



7 **Abstract**

8 Transcripts are frequently modified by structural variations, which leads to a fused transcript of either  
9 multiple genes (known as a fusion gene) or a gene and a previously non-transcribing sequence. Detecting  
10 these modifications (called transcriptomic structural variations, or TSVs), especially in cancer tumor  
11 sequencing, is an important and challenging computational problem. We introduce SQUID, a novel  
12 algorithm to accurately predict both fusion-gene and non-fusion-gene TSVs from RNA-seq alignments.  
13 SQUID unifies both concordant and discordant read alignments into one model, and doubles the accuracy  
14 on simulation data compared to other approaches. With SQUID, we identified novel non-fusion-gene  
15 TSVs on TCGA samples.

16 *Keywords:* transcriptomic structural variation, RNA-seq, TCGA

17 **1 Background**

18 Large-scale transcriptome sequence changes are known to be associated with cancer [1, 2]. Those changes  
19 are usually a consequence of genomic structural variation (SV). By pulling different genomic regions to-  
20 gether or separating one region into pieces, structural variants can potentially cause severe alteration to  
21 transcribed or translated products. Transcriptome changes induced by genomic SVs, called transcriptomic  
22 structural variants (TSVs), can have a particularly large impact on disease genesis and progression. In some  
23 cases, TSVs bring regions from one gene next to regions of another, causing exons from both genes to be  
24 transcribed into a single transcript (known as a fusion gene). Domains of the corresponding RNA or pro-  
25 teins can be fused, inducing new functions or causing loss of function, or the transcription or translation  
26 levels can be altered, leading to disease states. For example, *BCR-ABL1* is a well-known fusion oncogene  
27 for chronic myeloid leukemia [3], and the *TMPRSS2-ERG* fusion product leads to over-expression of *ERG*  
28 and helps triggers prostate cancer [4]. These fusion events are used as biomarkers for early diagnosis or  
29 treatment targets [5]. In other cases, TSVs can affect genes by causing a previously non-transcribed region  
30 to be incorporated into a gene, causing disruption to the function of the altered gene. There are fewer studies  
31 on these TSVs between transcribed and non-transcribed regions, but their ability to alter downstream RNA  
32 and protein structure is likely to lead to similar results as fusion gene TSVs.

33 Genomic SVs are typically detected from whole-genome sequencing (WGS) data by identifying reads and  
34 read pairs that are incompatible with a reference genome [e.g., 6–11]. However, WGS data are not com-

35 pletely suitable to infer TSVs since they neither inform which region is transcribed nor reveal how the tran-  
36 scribed sequence will change if SVs alter a splicing site or the stop codon. In addition, WGS data is more  
37 scarce and more expensive to obtain than RNA-seq [12] measurements, which target transcribed regions  
38 directly. RNA-seq is relatively inexpensive, high-throughput, and widely available in many existing and  
39 growing data repositories. For example, The Cancer Genome Atlas (TCGA, <https://cancergenome.nih.gov>)  
40 contains RNA-seq measurements from thousands of tumor sample across various cancer types, but 80% of  
41 tumor samples in TCGA have RNA-seq data but no WGS data (Supplementary Figure S1). While methods  
42 exist to detect fusion genes from RNA-seq measurements [e.g., 13–21], fusion genes are only a subset of  
43 TSVs, and existing fusion gene detection methods rely heavily on current gene annotations and are generally  
44 not able or at least not optimized to predict non-fusion-gene TSV events. The idea of de novo transcript as-  
45 sembly [e.g., 22–25] followed by transcript-to-genome alignment [e.g., 26–28] is used in some fusion-gene  
46 detection methods. These approaches rely on annotation-based filtering steps to achieve the high accuracy.  
47 Although it is possible to extend these approaches to non-fusion-gene TSV detection, the lack of annotation  
48 information for non-transcribing regions makes these approaches less suitable for finding non-fusion-gene  
49 TSV. This motivates the need for a method to detect both types of TSVs directly from RNA-seq data.

50 We present SQUID, the first computational tool that is designed to comprehensively predict both types  
51 of TSVs from RNA-seq data. SQUID divides the reference genome into segments and builds a genome  
52 segment graph from both concordant and discordant RNA-seq read alignments. In this way, it can detect  
53 both fusion-gene events and TSVs incorporating previously non-transcribed regions into transcripts. Using  
54 an efficient, novel integer linear program (ILP), SQUID rearranges the segments of the reference genome so  
55 that as many read alignments as possible are concordant with the rearranged sequence. TSVs are represented  
56 by pairs of breakpoints realized by the rearrangement. Discordant reads that cannot be made concordant  
57 through the optimal rearrangement given by the ILP are discarded as false positive discordant reads, likely  
58 due to misalignments. By building a consistent model of the entire rearranged genome and maximizing the  
59 number of overall concordant read alignments, SQUID drastically reduces the number of spurious TSVs  
60 reported compared with other methods.

61 SQUID features high accuracy. SQUID is usually  $> 20\%$  more accurate than applying WGS-based SV  
62 detection methods to RNA-seq data directly. It is similarly more accurate than the pipeline that uses de novo  
63 transcript assembly and transcript-to-genome alignment to detect TSVs. We also show that SQUID is able

64 to detect more TSVs involving non-transcribed regions than any existing fusion gene detection method.

65 We use SQUID to detect TSVs within 401 TCGA tumor samples of four cancer types (99–101 samples each  
66 of breast invasive carcinoma [29], bladder urothelial carcinoma [30], lung adenocarcinoma [31], and prostate  
67 adenocarcinoma [32]). SQUID’s predictions suggest that breast invasive carcinoma has more samples with a  
68 larger or smaller number of TSVs / non-fusion-gene TSVs than other cancer types. We also characterize the  
69 differences between fusion-gene TSVs and non-fusion-gene TSVs. Non-fusion-gene TSVs, for example,  
70 are more likely to be intra-chromosomal events. We show that breakpoints can occur in multiple samples,  
71 and among those that do repeatedly occur, their breakpoint partners are also often conserved. Finally, we  
72 identify several novel non-fusion-gene TSVs that affect known tumor suppressor genes, which may result  
73 in loss-of-function of corresponding proteins and play a role in tumor genesis.

## 74 **2 Results**

### 75 **2.1 A novel algorithm for detecting TSVs from RNA-seq**

76 SQUID predicts TSVs from RNA-seq alignments to the genome (Figure 1 provides an overview). To do this,  
77 it seeks to rearrange the reference genome to make as many of the observed alignments consistent with the  
78 rearranged genome as possible. Formally, SQUID constructs a graph from the alignments where the nodes  
79 represent boundaries of genome segments and the edges represent adjacencies implied by the alignments.  
80 These edges represent both concordant and discordant alignments, where concordant alignments are those  
81 consistent with the reference genome and discordant alignments are those that are not. SQUID then uses  
82 a novel integer linear program (Section 4.2) to order and orient the vertices of the graph to make as many  
83 edges consistent as possible. Adjacencies that are present in this rearranged genome but not present in  
84 the original reference are proposed as predicted TSVs. The identification of concordant and discordant  
85 alignments (Section 4.3), construction of the genome segments (Section 4.4), creation of the graph, and the  
86 reordering objective function (Section 4.1) are described in the Methods section.

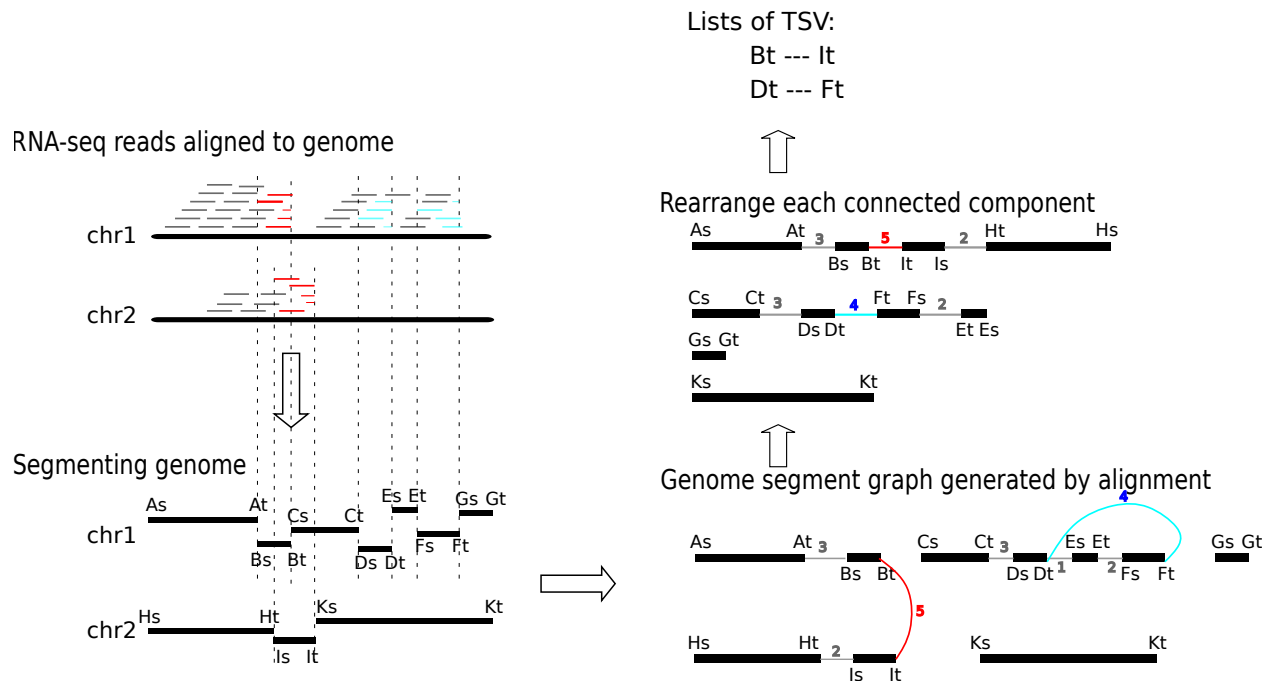


Figure 1: Overview of the SQUID algorithm. Based on the alignments of RNA-seq reads to the reference genome, SQUID partitions the genome into segments, connects the endpoints of the segments to indicate the actual adjacency in transcript, and finally reorders the endpoints along the most reliable path. Each edge in the final path that comes from discordant read alignments represents a TSV.

## 87 2.2 SQUID is accurate on simulation data

88 Overall, SQUID's predictions of TSVs are far more precise than other approaches at similar sensitivity on  
 89 simulated data (Section 4.7). SQUID achieves 60% to 80% percent precision and about 50% percent sensi-  
 90 tivity on simulation data (Figure 2). SQUID's precision is > 20% higher than several de novo transcriptome  
 91 assembly and transcript-to-genome alignment pipelines (for details see Supplementary Text), and the pre-  
 92 cision of WGS-based SV detection methods on RNA-seq data is even lower. The sensitivity of SQUID is  
 93 similar to de novo assembly with MUMmer3 [26], but a little lower than DELLY2 [6] and LUMPY [7] with  
 94 SpeedSeq [33] aligner. The overall sensitivity is not as high as precision, which is probably because there  
 95 are not enough supporting reads aligned correctly to some TSV breakpoints. The fact that assembly and  
 96 WGS-based SV detection methods achieve similar sensitivity corroborates the hypothesis that it is the data  
 97 limiting the achievable sensitivity.

98 The low specificity of the pipeline- and WGS-based methods shows neither of these types of approaches  
 99 are suitable for TSV detection from RNA-seq data. WGS-based SV detection methods are able to detect

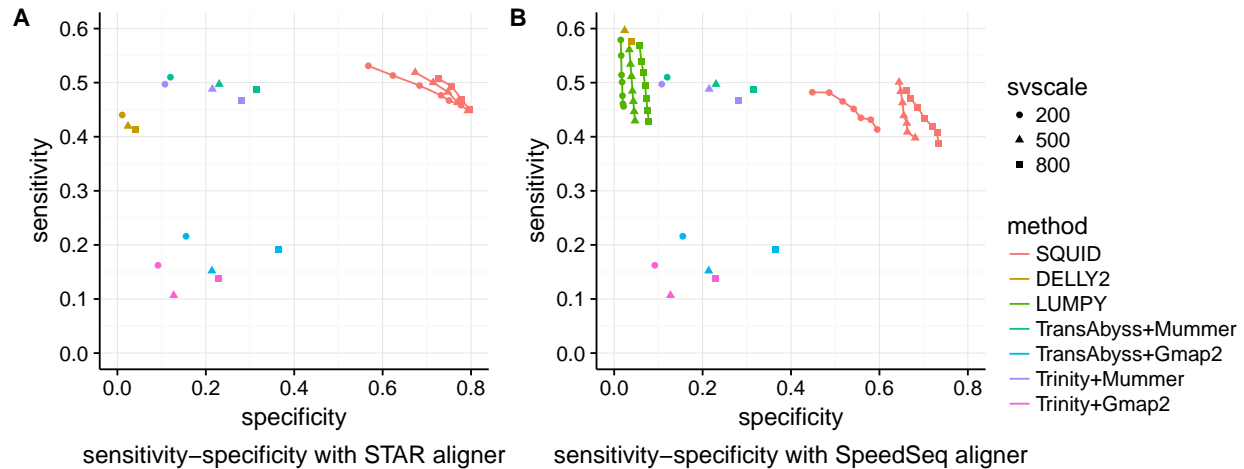


Figure 2: Performance of SQUID and other methods on simulation data. Different number of SVs (200, 500, 800 SVs) are simulated in each dataset. Each simulated read is aligned with both (A) STAR and (B) SpeedSeq aligner. If the method allows for user-defined minimum read support for prediction, we vary the threshold from 3 to 9, and plot a curve on sensitivity-specificity curve (SQUID and LUMPY), otherwise it is shown as a single point

100 TSV signals, but not able to filter out false positives. Assembly-based approaches require solving the tran-  
101 scriptome assembly problem which is a harder and more time-consuming problem, and thus errors are more  
102 easily introduced. Further, the performance of assembly pipelines depends heavily on the choice of software  
103 — for example, MUMmer3 [26] is better at discordantly aligning transcripts than GMAP [27]. Dissect [28]  
104 is another transcript-to-genome alignment method that is designed for the case where SVs exist. (Unfortu-  
105 nately, Dissect did not run to completion on the some of the dataset tested here.) It is possible that different  
106 combinations of de novo transcript assembly and transcript-to-genome alignment tools can improve the  
107 accuracy of the pipelines, but optimizing the pipeline is out of scope of this work.

108 SQUID's effectiveness is likely due to its unified model of both concordant reads and discordant reads.  
109 Coverage in RNA-seq alignment is proportional to the expression level of the transcript, and using one  
110 read count threshold for TSV evidence is not appropriate. Instead, the ILP in SQUID sets concordant and  
111 discordant alignments into competition and selects the winner as the most reliable TSVs.

### 112 2.3 SQUID is able to detect non-fusion-gene TSV on two previously-studied cell lines

113 Fusion gene events are a strict subset of TSVs where the two breakpoints are each within a gene region and  
114 the fused sequence corresponds to the sense strand of both genes. Fusion genes thus exclude TSV events

115 where a gene region is fused with a intergenic region or an anti-sense strand of another gene. Nevertheless,  
116 fusion genes have been implicated (likely because of available methods to detect them) in playing a role in  
117 cancer.

118 To probe SQUID's ability to detect TSVs from real data, we use two cell lines, HCC1954 and HCC1395, for  
119 which previous studies have experimentally validated predicted SVs and fusion gene events. Specifically,  
120 we compile results from Bignell et al. [34], Stephens et al. [35], Galante et al. [36], Zhao et al. [37] and  
121 Robinson et al. [38] for HCC1954, and results from Stephens et al. [35] and Zhang et al. [13] for HCC1395.  
122 After removing short deletions and overlapping structural variations among different studies, we have 326  
123 validated structural variations for HCC1954 cell line, in which 245 of them have at least one breakpoint  
124 outside a gene region, and the rest (81) have both breakpoints within gene region; we have 256 validated  
125 true structural variations for HCC1395 cell line, in which 94 have at least one breakpoint outside a gene  
126 region, while the rest (162) have both breakpoints within gene. For a predicted structural variation to be  
127 true positive, both predicted breakpoints should be within a window of 30kb of true breakpoints and the  
128 predicted orientation should agree with the true orientation. We use a relatively large window since the true  
129 breakpoints can be located within an intron or other non-transcribed region, while the observed breakpoint  
130 from RNA-seq reads will be at a nearby coding or expressed region.

131 We use publicly available RNA-seq data from the NIH Sequencing Read Archive (SRA; accessions:  
132 SRR2532344 and SRR925710 for HCC1954, SRR2532336 for HCC1395). Because the data are from a  
133 pool of experiments, the sample from which RNA-seq was collected may be different from those used for  
134 experimental validation. We align reads to the reference genome using STAR. We compare the result with  
135 the top fusion-gene detection tools evaluated in Liu et al. [39] and newer software not evaluated by Liu et al.  
136 [39], specifically, SOAPfuse [20], deFuse [14], FusionCatcher [16], JAFFA [15] and INTEGRATE [15].

137 When restricted to fusion gene events, SQUID achieves similar precision and sensitivity compared to fusion  
138 gene detection tools (Figure 3A). SQUID has the highest accuracy in the HCC1954 cell line, with very  
139 similar sensitivity as all fusion gene detection tools. For HCC1395, SQUID is in the middle of fusion gene  
140 detection methods, while INTEGRATE [13] and JAFFA [15] are the best performers on this sample.

141 It is even harder to predict non-fusion-gene TSVs accurately, since current annotations cannot be used to  
142 limit the search space for potential read alignments or TSV events. Only SQUID and deFuse are able to

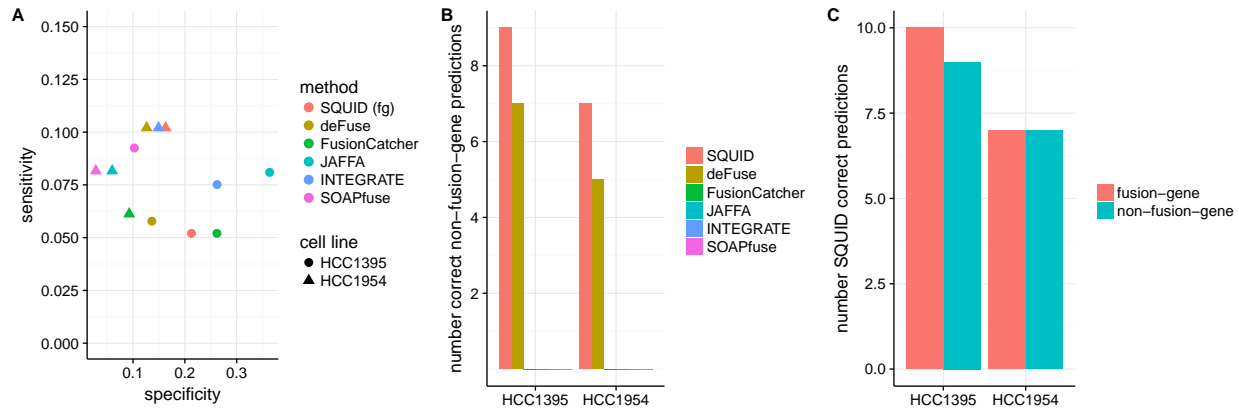


Figure 3: Performance of SQUID and fusion gene detection methods on breast cancer cell lines HCC1954 and HCC1395. Predictions are evaluated by previously validated SVs and fusions. (A) Sensitivity-specificity of different methods for predicting fusion gene events on both cell lines. (B) Number of correct non-fusion-gene TSV predictions that correspond to previously validated SVs. (C) Number of correctly predicted fusion-gene TSVs and non-fusion-gene TSVs from SQUID. Non-fusion-gene TSVs makes up a considerable proportion of all TSVs.

143 detect non-fusion-gene events. Between these two methods, SQUID is able to predict more known non-  
144 fusion-gene TSVs correctly (Figure 3B). At the same time, the precision of SQUID does not decrease  
145 very much by considering both fusion-gene and non-fusion-gene TSVs (HCC1954: fusion gene specificity  
146 is 16.28%, and overall specificity is 15.56%; HCC1395: fusion gene specificity is 21.28%, and overall  
147 specificity is 19.39%). A considerable proportion of validated TSVs are non-fusion-gene TSVs: correctly  
148 predicted non-fusion-gene TSVs compose almost half of all correct predictions of SQUID (Figure 3C).

#### 149 2.4 Charactering TSVs on four types of TCGA cancer samples

150 To compare the distributions and characteristics of TSVs among cancer types and between TSV types, we  
151 applied SQUID on arbitrarily selected 99 to 101 tumor samples from TCGA for each of four cancer types:  
152 breast invasive carcinoma (BRCA), bladder urothelial carcinoma (BLCA), lung adenocarcinoma (LUAD),  
153 and prostate adenocarcinoma (PRAD). (for details see Supplementary Text)

154 To estimate the accuracy of SQUID's prediction on selected TCGA samples, we use WGS data of the  
155 same patients to validate TSV junctions. There are in total 72 WGS experiments available for the 400  
156 samples (20 BLCA, 10 BRCA, 31 LUAD, 11 PRAD). For each TSV prediction, we extract a 25Kb sequence  
157 around both breakpoints and concatenate them according to the predicted TSV orientation. We then map the  
158 WGS reads against these junction sequences using SpeedSeq [33]. If a paired-end WGS read can only be



159 mapped concordantly to a junction sequence but not the reference genome, that paired-end read is marked  
160 as supporting the TSV. If at least 3 WGS reads support a TSV, the TSV is considered as validated. Using  
161 this approach, SQUID's overall validation rate is 88.21%, and this indicates that SQUID is quite accurate  
162 and reliable on TCGA data.

163 We find that most samples have  $\approx 15$ – $20$  TSVs including  $\approx 3$ – $5$  non-fusion-gene TSVs among all four  
164 cancer types (Figure 4A,B). BRCA has a longer tail on both sides of the distribution of TSV counts, where  
165 more samples contain a larger number of TSVs, and more samples contains a smaller number of TSVs. The  
166 same trend is observed when restricted to non-fusion-gene TSVs.

167 Inter-chromosomal TSVs are more prevalent than intra-chromosomal TSVs for all cancer types (Figure 4C),  
168 although this difference is much more pronounced in bladder and prostate cancer. Non-fusion-gene TSVs  
169 are more likely to have intra-chromosomal events than fusion gene TSVs (Figure 4D), and in fact in  
170 bladder, breast, and lung cancer, we detect more intra-chromosomal non-fusion-gene TSVs than inter-  
171 chromosomal non-fusion-gene TSVs. Prostate cancer is an exception in that, for non-fusion-gene TSVs,  
172 inter-chromosomal events are observed more often than intra-chromosomal events. Nevertheless, it also  
173 holds true that non-fusion-gene TSVs are more likely to be intra-chromosomal than fusion-gene, because  
174 the percentage of intra-chromosomal TSVs within non-fusion-gene TSVs is higher than that within all TSVs.

175 For a large proportion of breakpoints occurring multiple times within a cancer type, their partner in the TSV  
176 is likely to be fixed and to reoccur every time that breakpoint is used. To quantify this, for each breakpoint  
177 that occurred  $\geq 3$  times, we compute the entropy of its partner promiscuity. Specifically, we derive a  
178 discrete, empirical probability distribution of partners for each breakpoint and compute the entropy of this  
179 distribution. This measure thus represents the uncertainty of the partner given one breakpoint, with higher  
180 entropy corresponding to a less conserved partnering pattern. In Figure 4E, we see that there there is a high  
181 peak near 0 for all cancer types, which indicates that for a large proportion of recurring breakpoints, we  
182 are certain about its rejoined partner once we know the breakpoint. However, there are also promiscuous  
183 breakpoints with entropy larger than 0.5.

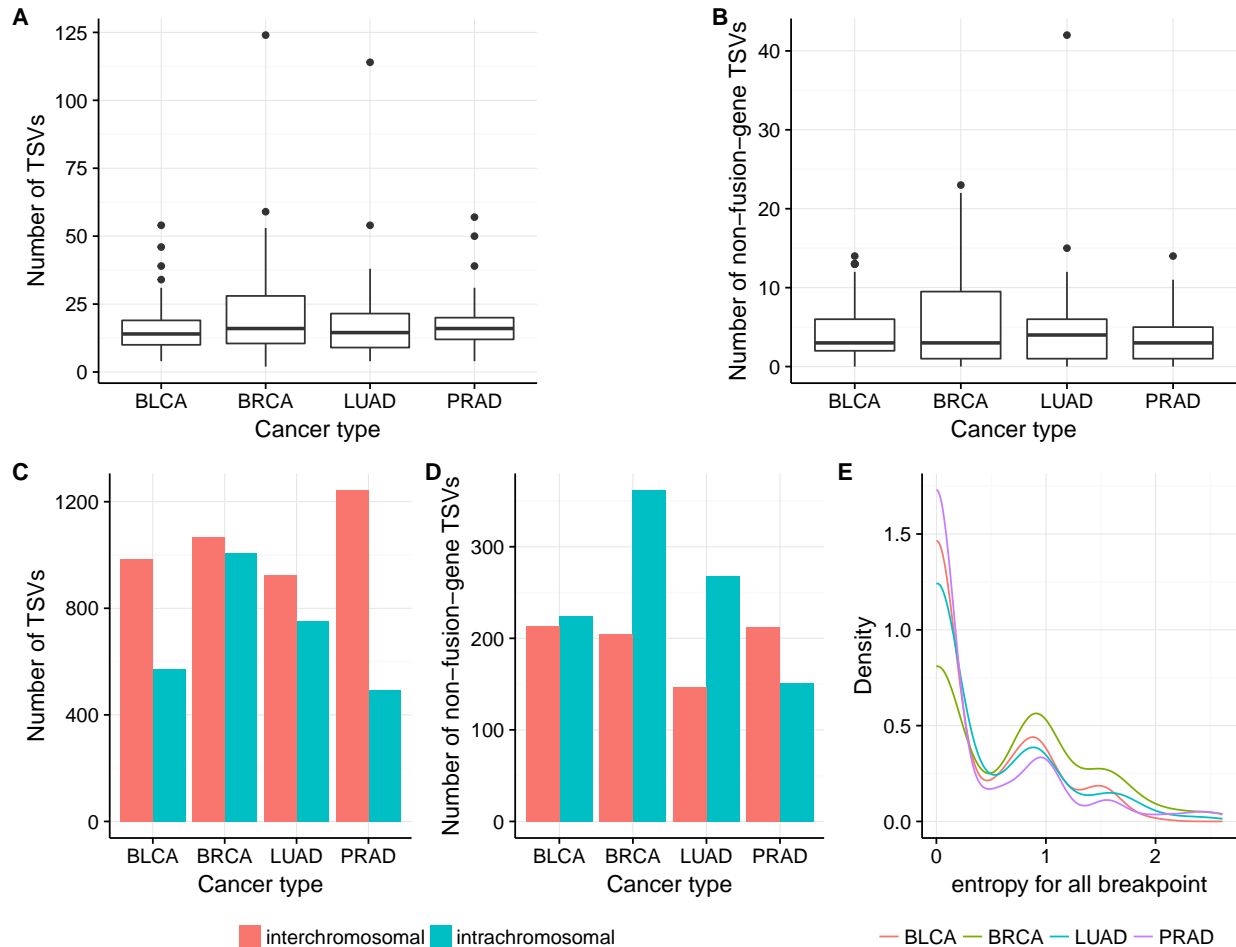


Figure 4: (A,B) Number of TSVs and non-fusion-gene TSVs in each sample in different cancer types. BRCA has slightly more samples with larger or smaller number of (non-fusion-gene) TSVs, thus showing a longer tail on both ends of y axis. (C,D) Number of inter-chromosomal and intra-chromosomal TSVs within all TSVs and within non-fusion-gene TSVs. Non-fusion-gene TSVs contain more intra-chromosomal events than fusion-gene TSVs. (E) For breakpoints occurring more than 3 times in the same cancer type, the distribution of the entropy of its TSV partner. The lower the entropy, the more likely the breakpoint has a fixed partner. The peak near 0 indicates a large portion of breakpoints are likely to be rejoined with the same partner in TSV. However, there are still some breakpoints that have multiple rejoined partners.

## 184 **2.5 Tumor suppressor genes can undergo TSV and generate altered transcripts**

185 Tumor suppressor genes (TSG) protect cells from becoming cancer cells. Usually their functions involve  
186 inhibiting cell cycle, facilitating apoptosis, and so on [40]. Mutations in TSGs may lead to loss of function of  
187 the corresponding proteins and benefit tumor growth. For example, homozygous loss-of-function mutation  
188 in p53 is found in about half of cancer samples across various cancer types [41]. TSVs are likely to cause  
189 loss of function of TSGs as well. Indeed, we observe several TSGs that are affected by TSVs, both of the  
190 fusion-gene type and the non-fusion-gene type.

191 The *ZFHX3* gene encodes a transcription factor that transactivates cyclin-dependent kinase inhibitor 1A  
192 (aka *CDKN1A*), a cell cycle inhibitor [42]. We find that in one BLCA and one BRCA sample, there are  
193 TSVs affecting *ZFHX3*. These two TSVs events are different from each other in terms of the breakpoint  
194 partner outside of *ZFHX3*. In the BLCA tumor sample, a intergenic region is inserted after the third exon of  
195 *ZFHX3* (Figure 5A). The fused transcript stops at the inserted region, causing the *ZFHX3* transcript to lose  
196 the rest of its exons. In the BRCA tumor sample, a region of the anti-sense strand of gene *MYLK3* is inserted  
197 after the third exon of *ZFHX3* gene (Figure 5B). Because codons and splicing sites are not preserved on the  
198 anti-sense strand, the transcribed insertion region does not correspond to known exons of *MYLK3* gene, but  
199 covers the range of first exon of *MYLK3* and extend to the first intron and 5' intergenic region. Transcription  
200 stops within inserted region, and causes the *ZFHX3* transcript to lose exons after exon 3, which resembles  
201 the fusion with intergenic region in BLCA sample.

202 Another example is given by the *ASXLI* gene, which is essential for activating *CDKN2B* to inhibit tumor-  
203 genesis [43]. We observe two distinct TSVs related to *ASXLI* from BLCA and BRCA samples. The first  
204 TSV merges the first 11 exons and half of exon 12 of *ASXLI* with a intergenic region on chromosome 4  
205 (Figure 5C). Transcription stops at the inserted intergenic region, leaving the rest of exon 12 not transcribed.  
206 The breakpoint within the *ASXLI* is before the 3' UTR, so the downstream protein sequence from exon 12  
207 will be affected. The other TSV involving *ASXLI* is a typical fusion-gene TSV where the first three exons of  
208 *ASXLI* are fused with the last three exons from the *PDRG1* gene (Figure 5D). Protein domains after *ASXLI*  
209 exon 4 and before *PDRG1* exon 2 are lost in the fused transcript.

210 These non-fusion-gene examples are novel predicted TSV events that are not typically detectable via tradi-  
211 tional fusion-gene detection methods using RNA-seq data. They suggest that non-fusion-gene events can

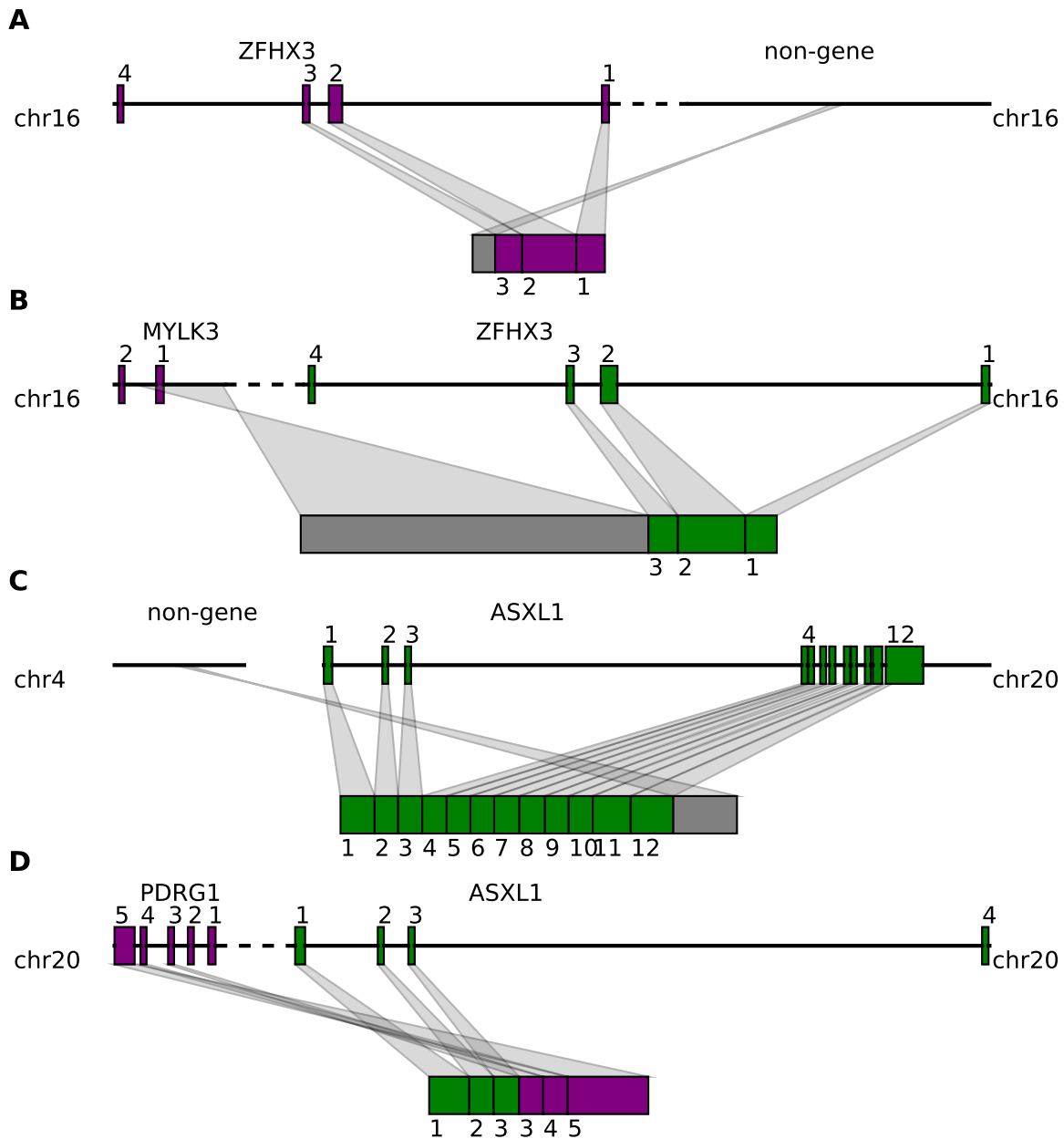


Figure 5: Tumor suppressor genes are affected by both fusion-gene and non-fusion-gene TSVs and generate transcripts with various features. (A) *ZFH3* is fused with a intergenic region after exon 3. The transcript stops at the inserted region, losing the rest of exons. (B) *ZFH3* is fused with a part of *MYLK3* anti-sense strand after exon 3. Codon and splicing signals are not preserved on anti-sense strand, thus *MYLK3* anti-sense insertion acts the same as intergenic region insertion, and causes transcription stop before reaching the rest of *ZFH3* exons. (C) *ASXL1* is fused with an intergenic region in the middle of exon 12. The resulting transcript contains a truncated *ASXL1* exon 12 and intergenic sequence. (D) The first 3 exons of *ASXL1* gene are joined with last 3 exons of *PDRG1*, resulting in a fused transcript containing 6 complete exons from both *ASXL1* and *PDRG1*.

212 also be involved in tumorigenesis by causing disruption of tumor suppressor genes.

### 213 **3 Discussion**

214 We developed SQUID, the first algorithm for accurate and comprehensive TSV detection that targets both  
215 traditional fusion-gene detection and the much broader class of general TSVs. SQUID exhibits far higher  
216 precision at similar sensitivities compared with WGS-based SV detection methods and pipelines of de novo  
217 transcriptome assembly and transcript-to-genome alignment. In addition, it has the ability to detect non-  
218 fusion-gene TSVs. These features are derived from its unique approach to predicting TSVs, whereby it  
219 constructs a consistent model of the underlying rearranged genome that explains as much of the data as pos-  
220 sible. In particular, it simultaneously considers both concordant and discordant reads, and by rearranging  
221 genome segments to maximize the number of concordant reads, SQUID generates a set of compatible TSVs  
222 that are most reliable in terms of the numbers of reads supporting them. Instead of a universal read support  
223 threshold, the objective function in SQUID naturally balances reads supporting and not supporting a candi-  
224 date TSV. This design is efficient in filtering out sequencing and alignment noise in RNA-seq, especially in  
225 the annotation-free context for predicting non-fusion-gene TSV events.

226 We use SQUID to analyze TCGA RNA-seq data of tumor samples. We identify BRCA to have a flatter  
227 distribution of number of per-sample TSVs than the other cancer types studied. We observe that non-fusion-  
228 gene TSVs are more likely to be intra-chromosomal events than fusion-gene TSVs. This is likely due to  
229 the different sequence composition features in gene vs. non-gene regions. PRAD also stands out because  
230 the percentage of inter-chromosomal TSVs is the largest. Overall, these findings continue to suggest that  
231 different cancer types have different preferred patterns of TSVs, although the question remains whether  
232 these differences will hold up as more samples are analyzed and whether the different patterns are causal,  
233 correlated, or mostly due to non-functional randomness.

234 We also use SQUID to observe both non-fusion-gene and fusion-gene TSVs involving known tumor sup-  
235 pressor genes *ZFHX3* and *ASXL1*. In these cases, transcription usually stops within the inserted region of  
236 the non-fusion-gene TSVs, which causes the TSG transcript to lose some of its exons, reasonably leading to  
237 downstream loss of function.

238 Other important uses and implications for general TSVs have yet to be explored and represent possible

239 directions for future work. TSVs will impact accuracy of transcriptome assembly and expression quantifi-  
240 cation, and methodological advancements are needed to correct those downstream analyses for the effect  
241 of TSVs. For example, current reference-based transcriptome assemblers are not able to assemble from  
242 different chromosomes to handle the case of inter-chromosomal TSVs. In addition, expression levels of  
243 TSV-affected transcripts cannot be quantified if they are not present in the transcript database. Incorporat-  
244 ing TSVs into transcriptome assembly and expression quantification can potentially improve their accuracy.  
245 SQUID's ability to provide a new genome sequence that is as consistent as possible with the observed reads  
246 will facilitate its use as a pre-processing step for transcriptome assembly and expression quantification,  
247 though optimizing this pipeline remains a task for future work.

248 Several natural directions exist for extending SQUID. First, SQUID is not able to predict small deletions,  
249 instead, it treats the small deletions the same as introns. This is to some extent a limitation of using RNA-seq  
250 data: introns and deletions are difficult to distinguish, as both result in concordant split reads or stretched  
251 mate pairs. The use of gene annotations could somewhat address this problem. Second, when the RNA-  
252 seq reads are derived from a highly heterogeneous sample, SQUID is likely not able to predict all TSVs  
253 occurring in the same region if they are conflicting since it seeks a single, consistent genome model. Instead,  
254 SQUID will only pick the dominating one that is compatible with other predicted TSVs. One approach to  
255 handle this would be to iteratively re-run SQUID, removing reads that are explained at each step. Again,  
256 this represents an attractive avenue for future work.

257 SQUID is open source and available at <http://www.github.com/Kingsford-Group/squid> and the scripts to  
258 replicate the computational experiments described here are available at <http://www.github.com/Kingsford-Group/squidtest>.

## 260 4 Methods

### 261 4.1 The computational problem: rearrangement of genome segments

262 We formulate the TSV detection problem as the optimization problem of rearranging genome segments to  
263 maximize the number of observed reads that are consistent (termed *concordant*) with the rearranged genome.  
264 This approach requires defining the genome segments that can be independently rearranged. It also requires

265 defining what reads are consistent with a particular arrangement of the segments. We will encode both of  
266 these (segments and read consistency) within a *Genome Segment Graph* (GSG). See Figure 6 as an example.

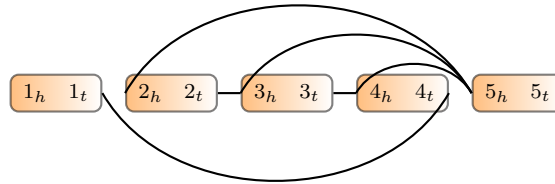


Figure 6: Example of genome segment graph. Boxes are genome segments, each of which has two ends subscripted by  $h$  and  $t$ . The color gradient indicates the orientation from head to tail. Edges connect ends of genome segments.

267 **Definition 1** (Segment). A segment is a pair  $s = (s_h, s_t)$ , where  $s$  represents a continuous sequence in  
268 reference genome and  $s_h$  represents its head and  $s_t$  represents its tail in reference genome coordinates. In  
269 practice, segments will be derived from the read locations (Section 4.4).

270 **Definition 2** (Genome Segment Graph (GSG)). A genome segment graph  $G = (V, E, w)$  is an undirected  
271 weighted graph, where  $V$  contains both endpoints of each segment in a set of segments  $S$ , i.e.,  $V = \{s_h : s \in S\} \cup \{s_t : s \in S\}$ . Thus, each vertex in the GSG represents a location in the genome. An edge  
272  $(u, v) \in E$  indicates that there is evidence that the location  $u$  is in fact adjacent to location  $v$ . Weight  
273 function,  $w : E \rightarrow \mathbb{R}^+$ , represents the reliability of an edge. Generally speaking, the weight is the number  
274 of read alignments supporting the edge, but we allow a multiplier to calculate edge weight which will be  
275 discussed below. In practice,  $E$  and  $w$  will be derived from split-aligned and paired-end reads (Section 4.5).

277 Defining vertices by endpoints of segments is required to avoid ambiguity. Only knowing that segment  $i$  is  
278 connected with segment  $j$  is not enough to recover the sequence, since different relative positions of  $i$  and  
279  $j$  spell out different sequences. Instead, for example, an edge  $(i_t, j_h)$  indicates that the tail of segment  $i$  is  
280 connected head of segment  $j$ , and this specifies a unique desired local sequence with only another possibility  
281 of the reverse complement (i.e. it could be that the true sequence is  $i \cdot j$  or  $rev(j) \cdot rev(i)$ ; here  $\cdot$  indicates  
282 concatenation and  $rev(i)$  is the reverse complement of segment  $i$ ).

283 The GSG is similar to the breakpoint graph [44] but with critical differences. A breakpoint graph has edges  
284 representing both connections in reference genome and in target genome. While edges in the GSG only  
285 represents the target genome, and they can be either concordant or discordant. In addition, the GSG does  
286 not require that the degree of every vertex is two, and thus alternative splicing and erroneous edges can exist

287 in the GSG.

288 Our goal is to reorder and reorient the segments in  $S$  so that as many edges in  $G$  are compatible with the  
289 rearranged genome as possible.

290 **Definition 3** (Permutation). *A permutation  $\pi$  on a set of segments  $S$  projects a segment in  $S$  to a set of*  
291 *integers from 1 to  $|S|$  (the size of  $S$ ) representing the indices of the segments in an ordering of  $S$ . In other*  
292 *words, each permutation  $\pi$  defines a new order of segments in  $S$ .*

293 **Definition 4** (Orientation Function). *An orientation function  $f$  maps both ends of segments to 0 or 1:*

$$f : \{s_h : s \in S\} \cup \{s_t : s \in S\} \longrightarrow \{0, 1\}$$

294 *subject to  $f(s_h) + f(s_t) = 1$  for all  $s = (s_h, s_t) \in S$ . An orientation function specifies the orientations of*  
295 *all segments in  $S$ . Specifically,  $f(s_h) = 1$  means  $s_h$  goes first and  $s_t$  next, corresponding to forward strand*  
296 *of segment, and  $f(s_t) = 1$  corresponds to the reverse strand of the segment.*

297 With a permutation  $\pi$  and an orientation function  $f$ , the exact and unique sequence of genome is determined.  
298 The reference genome also corresponds to a permutation and an orientation function, where the permutation  
299 is the identity permutation, and the orientation function maps all  $s_h$  to 1 and all  $s_t$  to 0.

300 **Definition 5** (Edge Compatibility). *Given a set of segments  $S$ , a genome segment graph  $G = (V, E, w)$ , a*  
301 *permutation  $\pi$  on  $S$ , and an orientation function  $f$ , an edge  $e = (u_i, v_j) \in E$ , where  $u_i \in \{u_h, u_t\}$  and*  
302  *$v_j \in \{v_h, v_t\}$ , is compatible with permutation  $\pi$  and orientation  $f$  if and only if*

$$1 - f(v_j) = \mathbf{I}[\pi(v) < \pi(u)] = f(u_i) \tag{1}$$

303 *where  $\mathbf{I}[x]$  is the indicator function that is 1 if  $x$  is true and 0 otherwise. We write  $e \sim (\pi, f)$  if  $e$  is*  
304 *compatible with  $\pi$  and  $f$ .*

305 The above two edge compatibility equations (1) require that, in order for an edge to be compatible with  
306 the rearranged and reoriented sequence determined by  $\pi$  and  $f$ , the edge needs to connect the right side  
307 of the segment in front to the left side of segment following it. As we will see in Section 4.5, edges of  
308 GSG are derived from reads alignments. An edge being compatible with  $\pi$  and  $f$  is essentially equivalent to



309 the statement that the corresponding read alignments are concordant (Section 4.3) with respect to the target  
310 genome determined by  $\pi$  and  $f$ . When  $(\pi, f)$  is clear, we refer to edges that are compatible as concordant  
311 edges, and edges that are incompatible as discordant edges.

312 With the above definitions, we formulate an optimization problem as follows:

313 **Problem 1. Input:** A set of segments  $S$  and a GSG  $G = (V, E, w)$ .

314 **Output:** Permutation  $\pi$  on  $S$  and orientation function  $f$  that maximizes:

$$\max_{\pi, f} \sum_{e \in E} w(e) \cdot \mathbf{I}[e \sim (\pi, f)] \quad (2)$$

315 This objective function tries to find a rearrangement of genome segments  $(\pi, f)$ , such that when aligning  
316 reads to the rearranged sequence, as many reads as possible will be aligned concordantly. This objec-  
317 tive function includes both concordant alignments and discordant alignments and sets them in competition,  
318 which will be effective in reducing false positives when tumor transcripts out-number normal transcripts.  
319 There is the possibility that some rearranged tumor transcripts are out-numbered by normal counterparts.  
320 In order to be able to detect TSV in this case, depending on the setting, we may weight discordant read  
321 alignments more than concordant read alignments. Specifically, for each discordant edge  $e$ , we multiply the  
322 weight  $w(e)$  by a constant  $\alpha$ , which represents our estimate of the ratio of normal transcripts over tumor  
323 counterparts.

324 The final TSVs are modeled as pairs of breakpoints. Denote the permutation and orientation corresponding  
325 to an optimally rearranged genome as  $(\pi^*, f^*)$  and those that correspond to reference genome as  $(\pi_0, f_0)$ .  
326 An edge  $e$  can be predicted as a TSV if  $e \sim (\pi^*, f^*)$  and  $e \not\sim (\pi_0, f_0)$ .

## 327 4.2 Integer linear programming formulation

328 We use integer linear programming (ILP) to compute an optimal solution  $(\pi^*, f^*)$  of Problem 1. To do this,  
329 we introduce the following boolean variables:

- 330 •  $x_e$ :  $x_e = 1$  if edge  $e \sim (\pi^*, f^*)$ , and  $x_e = 0$  if not.
- 331 •  $z_{uv}$ :  $z_{uv} = 1$  if segment  $u$  is before  $v$  in the permutation  $\pi^*$ , and 0 otherwise.

332 •  $y_u$ :  $y_u = 1$  if  $f^*(u_h) = 1$  for segment  $u$ .

333 With this representation, the objective function can be rewritten as

$$\max_{x_e, y_u, z_{uv}} w(e) \cdot x_e \quad (3)$$

334 We add constraints to the ILP derived from edge compatibility equations (1). Without loss of generality,  
 335 we first suppose segment  $u$  is in front of  $v$  in the reference genome, and edge  $e$  connects  $u_t$  and  $v_h$  (which  
 336 is a tail-head connection). Plugging in  $u_t$ , the first equation in (1) is equivalent to  $1 - \mathbf{1}[\pi(u) > \pi(v)] =$   
 337  $1 - f(u_t)$ , and can be rewritten as  $\mathbf{1}[\pi(u) < \pi(v)] = f(u_h) = y_u$ . Note that  $\mathbf{1}[\pi(u) < \pi(v)]$  has the  
 338 same meaning as  $z_{uv}$ ; it leads to the constraint  $z_{uv} = y_u$ . Similarly, the second equation in (1) indicates  
 339  $z_{uv} = y_v$ . Therefore,  $x_e$  can only reach 1 when  $y_u = y_v = z_{uv}$ . This is equivalent to the inequalities (4)  
 340 below. Analogously, we can write constraints for other three types of edge connections: tail-tail connec-  
 341 tions impose inequalities (5); head-head connections impose inequalities (6); head-tail connections impose  
 342 inequalities (7):

$$\begin{aligned} x_e &\leq y_u - y_v + 1 & x_e &\leq y_u - (1 - y_v) + 1 \\ x_e &\leq y_v - y_u + 1 & x_e &\leq (1 - y_v) - y_u + 1 \\ x_e &\leq y_u - z_{uv} + 1 & x_e &\leq y_u - z_{uv} + 1 \\ x_e &\leq z_{uv} - y_u + 1 & x_e &\leq z_{uv} - y_u + 1 \end{aligned} \quad (4) \quad (5)$$

$$\begin{aligned} x_e &\leq (1 - y_u) - y_v + 1 & x_e &\leq (1 - y_u) - (1 - y_v) + 1 \\ x_e &\leq y_v - (1 - y_u) + 1 & x_e &\leq (1 - y_v) - (1 - y_u) + 1 \\ x_e &\leq (1 - y_u) - z_{uv} + 1 & x_e &\leq (1 - y_u) - z_{uv} + 1 \\ x_e &\leq z_{uv} - (1 - y_u) + 1 & x_e &\leq z_{uv} - (1 - y_u) + 1 \end{aligned} \quad (6) \quad (7)$$

343 We also add constraints to enforce that  $z_{uv}$  forms a valid topological ordering. For each pair of nodes  $u$  and  
 344  $v$ , one must be in front of other, that is  $z_{uv} + z_{vu} = 1$ . In addition, for each triple of nodes,  $u$ ,  $v$  and  $w$ , they  
 345 cannot be all in front of another; one must be at the beginning of these three and one must be at the end.

346 Therefore we add  $1 \leq z_{uv} + z_{vw} + z_{wu} \leq 2$ .

347 Solving an ILP in theory takes exponential time, but in practice, solving the above ILP to rearrange genome  
348 segments is very efficient. The key is that we can solve for each connected component separately. Because  
349 the objective maximizes the sum of compatible edge weights, the best rearrangement of one connected com-  
350 ponent is independent from the rearrangement of another because by definition there are no edges between  
351 connected components.

### 352 4.3 Concordant and discordant alignments

353 Discordant alignments are alignments of reads that contradict library preparation in sequencing. Concordant  
354 alignments are alignments of reads that agree with the library preparation. Take Illumina sequencing as an  
355 example. In order for a paired-end read alignment to be concordant, one end should be aligned to the forward  
356 strand and the other to the reverse strand, and the forward strand aligning position should be in front of the  
357 reverse strand aligning position (Figure 7A). Concordant alignment traditionally used in WGS also requires  
358 that a read cannot be split and aligned to different locations. But these requirements are invalid in RNA-seq  
359 alignments because alignments of reads can be separated by an intron with unknown length.

360 We define concordance criteria separately for split-alignment and paired-end alignment. If one end of a  
361 paired-end read is split into several parts and each part is aligned to a location, the end has split-alignments.  
362 Denote the vector of the split alignments of an end to be  $R = [A_1, A_2, \dots, A_r]$  ( $r$  depends on the number  
363 of splits). Each alignment  $R[i] = A_i$  is comprised of 4 components: chromosome (Chr), alignment starting  
364 position (Spos), alignment ending position and orientation (Ori, with value either + or -). We require  
365 that the alignments  $A_i$  are sorted by their position in read. A split-aligned end  $R = [A_1, A_2, \dots, A_r]$  is  
366 concordant if all the following conditions hold:

$$\begin{aligned} A_i.Chr &= A_j.Chr && \forall i, \forall j \\ A_i.Ori &= A_j.Ori && \forall i, \forall j \\ A_i.Spos &< A_j.Spos && \text{if } A_i.Ori = + \text{ for all } i < j \\ A_i.Spos &> A_j.Spos && \text{if } A_i.Ori = - \text{ for all } i < j \end{aligned} \tag{8}$$

367 If the end is not split, but continuous aligned, the alignment automatically satisfies equation (8). Denote the

368 alignments of  $R$ 's mate as  $M = [B_1, B_2, \dots, B_m]$ . An alignment of the paired-end read is concordant if  
369 the following conditions all hold:

$$\begin{aligned} A_i.Chr &= B_j.Chr && \forall i, \forall j \\ A_i.Ori &\neq B_j.Ori && \forall i, \forall j \\ A_1.Spos &< B_m.Spos && \text{if } A_1.Ori = + \\ A_m.Spos &> B_1.Spos && \text{if } A_1.Ori = - \end{aligned} \tag{9}$$

370 We only require the left-most split of the forward read  $R$  be in front of the left-most split of the reverse read  
371  $M$  since the two ends in a read pair may overlap. In order for a paired-end read to be concordant, each  
372 end should satisfy split-read alignment concordance (8), and the pair should satisfy paired-end alignment  
373 concordance (9).

#### 374 4.4 Splitting the genome into segments $S$

375 We use a set of breakpoints to partition the genome. The set of breakpoints contains two types of positions:  
376 (1) the start position and end position of each interval of overlapping discordant alignments, (2) an arbitrary  
377 position in each 0-coverage region.

378 Ideally, both ends of a discordant read should be located in separate segments, otherwise, the discordant  
379 read contained in a single segment will always be discordant no matter how the segments are rearranged.  
380 Assuming discordant read alignments of each TSV pile up around the breakpoints and do not overlap with  
381 discordant alignments of other TSVs, we set a breakpoint on the start and end positions of each contiguous  
382 interval of overlapping discordant alignments.

383 For each segment that contains discordant read alignments, it may also contain concordant alignments that  
384 connect the segment to its adjacent segments. To avoid having all segments in GSG connected to their  
385 adjacent segments and thus creating one big connected component, we pick the starting point of each 0-  
386 coverage region as a breakpoint. By adding those breakpoint, different genes will be in separate connected  
387 components unless some discordant reads support their connection. Overall, the size of each connected  
388 component is not very large: the number of nodes generated by each gene is approximately the number of  
389 exons located in them and these gene subgraphs are connected only when there is a potential TSV between

390 them.

### 391 4.5 Defining edges in the genome segment graph

392 In a GSG, an edge is added between two vertices when there are reads supporting the connection. For each  
 393 read spanning different segments, we build an edge such that when traversing the segments along the edge,  
 394 the read is concordant with the new sequence (equations (8) and (9)). Examples of deriving an edge from a  
 395 read alignment are given in Figure 7. In this way, concordance of an alignment and compatibility of an edge  
 396 with respect to a genome sequence is equivalent.

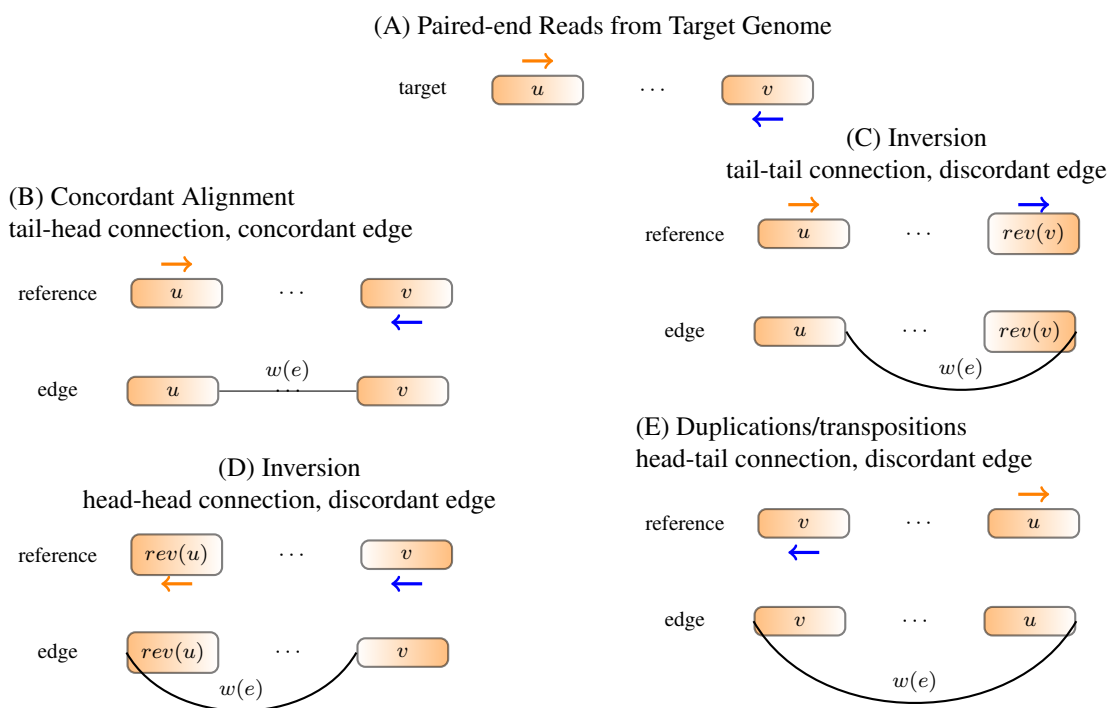


Figure 7: Constructing edges from alignment. (A) Read positions and orientations generated from the target genome. (B) If the reference genome does not have rearrangements, the read should be concordantly aligned to reference genome. An edge is added to connect the right end of  $u$  to the left end of  $v$ . Traversing the two segments along the edge reads out  $u \cdot v$ , which is the same as reference. (C) Both ends of the read align to forward strand. An edge is added to connect the right end of  $u$  to the right end of  $rev(v)$ . Traversing the segments along the edge reads out sequence  $u \cdot rev(rev(v)) = u \cdot v$ , which recovers the target sequence and the read can be concordantly aligned to. (D) If both ends align to the reverse strand, an edge is added to connect the left end of front segment to the left end of back segment. (E) If two ends of a read point out of each other, an edge is added to connect the left end of front segment to the right end of back segment.

397 The weight of a concordant edge is the number of read alignments supporting the connection, while the  
 398 weight of a discordant edge is the number of supporting alignments multiplied by discordant edge weight

399 coefficient  $\alpha$ . Edges with very low read support are likely to be a result of alignment error, therefore we filter  
400 out edges with weight lower than a threshold  $\theta$ . Segments with too many connections to other regions are  
401 likely to have low mappability, so we also filter out segments connecting to more than  $\gamma$  other segments. The  
402 parameters  $\alpha$ ,  $\theta$ , and  $\gamma$  are the most important user-defined parameters to SQUID (Supplementary Table S1  
403 and Supplementary Figure S2).

#### 404 **4.6 Identifying TSV breakpoint locations**

405 Edges that are discordant in the reference genome indicate potential rearrangements in transcripts. Among  
406 those edges, some are compatible with the permutation and orientation from ILP. These edges are taken to be  
407 the predicted TSVs. For each edge that is discordant initially but compatible with the optimal rearrangement  
408 found by the ILP, we examine the discordant read alignments to determine the exact breakpoint located  
409 within related segments. Specifically, for each end of a discordant alignment, if there are 2 other read  
410 alignments that start or end in the same position and support the same edge, then the end of the discordant  
411 alignment is predicted to be the exact TSV breakpoint. Otherwise, the boundary of the corresponding  
412 segment will be output as the exact TSV breakpoint.

#### 413 **4.7 Simulation methodology**

414 Simulations with randomly added structural variations and simulated RNA-seq reads were used to evalu-  
415 ate SQUID's performance in situations with a known correct answer. RSVsim [45] was used to simulate  
416 SV on the human genome (Ensembl 87 or hg38) [46]. We use the 5 longest chromosomes for simulation  
417 (chromosome 1 to chromosome 5). RSVsim introduces 5 different types of SVs: deletion, inversion, inser-  
418 tion, duplication, and inter-chromosomal translocation. To vary the complexity of the resulting inference  
419 problem, we simulated genomes with 200 SVs of each type, 500 SVs of each type, and 800 SVs of each  
420 type. We generated 4 replicates for each level of SV complexity (200, 500, 800). For inter-chromosomal  
421 translocations, we only simulate 2 events because only 5 chromosomes were used.

422 In the simulated genome with SVs, the original gene annotations are not applicable, and we cannot simulate  
423 gene expression from the rearranged genome. Therefore, for testing purposes, we interchange the role  
424 of the reference (hg38) and rearranged genome, and use the new genome as the reference genome for  
425 alignment, and hg38 with the original annotated gene positions as the target genome for sequencing. Flux

426 Simulator [47] was used to simulate RNA-seq reads from the hg38 genome using the Ensembl annotation  
427 version 87 [48].

428 After simulating SVs on genome, we need to transform the SVs into a set of TSVs, because not all SVs affect  
429 transcriptome, and thus not all SVs can be detected by RNA-seq. To derive the list of TSVs, we compare  
430 the positions of simulated SVs with the gene annotation. If a gene is affected by an SV, some adjacent  
431 nucleotides in the corresponding transcript may be located far part in the RSVsim-generated genome. The  
432 adjacent nucleotides can be consecutive nucleotides inside an exon if the breakpoint breaks the exon, or the  
433 end points of two adjacent exons if the breakpoint hits the intron. So for each SV that hits a gene, we find  
434 the pair of nucleotides that are adjacent in transcript and separated by the breakpoints, and convert them into  
435 coordinate of the RSVsim-generated genome, thus deriving the TSV.

436 We compare SQUID to the pipeline of de novo transcriptome assembly and transcript-to-genome alignment.  
437 We also use the same set of simulations to test whether existing WGS-based SV detection methods can be  
438 directly applied to RNA-seq data. For the de novo transcriptome assembly and transcript-to-genome align-  
439 ment pipeline, we use all combinations of the existing software Trinity [23], Trans-ABYSS [22], GMAP [27]  
440 and MUMmer3 [26]. For WGS-based SV detection methods, we test LUMPY [7] and DELLY2 [6]. We  
441 test both STAR [49] and SpeedSeq [33] (which is based on BWA-MEM [50]) to align RNA-seq reads to the  
442 genome. LUMPY is only compatible with SpeedSeq output, so we do not test it with STAR alignments.

443 **Abbreviations** ILP: integer linear programming; SV: structural variation; TSV: transcriptomic structural  
444 variation; TCGA: The Cancer Genome Atlas; WGS: whole genome sequencing.

445 **Acknowledgements** We thank Jacob West-Roberts for useful discussions. This research is funded in part  
446 by the Gordon and Betty Moore Foundation's Data-Driven Discovery Initiative through Grant GBMF4554  
447 to C.K., by the US National Science Foundation (CCF-1256087, CCF-1319998) and by the US National  
448 Institutes of Health (R21HG006913, R01HG007104), and by the Curci Foundation. C.K. received support  
449 as an Alfred P. Sloan Research Fellow. This project is funded, in part, under a grant (#4100070287) with the  
450 Pennsylvania Department of Health. The Department specifically disclaims responsibility for any analyses,  
451 interpretations or conclusions.

## References

- 452
- 453 [1] A Sveen, S Kilpinen, A Ruusulehto, RA Lothe, and RI Skotheim. Aberrant RNA splicing in cancer;  
454 expression changes and driver mutations of splicing factor genes. *Oncogene*, 35(19):2413–2428, 2015.
- 455 [2] Fredrik Mertens, Bertil Johansson, Thoas Fioretos, and Felix Mitelman. The emerging complexity of  
456 gene fusions in cancer. *Nature Reviews Cancer*, 15(6):371–381, 2015.
- 457 [3] Michael WN Deininger, John M Goldman, and Junia V Melo. The molecular biology of chronic  
458 myeloid leukemia. *Blood*, 96(10):3343–3356, 2000.
- 459 [4] Scott A Tomlins, Daniel R Rhodes, Sven Perner, Saravana M Dhanasekaran, Rohit Mehra, Xiao-Wei  
460 Sun, Sooryanarayana Varambally, Xuhong Cao, Joelle Tchinda, Rainer Kuefer, et al. Recurrent fusion  
461 of TMPRSS2 and ETS transcription factor genes in prostate cancer. *Science*, 310(5748):644–648,  
462 2005.
- 463 [5] Jianghua Wang, Yi Cai, Wendong Yu, Chengxi Ren, David M Spencer, and Michael Ittmann.  
464 Pleiotropic biological activities of alternatively spliced TMPRSS2/ERG fusion gene transcripts. *Cancer  
465 Research*, 68(20):8516–8524, 2008.
- 466 [6] Tobias Rausch, Thomas Zichner, Andreas Schlattl, Adrian M Stütz, Vladimir Benes, and Jan O Korbel.  
467 DELLY: structural variant discovery by integrated paired-end and split-read analysis. *Bioinformatics*,  
468 28(18):i333–i339, 2012.
- 469 [7] Ryan M Layer, Colby Chiang, Aaron R Quinlan, and Ira M Hall. LUMPY: a probabilistic framework  
470 for structural variant discovery. *Genome Biology*, 15(6):1, 2014.
- 471 [8] Ken Chen, John W Wallis, Michael D McLellan, David E Larson, Joelle M Kalicki, Craig S Pohl,  
472 Sean D McGrath, Michael C Wendl, Qunyuan Zhang, Devin P Locke, et al. BreakDancer: an algorithm  
473 for high-resolution mapping of genomic structural variation. *Nature Methods*, 6(9):677–681, 2009.
- 474 [9] Aaron R Quinlan, Royden A Clark, Svetlana Sokolova, Mitchell L Leibowitz, Yujun Zhang, Matthew E  
475 Hurles, Joshua C Mell, and Ira M Hall. Genome-wide mapping and assembly of structural variant  
476 breakpoints in the mouse genome. *Genome Research*, 20(5):623–635, 2010.
- 477 [10] Fereydoun Hormozdiari, Iman Hajirasouliha, Phuong Dao, Faraz Hach, Deniz Yorukoglu, Can Alkan,



- 478 Evan E Eichler, and S Cenk Sahinalp. Next-generation VariationHunter: combinatorial algorithms for  
479 transposon insertion discovery. *Bioinformatics*, 26(12):i350–i357, 2010.
- 480 [11] Jiali Zhuang and Zhiping Weng. Local sequence assembly reveals a high-resolution profile of somatic  
481 structural variations in 97 cancer genomes. *Nucleic Acids Research*, 43(17):8146–8156, 2015.
- 482 [12] Andrea Sboner, Xinmeng Jasmine Mu, Dov Greenbaum, Raymond K Auerbach, and Mark B Gerstein.  
483 The real cost of sequencing: higher than you think! *Genome Biology*, 12(8):125, 2011.
- 484 [13] Jin Zhang, Nicole M White, Heather K Schmidt, Robert S Fulton, Chad Tomlinson, Wesley C War-  
485 ren, Richard K Wilson, and Christopher A Maher. INTEGRATE: gene fusion discovery using whole  
486 genome and transcriptome data. *Genome Research*, 26(1):108–118, 2016.
- 487 [14] Andrew McPherson, Fereydoun Hormozdiari, Abdalnasser Zayed, Ryan Giuliany, Gavin Ha, Mark GF  
488 Sun, Malachi Griffith, Alireza Heravi Moussavi, Janine Senz, Nataliya Melnyk, et al. deFuse: an  
489 algorithm for gene fusion discovery in tumor RNA-Seq data. *PLoS Comput. Biol.*, 7(5):e1001138,  
490 2011.
- 491 [15] Nadia M Davidson, Ian J Majewski, and Alicia Oshlack. JAFFA: High sensitivity transcriptome-  
492 focused fusion gene detection. *Genome Medicine*, 7(1):43, 2015.
- 493 [16] Daniel Nicorici, Mihaela Satalan, Henrik Edgren, Sara Kangaspeska, Astrid Murumagi, Olli Kallion-  
494 iemi, Sami Virtanen, and Olavi Kilkku. FusionCatcher - a tool for finding somatic fusion genes in  
495 paired-end RNA-sequencing data. *bioRxiv*, 2014. doi: 10.1101/011650.
- 496 [17] Matthew K Iyer, Arul M Chinnaiyan, and Christopher A Maher. ChimeraScan: a tool for identifying  
497 chimeric transcription in sequencing data. *Bioinformatics*, 27(20):2903–2904, 2011.
- 498 [18] Lucas Swanson, Gordon Robertson, Karen L Mungall, Yaron S Butterfield, Readman Chiu, Richard D  
499 Corbett, T Roderick Docking, Donna Hogge, Shaun D Jackman, Richard A Moore, et al. Barna-  
500 cle: detecting and characterizing tandem duplications and fusions in transcriptome assemblies. *BMC*  
501 *genomics*, 14(1):550, 2013.
- 502 [19] Matteo Benelli, Chiara Pescucci, Giuseppina Marseglia, Marco Severgnini, Francesca Torricelli, and  
503 Alberto Magi. Discovering chimeric transcripts in paired-end RNA-seq data by using EricScript. *Bioin-*  
504 *formatics*, 28(24):3232–3239, 2012.

- 505 [20] Wenlong Jia, Kunlong Qiu, Minghui He, Pengfei Song, Quan Zhou, Feng Zhou, Yuan Yu, Dandan  
506 Zhu, Michael L Nickerson, Shengqing Wan, et al. SOAPfuse: an algorithm for identifying fusion  
507 transcripts from paired-end RNA-Seq data. *Genome Biology*, 14(2):R12, 2013.
- 508 [21] Wandaliz Torres-García, Siyuan Zheng, Andrey Sivachenko, Rahulshimham Vegesna, Qianghu Wang,  
509 Rong Yao, Michael F Berger, John N Weinstein, Gad Getz, and Roel GW Verhaak. PRADA: pipeline  
510 for RNA sequencing data analysis. *Bioinformatics*, 30(15):2224–2226, 2014.
- 511 [22] Gordon Robertson, Jacqueline Schein, Readman Chiu, Richard Corbett, Matthew Field, Shaun D Jack-  
512 man, Karen Mungall, Sam Lee, Hisanaga Mark Okada, Jenny Q Qian, et al. De novo assembly and  
513 analysis of RNA-seq data. *Nature Methods*, 7(11):909–912, 2010.
- 514 [23] Manfred G Grabherr, Brian J Haas, Moran Yassour, Joshua Z Levin, Dawn A Thompson, Ido Amit,  
515 Xian Adiconis, Lin Fan, Raktima Raychowdhury, Qiandong Zeng, et al. Trinity: reconstructing a  
516 full-length transcriptome without a genome from RNA-Seq data. *Nature Biotechnology*, 29(7):644,  
517 2011.
- 518 [24] Marcel H Schulz, Daniel R Zerbino, Martin Vingron, and Ewan Birney. Oases: robust de novo RNA-  
519 seq assembly across the dynamic range of expression levels. *Bioinformatics*, 28(8):1086–1092, 2012.
- 520 [25] Yinlong Xie, Gengxiong Wu, Jingbo Tang, Ruibang Luo, Jordan Patterson, Shanlin Liu, Weihua  
521 Huang, Guangzhu He, Shengchang Gu, Shengkang Li, et al. SOAPdenovo-Trans: de novo transcrip-  
522 tome assembly with short RNA-Seq reads. *Bioinformatics*, 30(12):1660–1666, 2014.
- 523 [26] Stefan Kurtz, Adam Phillippy, Arthur L Delcher, Michael Smoot, Martin Shumway, Corina Antonescu,  
524 and Steven L Salzberg. Versatile and open software for comparing large genomes. *Genome Biology*, 5  
525 (2):R12, 2004.
- 526 [27] Thomas D Wu and Colin K Watanabe. GMAP: a genomic mapping and alignment program for mRNA  
527 and EST sequences. *Bioinformatics*, 21(9):1859–1875, 2005.
- 528 [28] Deniz Yorukoglu, Faraz Hach, Lucas Swanson, Colin C Collins, Inanc Birol, and S Cenk Sahinalp.  
529 Dissect: detection and characterization of novel structural alterations in transcribed sequences. *Bioin-  
530 formatics*, 28(12):i179–i187, 2012.

- 531 [29] Cancer Genome Atlas Network et al. Comprehensive molecular portraits of human breast tumors.  
532 *Nature*, 490(7418):61, 2012.
- 533 [30] Cancer Genome Atlas Research Network et al. Comprehensive molecular characterization of urothelial  
534 bladder carcinoma. *Nature*, 507(7492):315–322, 2014.
- 535 [31] Cancer Genome Atlas Research Network et al. Comprehensive molecular profiling of lung adenocar-  
536 cinoma. *Nature*, 511(7511):543–550, 2014.
- 537 [32] Cancer Genome Atlas Research Network et al. The molecular taxonomy of primary prostate cancer.  
538 *Cell*, 163(4):1011–1025, 2015.
- 539 [33] Colby Chiang, Ryan M Layer, Gregory G Faust, Michael R Lindberg, David B Rose, Erik P Garrison,  
540 Gabor T Marth, Aaron R Quinlan, and Ira M Hall. SpeedSeq: ultra-fast personal genome analysis and  
541 interpretation. *Nature Methods*, 12(10):966, 2015.
- 542 [34] Graham R Bignell, Thomas Santarius, Jessica CM Pole, Adam P Butler, Janet Perry, Erin Pleasance,  
543 Chris Greenman, Andrew Menzies, Sheila Taylor, Sarah Edkins, et al. Architectures of somatic ge-  
544 nomic rearrangement in human cancer amplicons at sequence-level resolution. *Genome Research*, 17  
545 (9):1296–1303, 2007.
- 546 [35] Philip J Stephens, David J McBride, Meng-Lay Lin, Ignacio Varela, Erin D Pleasance, Jared T Simp-  
547 son, Lucy A Stebbings, Catherine Leroy, Sarah Edkins, Laura J Mudie, et al. Complex landscapes of  
548 somatic rearrangement in human breast cancer genomes. *Nature*, 462(7276):1005–1010, 2009.
- 549 [36] Pedro AF Galante, Raphael B Parmigiani, Qi Zhao, Otávia L Caballero, Jorge E De Souza, Fábio CP  
550 Navarro, Alexandra L Gerber, Marisa F Nicolás, Anna Christina M Salim, Ana Paula M Silva, et al.  
551 Distinct patterns of somatic alterations in a lymphoblastoid and a tumor genome derived from the same  
552 individual. *Nucleic Acids Research*, 39(14):6056–6068, 2011.
- 553 [37] Qi Zhao, Otavia L Caballero, Samuel Levy, Brian J Stevenson, Christian Iseli, Sandro J De Souza,  
554 Pedro A Galante, Dana Busam, Margaret A Leversha, Kalyani Chadalavada, et al. Transcriptome-  
555 guided characterization of genomic rearrangements in a breast cancer cell line. *Proceedings of the  
556 National Academy of Sciences, USA*, 106(6):1886–1891, 2009.
- 557 [38] Dan R Robinson, Shanker Kalyana-Sundaram, Yi-Mi Wu, Sunita Shankar, Xuhong Cao, Bushra Ateeq,

- 558 Irfan A Asangani, Matthew Iyer, Christopher A Maher, Catherine S Grasso, et al. Functionally recur-  
559 rent rearrangements of the MAST kinase and Notch gene families in breast cancer. *Nature Medicine*,  
560 17(12):1646–1651, 2011.
- 561 [39] Silvia Liu, Wei-Hsiang Tsai, Ying Ding, Rui Chen, Zhou Fang, Zhiguang Huo, SungHwan Kim,  
562 Tianzhou Ma, Ting-Yu Chang, Nolan Michael Priedigkeit, Adrian V. Lee, Jianhua Luo, Hsei-Wei  
563 Wang, I-Fang Chung, and George C. Tseng. Comprehensive evaluation of fusion transcript detection  
564 algorithms and a meta-caller to combine top performing methods in paired-end RNA-seq data. *Nucleic*  
565 *Acids Research*, 44(5):e47, 2016. doi: 10.1093/nar/gkv1234. URL +[http://dx.doi.org/10.1093/nar/](http://dx.doi.org/10.1093/nar/gkv1234)  
566 [gkv1234](http://dx.doi.org/10.1093/nar/gkv1234).
- 567 [40] Charles J Sherr. Principles of tumor suppression. *Cell*, 116(2):235–246, 2004.
- 568 [41] Monica Hollstein, David Sidransky, Bert Vogelstein, and Curtis C Harris. p53 mutations in human  
569 cancers. *Science*, 253(5015):49–54, 1991.
- 570 [42] Donna Maglott, Jim Ostell, Kim D Pruitt, and Tatiana Tatusova. Entrez Gene: gene-centered informa-  
571 tion at NCBI. *Nucleic Acids Research*, 39(suppl 1):D52–D57, 2011.
- 572 [43] Xudong Wu, Ida Holst Bekker-Jensen, Jesper Christensen, Kasper Dindler Rasmussen, Simone Sidoli,  
573 Yan Qi, Yu Kong, Xi Wang, Yajuan Cui, Zhijian Xiao, et al. Tumor suppressor ASXL1 is essential for  
574 the activation of *INK4B* expression in response to oncogene activity and anti-proliferative signals. *Cell*  
575 *Research*, 25(11):1205–1218, 2015.
- 576 [44] Vineet Bafna and Pavel A Pevzner. Genome rearrangements and sorting by reversals. *SIAM Journal*  
577 *on Computing*, 25(2):272–289, 1996.
- 578 [45] Christoph Bartenhagen and Martin Dugas. RSVSim: an R/Bioconductor package for the simulation of  
579 structural variations. *Bioinformatics*, 29(13):1679–1681, 2013.
- 580 [46] Andrew Yates, Wasiu Akanni, M Ridwan Amode, Daniel Barrell, Konstantinos Billis, Denise  
581 Carvalho-Silva, Carla Cummins, Peter Clapham, Stephen Fitzgerald, Laurent Gil, et al. Ensembl  
582 2016. *Nucleic Acids Research*, 44(D1):D710–D716, 2015.
- 583 [47] Thasso Griebel, Benedikt Zacher, Paolo Ribeca, Emanuele Raineri, Vincent Lacroix, Roderic Guigó,

- 584 and Michael Sammeth. Modelling and simulating generic RNA-Seq experiments with the flux simu-  
585 lator. *Nucleic Acids Research*, 40(20):10073–10083, 2012.
- 586 [48] Bronwen L Aken, Sarah Ayling, Daniel Barrell, Laura Clarke, Valery Curwen, Susan Fairley, Julio  
587 Fernandez Banet, Konstantinos Billis, Carlos García Girón, Thibaut Hourlier, et al. The Ensembl gene  
588 annotation system. *Database*, 2016, 2016.
- 589 [49] Alexander Dobin, Carrie A Davis, Felix Schlesinger, Jorg Drenkow, Chris Zaleski, Sonali Jha, Philippe  
590 Batut, Mark Chaisson, and Thomas R Gingeras. STAR: ultrafast universal RNA-seq aligner. *Bioinfor-*  
591 *matics*, 29(1):15–21, 2013.
- 592 [50] Heng Li and Richard Durbin. Fast and accurate short read alignment with Burrows–Wheeler transform.  
593 *Bioinformatics*, 25(14):1754–1760, 2009.

## 594 **Supplementary Text**

595 All experiments here are performed with SQUID version 1.0.

### 596 **Using de novo assembly and transcript to genome alignment to predict TSV**

597 For the pipeline of de novo transcriptome assembly and transcript-to-genome alignment, the direct output is  
598 a series of alignment pieces for each assembled transcript. To derive TSV from the pieces of alignment of  
599 each transcript, we still need to use the split-read alignment concordance criteria (8) and the edge-building  
600 approach. In the case of no TSV, equation (8) still holds, since a transcript is generated from one strand of  
601 one chromosome, without rearrangements but only deletion of introns. Any violation of (8) is treated as a  
602 TSV. Here TSVs are still able to be represented by edges in GSG, where segments are the intervals of each  
603 piece of alignment, and edges are added in the same principle that traversing segments along the edges will  
604 result in a concordant alignment of the assembled transcript. The positions of both breakpoints in a TSV are  
605 exactly the two positions linked by the discordant edge, and the orientations corresponds to the connection  
606 type of the edge.

### 607 **Processing TCGA RNA-seq data**

608 We use STAR aligner [49] to align TCGA RNA-seq reads to Ensemble genome 87 [46] with the corre-  
609 sponding gene annotation. STAR aligner [49] is set with the option of outputting chimeric alignments with  
610 hanging length 15bp. The chimeric alignments generated by STAR [49] are further filtered out if the paired-  
611 end reads can be aligned concordantly by SpeedSeq aligner [33] SQUID is applied to concordant alignment  
612 generated by STAR [49] and filtered chimeric alignment. The discordant edge weight coefficient  $\alpha$  is set to  
613 be 1, that is, we require tumor transcripts to dominate normal transcripts in order to predict corresponding  
614 TSVs.

615 A large number of fusions between immunoglobulin genes are predicted by SQUID. However, there is  
616 possibility that B cells are in the mixture of sequencing and have very high expression of immunoglobulin  
617 genes (Ig). We cannot tell whether Ig rearrangements are generated by tumor cells or B cells. Therefore, we  
618 exclude Ig TSVs during post-processing and exclude them from the descriptive statistics. Note that SQUID  
619 does not exclude Ig TSVs internally, because Ig expression and VDJ recombination have been observed to

620 exist in tumor cells, and revealing the role of Ig in tumor can deepen our understanding of cancer. When  
621 normal cells are removed from tumor samples, using SQUID to predict Ig TSV will help the study of Ig and  
622 tumor.

### 623 SQUID parameters

Table S1: Value of SQUID's parameters used in experiments

Symbol	Description	Value
$\gamma$	segment degree threshold	4
$\theta$	edge weight threshold	5
$\alpha$	discordant edge weight coefficient	8 (simulation and HCC cell line), 1 (TCGA)
$mq$	minimum mapping quality	255 (STAR), 1 (SpeedSeq)
$pq$	low Phred quality threshold	4 ( $p = 10^{-0.4}$ )
$l$	maximum allowed low Phred quality length	10

624 Note:  $mq$ ,  $pq$  and  $l$  are controls for sequencing quality and mapping quality. If mapping quality of a read is  
625 less than threshold  $mq$ , the read will not be used in edge building. If the read has a low sequencing, in terms  
626 of having more than  $l$  bases of sequencing quality lower than  $pq$ , the read will not be used in edge building.

627 **Supplementary Figures**

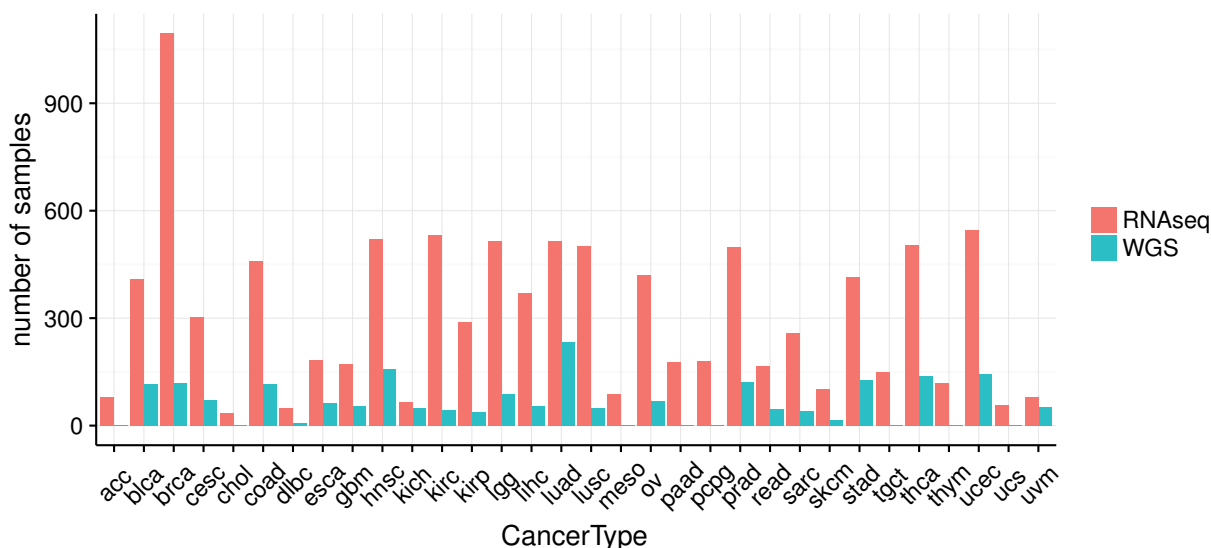


Figure S1: Number of samples with RNA-seq or WGS data in TCGA

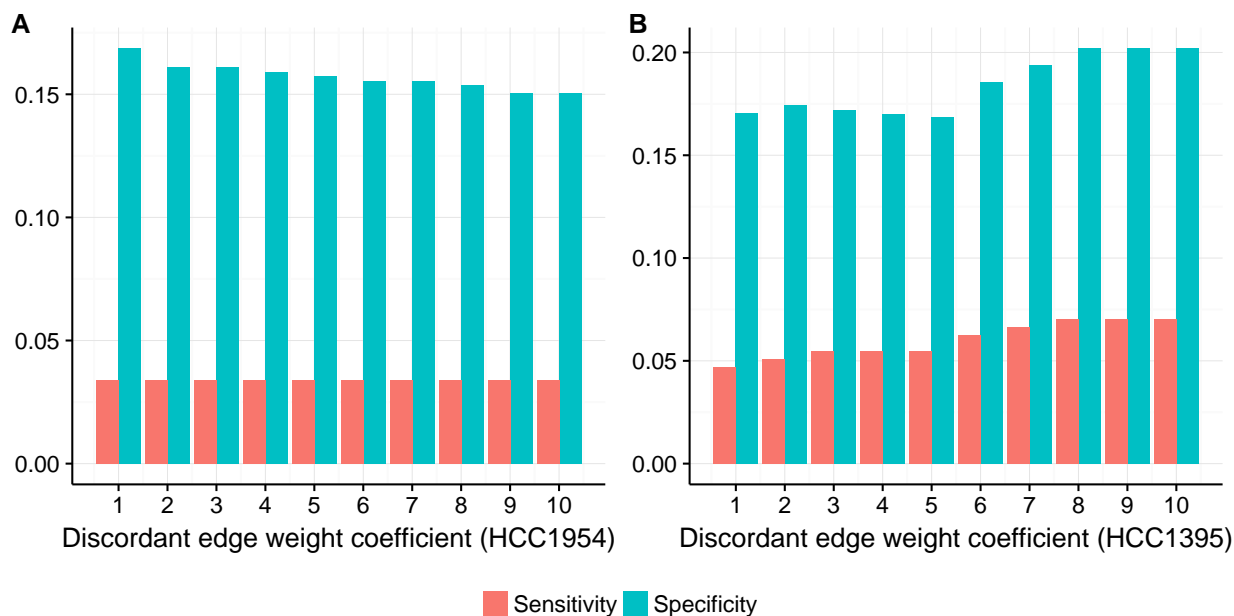


Figure S2: Specificity and sensitivity of SQUID against different value of discordant edge weight coefficient. (A) HCC1954 cell line. Sensitivity does not change when increasing discordant edge weight coefficient, indicating rearranged tumor transcripts out-number their normal counterparts. Specificity decreases slightly because SQUID predicts more as discordant edge weight coefficient increases. (B) HCC1395 cell line. Sensitivity and specificity reach the highest at discordant edge weight coefficient 8 and remain unchanged at 9 and 10. Some normal transcripts out-number the rearranged tumor transcripts, increasing this parameter allows SQUID to capture these TSVs.