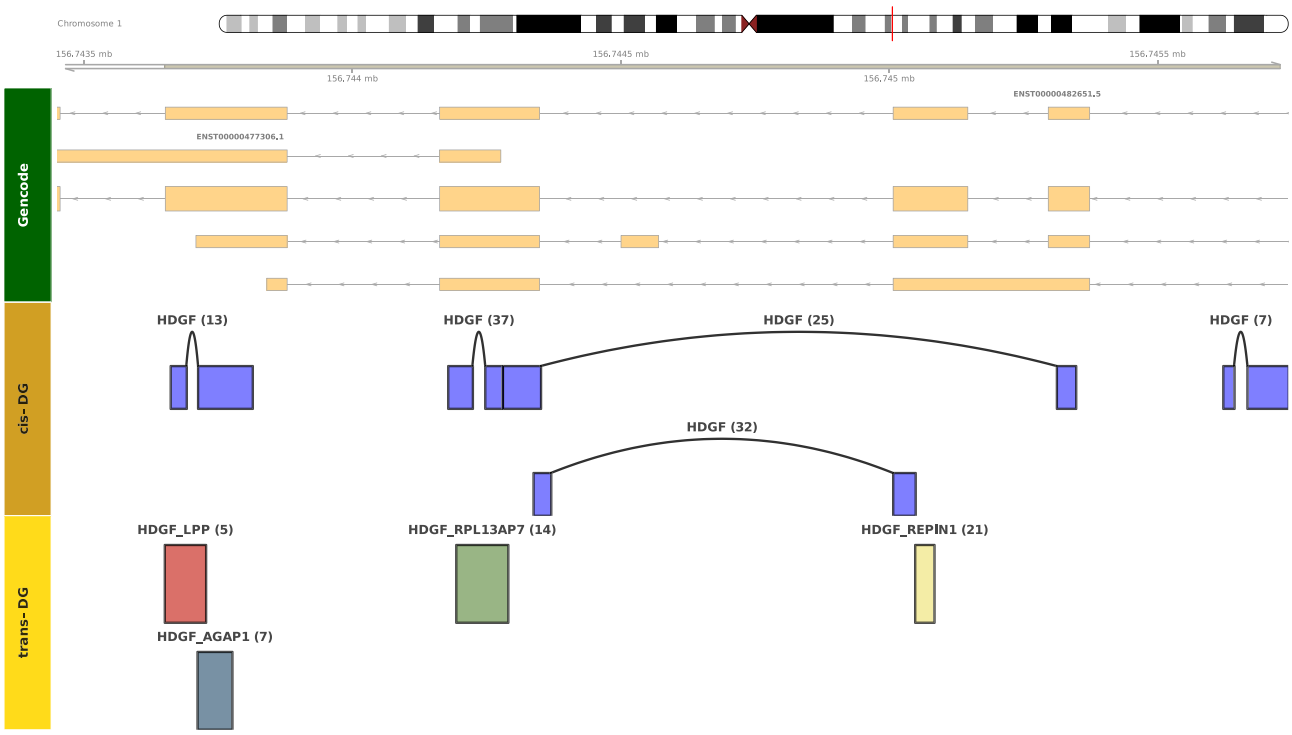


Figure 5. Example of visualization of LIGR-seq DGs on the HDGF gene detected with DuplexDiscoverer. *Cis* and *trans* DGs are split into separate tracks. For the *trans* DGs, both names of interacting genes are displayed. The number of reads supporting each DG is shown in brackets.

We found that the group 'multiple replicates—single method' has substantial size for each type of experiment and computational method. Prioritization based on hybridization energy or *p*-value is particularly important for this category because it includes both 'true' RNA duplexes detected by only one method and the spurious false positive observations. Most of the DGs in this group for DuplexDiscoverer have low statistical significance, which discriminates it from the DGs detected by multiple methods—see Supplementary Fig. S9.

Finally, the group of 'both replicates—all methods' consistently has a high fraction of the statistically significant DGs. The majority of the RNA duplexes detected in all protocols ('both replicates all methods' on Fig. 4 and Supplementary Figs S9 and S11) are also those with the best *p*-value ranking according to the *p*-value estimates by DuplexDiscoverer. Overall, the DGs detected by DuplexDiscoverer coincide most with those of CRSSANT for all probing methods, while the overlap with the results of ChiRA is moderate. Taken together, we conclude that selecting the DGs predicted by DuplexDiscoverer that are (i) reproducible between replicates and (ii) filtered by the estimated *p*-value selects the most trustworthy RNA duplexes that generally have lower hybridization energies and can also be found by other methods.

Visualization

DuplexDiscoverer provides the functionality to create a Gviz-based annotation track for visualization of detected DGs, see Fig. 5. This is particularly useful when analysing RNA duplex data in parallel with other assays, alleviating the need to switch to another environment for exploratory analysis or to create the track hub for web explorers i.e. UCSC [45]. Its use cases are not limited to *cis* RNA–RNA interactions and can be used to visualize the distant loci of the *trans* RNA duplexes, which can be as far apart as the underlying genomic coordinates [45].

Conclusions

The field of RNA duplex probing is progressing. We expect further development of experimental protocols and more data to be generated in the near future. Technological advances may open up the possibility to quantitatively analyse and compare RNA structures and interactomes in different biological contexts and between conditions, possibly translating findings from fundamental RNA research into clinical applications. Potential developments include the ability to probe RNA–RNA interactions by targeting specific transcripts, reducing bias and increasing probe efficiency in high-throughput methods. DuplexDiscoverer can easily be used by authors of new RNA duplex probing methods. The new levels of analysis i.e. quantification procedures or new statistical models for filtering and comparing RNA–RNA interaction abundances may be required to exploit the full potential of RNA duplex data generated by a new generation of experimental methods. DuplexDiscoverer provides a solid foundation on which such extensions can be built, facilitating the reproducibility and reuse of results.

By making DuplexDiscoverer available to the community, we also hope that it will encourage and enable other researchers to incorporate the existing results from RNA duplex probing experiments into their analyses. As we show, there has been no reliable and convenient way to do this before. DuplexDiscoverer is a highly customizable RNA du-

plex data analysis method that implements raw data processing, essential statistical filtering, inter-sample comparisons, and visualization. DuplexDiscoverer is not tied to any particular mapping tool and is also suitable for the discovery of RNA–RNA interactions with un-annotated transcripts. Combined with the fact that the complete analysis takes significantly less time than other pipelines, DuplexDiscoverer is currently the fastest and most user-friendly tool for the computational analysis of RNA duplex data.

Acknowledgements

The authors thank Stefan R. Stefanov for his contributions to the conceptualization in the early stages of the project.

Author contributions: E.S. contributed to the conceptualization, data curation, formal analysis, investigation, methodology, software, validation, and writing; V.T. to the data curation, software, and visualization; and I.M.M. to the conceptualization, methodology, supervision, and writing of this project.

Supplementary data

Supplementary data is available at NAR online.

Conflict of interest

None declared.

Funding

Helmholtz-Gemeinschaft. Funding to pay the Open Access publication charges for this article was provided by Helmholtz Association, Germany.

Data availability

DuplexDiscoverer is available as R 4.4 package on Bioconductor [46] and at https://doi.org/10.5281/zenodo.15013338. Package tutorial on Bioconductor contains information on the main data formats, functions, and example analyses on the test subset of RNA duplex data. The artificial datasets generated for the benchmarking and the version of the package source code used for this article are available at Zenodo [47] https://doi.org/10.5281/zenodo.10789913.

References

1. Sharp PA. The centrality of RNA. *Cell* 2009;**136**:577–80.
2. Vicens Q, Kieft JS. Thoughts on how to think (and talk) about RNA structure. *Proc Natl Acad Sci* 2022;**119**:e2112677119.
3. Mattick JS, Amaral PP, Carninci P *et al.* Long non-coding RNAs: definitions, functions, challenges and recommendations. *Nat Rev Mol Cell Biol* 2023;**24**:430–47.
4. Watters KE, Strobel EJ, Yu AM *et al.* Cotranscriptional folding of a riboswitch at nucleotide resolution. *Nat Struct Mol Biol* 2016;**23**:1124–31.
5. Kalmykova S, Kalinina M, Denisov S *et al.* Conserved long-range base pairings are associated with pre-mRNA processing of human genes. *Nat Commun* 2021;**12**:2300.
6. Xu JZ, Zhang JL, Zhang WG. Antisense RNA: the new favorite in genetic research. *J Zhejiang Univ Sci B* 2018;**19**:739.
7. Amodio N, Raimondi L, Juli G *et al.* MALAT1: a druggable long non-coding RNA for targeted anti-cancer approaches. *J Hematol Oncol* 2018;**11**:1–19.
8. Flores JK, Ataide SF. Structural changes of RNA in complex with proteins in the SRP. *Front Mol Biosci* 2018;**5**:1–8. https://doi.org/10.3389/fmolb.2018.00007.
9. Villamizar O, Chambers CB, Riberdy JM *et al.* Long noncoding RNA Saf and splicing factor 45 increase soluble Fas and resistance to apoptosis. *Oncotarget* 2016;**7**:13810–26.
10. Stefanov SR, Meyer IM. Deciphering the universe of RNA structures and trans RNA–RNA interactions of transcriptomes *in vivo*: from experimental protocols to computational analyses. *Syst Biol*. 2018, 173–216. https://doi.org/10.1007/978-3-319-92967-5_9
11. Singh S, Shyamal S, Panda AC. Detecting RNA–RNA interactome. *Wiley Interdiscip Rev RNA* 2022;**13**:e1715. https://doi.org/10.1002/wrna.1715
12. Gong J, Shao D, Xu K *et al.* RISE: a database of RNA interactome from sequencing experiments. *Nucleic Acids Res* 2018;**46**:D194–201.
13. Kudla G, Wan Y, Helwak A. RNA conformation capture by proximity ligation. *Annu Rev Genom Hum Genet* 2020;**21**:81–100. https://doi.org/10.1146/annurev-genom-120219-073756
14. Aw JGA, Shen Y, Wilm A *et al. In vivo* mapping of eukaryotic RNA interactomes reveals principles of higher-order organization and regulation. *Mol Cell* 2016;**62**:603–17. https://doi.org/10.1016/j.molcel.2016.04.028
15. Lu Z, Zhang QC, Lee B *et al.* RNA duplex map in living cells reveals higher-order transcriptome structure. *Cell* 2016;**165**:1267–79. https://doi.org/10.1016/j.cell.2016.04.028
16. Sharma E, Sterne-Weiler T, O'Hanlon D *et al.* Global mapping of human RNA–RNA interactions. *Mol Cell* 2016;**62**:618–26. https://doi.org/10.1016/j.molcel.2016.04.030
17. Helwak A, Tollervey D. Mapping the miRNA interactome by cross-linking ligation and sequencing of hybrids (CLASH). *Nat Protoc* 2014;**9**:711–28.
18. Nguyen TC, Cao X, Yu P *et al.* Mapping RNA–RNA interactome and RNA structure *in vivo* by MARIO. *Nat Commun* 2016;**7**:12023. https://doi.org/10.1038/ncomms12023
19. Melamed S, Faigenbaum-Romm R, Peer A *et al.* Mapping the small RNA interactome in bacteria using RIL-seq. *Nat Protoc* 2018;**13**:1–33.
20. Cai Z, Cao C, Ji L *et al.* RIC-seq for global in situ profiling of RNA–RNA spatial interactions. *Nature* 2020;**582**:432–37. https://doi.org/10.1038/s41586-020-2249-1
21. Gong J, Ju Y, Shao D *et al.* Advances and challenges towards the study of RNA–RNA interactions in a transcriptome-wide scale. *Quant Biol* 2016;**6**:239–52. https://doi.org/10.1007/s40484-018-0146-5
22. Zhang M, Li K, Bai J *et al.* Optimized photochemistry enables efficient analysis of dynamic RNA structuromes and interactomes in genetic and infectious diseases. *Nat Commun* 2021;**12**:2344. https://doi.org/10.1038/s41467-021-22552-y
23. Videm P, Kumar A, Zharkov O *et al.* ChiRA: an integrated framework for chimeric read analysis from RNA–RNA interactome and RNA structurome data. *Gigascience* 2021;**10**:giaa158. https://doi.org/10.1093/gigascience/giaa158
24. Schäfer RA, Voß B. RNAnue: efficient data analysis for RNA–RNA interactomics. *Nucleic Acids Res* 2021;**49**:5493–501.
25. Zhang M, Hwang IT, Li K *et al.* Classification and clustering of RNA crosslink-ligation data reveal complex structures and homodimers. *Genome Res* 2022;**32**:968–85.
26. Hoffmann S, Otto C, Doose G *et al.* A multi-split mapping algorithm for circular RNA, splicing, trans-splicing and fusion detection. *Genome Biol* 2014;**15**:R34. https://doi.org/10.1186/gb-2014-15-2-r34

27. Zhong C, Zhang S. Accurate and efficient mapping of the cross-linked microRNA–mRNA duplex reads. *Iscience* 2019;**18**:11–9.
28. Li H. Aligning sequence reads, clone sequences and assembly contigs with BWA-MEM. arXiv, https://doi.org/10.48550/arXiv.1303.3997, 16 March 2013, preprint: not peer reviewed.
29. Dobin A, Davis CA, Schlesinger F *et al.* STAR: ultrafast universal RNA-seq aligner. *Bioinformatics* 2013;**29**:15–21.
30. Margasyuk SD, Vlasenok MA, Li G *et al.* RNAcontacts: a pipeline for predicting contacts from RNA proximity ligation assays. *Acta Naturae* 2023;**15**:51–7. https://doi.org/10.32607/actanaturae.11893
31. Hiltemann S, Rasche H, Gladman S *et al.* Galaxy Training: a powerful framework for teaching! *PLoS Comput Biol* 2023;**19**:e1010752. https://doi.org/10.1371/journal.pcbi.1010752
32. Mölder F, Jablonski KP, Letcher B *et al.* Sustainable data analysis with Snakemake. *F1000Research* 2021;**10**:33.
33. Liu Y, Mi Y, Mueller T *et al.* Multi-omic measurements of heterogeneity in HeLa cells across laboratories. *Nat Biotechnol* 2019;**37**:314–22.
34. Quinlan AR, Hall IM. BEDTools: a flexible suite of utilities for comparing genomic features. *Bioinformatics* 2010;**26**:841–42.
35. Csardi G, Nepusz T. The igraph software package for complex network research. *InterJournal* 2006;**Complex Systems**:1695.
36. Lawrence M, Huber W, Pagès H *et al.* Software for computing and annotating genomic ranges. *PLoS Comput Biol* 2013;**9**:e1003118.
37. Lorenz R, Bernhart SH, Höner zu Siederdissen C *et al.* ViennaRNA Package 2.0. *Algorithms Mol Biol* 2011;**6**:1–14.
38. Lun AT, Perry M, Ing-Simmons E. Infrastructure for genomic interactions: Bioconductor classes for Hi-C, ChIA-PET and related experiments. *F1000Research* 2016;**5**:950.
39. Hahne F, Ivanek R. Visualizing genomic data using Gviz and Bioconductor. *Methods Mol Biol* 2016;**1418**:335–51.
40. Mann M, Wright PR, Backofen R. IntaRNA 2.0: enhanced and customizable prediction of RNA–RNA interactions. *Nucleic Acids Res* 2017;**45**:W435–9.
41. Bergeron D, Paraqindes H, Fafard-Couture É *et al.* snoDB 2.0: an enhanced interactive database, specializing in human snoRNAs. *Nucleic Acids Res* 2023;**51**:D291–6.
42. Huang HY, Lin YCD, Li J *et al.* miRTarBase 2020: updates to the experimentally validated microRNA–target interaction database. *Nucleic Acids Res* 2020;**48**:D148–54.
43. Lin Y, Liu T, Cui T *et al.* RNAInter in 2020: RNA interactome repository with increased coverage and annotation. *Nucleic Acids Res* 2020;**48**:D189–97.
44. Uhrig S, Ellermann J, Walther T *et al.* Accurate and efficient detection of gene fusions from RNA sequencing data. *Genome Res* 2021;**31**:448–60.
45. Kent WJ, Sugnet CW, Furey TS *et al.* The human genome browser at UCSC. *Genome Res* 2002;**12**:996–1006.
46. Gentleman RC, Carey VJ, Bates DM *et al.* Bioconductor: open software development for computational biology and bioinformatics. *Genome Biol* 2004;**5**:1–16.
47. European Organization For Nuclear Research OpenAIRE. Zenodo. 2013. https://doi.org/10.25495/7GXK-RD71

Received: January 29, 2024. **Revised:** March 7, 2025. **Editorial Decision:** March 11, 2025. **Accepted:** March 31, 2025
© The Author(s) 2025. Published by Oxford University Press on behalf of Nucleic Acids Research.
This is an Open Access article distributed under the terms of the Creative Commons Attribution-NonCommercial License (https://creativecommons.org/licenses/by-nc/4.0/), which permits non-commercial re-use, distribution, and reproduction in any medium, provided the original work is properly cited. For commercial re-use, please contact reprints@oup.com for reprints and translation rights for reprints. All other permissions can be obtained through our RightsLink service via the Permissions link on the article page on our site—for further information please contact journals.permissions@oup.com.