BIOINFORMATICS

Vol. 00 no. 00 2015
Pages 1–8

IgRepertoireConstructor: a novel algorithm for antibody repertoire construction and immunoproteogenomics analysis. Appendix

Yana Safonova [1,4,6], Stefano Bonissone [2,6] Eugene Kurpilyansky [4], Ekaterina Starostina [1,4], Alla Lapidus [1,4], Wendy Sandoval [5], Jennie Lill [5] and Pavel A. Pevzner [1,3,4,*]

[1]Center for Algorithmic Biotechnology, St. Petersburg State University, Russia
[2]Bioinformatics Program, University of California at San Diego, USA
[3]Dept. of Computer Science and Engineering, University of California at San Diego, USA
[4]Algorithmic Biology Laboratory, St. Petersburg Academic University, Russia
[5]Genentech, South San Francisco, California, USA
[6]These authors have contributed equally.

Received on XXXXX; revised on XXXXX; accepted on XXXXX

Associate Editor: XXXXXXX

APPENDIX A THE HAMMING GRAPH BOUND SELECTION

The important question is how to select the bound τ while constructing the Bounded Hamming Graph $HG(Strings, \tau)$. The input to IGREPERTOIRECONSTRUCTOR is overlapping paired-end reads that are merged into single reads covering the variable region of the antibody (about 400 nt). If sequencing errors in reads (the error rate in Illumina reads is $\approx 1\%$) are not corrected by merging, the merged read are expected to contain ≈ 4 errors on average. Thus, unless the rate of sequencing errors is reduced by the merging procedure, two merged reads are expected to have $\approx 4 + 4 = 8$ mismatches on average. Unfortunately, the threshold $\tau = 8$ will not work for error-correction in immunosequencing since different antibodies often differ by less than 8 mismatches (Figure A10).

However, it turns out that our algorithm benefits from the fact that most errors are concentrated at the ends of reads resulting in merged reads with significantly higher accuracy than the accuracy of the original paired-end reads (see Appendix F). To estimate the average number of errors in the merged reads, we extracted reads corresponding to the known contaminant (*Streptococcus pneumoniae*) in our Ig-seq dataset and aligned them to the *Streptococcus pneumoniae* genome. It turned out that 96% of merged reads differ from the reference genome by at most 1 mismatch (98% of merged reads differ from the reference genome by at most 2 mismatches). Thus, we have selected the bound $\tau = 3$ for constructing the Bounded Hamming Graph.

APPENDIX B DENSE SUBGRAPHS AND CLIQUES IN TRIANGULATED GRAPH

Fig. A11a shows a triangulated graph G containing three dense subgraphs, yellow, green, and violet. Fig. A11b shows a *clique overlap graph* of G where vertices correspond to maximal cliques in G and edges connect cliques that share vertices. The weight of an edge is the number of vertices shared between two cliques. A *clique tree* is defined as a maximum spanning tree of the clique overlap graph (Fig. A11c). Fig. A11d show a perfect elimination order for the graph from Fig. A11a. Fig. A11e shows the vertex elimination process for the same graph.

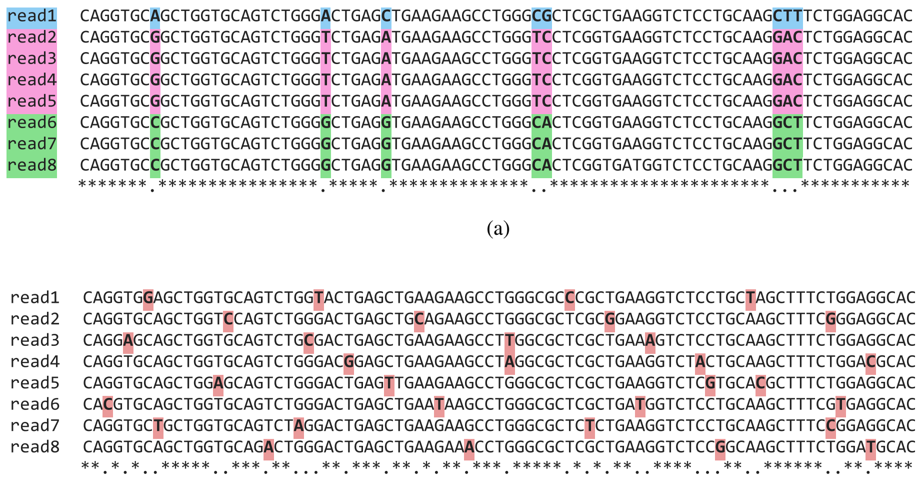
Dense subgraphs are formed by mutliple cliques in the clique tree. For example, vertices from the dense yellow subgraph can be found among four nodes from the clique tree. To construct dense subgraphs, we merge maximal cliques connected by many edges. E.g., we merge two yellow cliques resulting in a subgraph on 5 vertices with a high edge fill-in.

Note that some of the resulting dense subgraphs may share vertices forcing us to assign these shared vertices to one of the dense subgraphs. To assign each vertex v to a single dense subgraph, we select a the subgraph that has the maximum number of vertices adjacent to v among all subgraphs containing v.

APPENDIX C SPLITTING DENSE SUBGRAPHS USING SHM DETECTION

In practice, SHMs associate with various patterns (Dorner *et al.* (1997)) making it difficult to apply the approach for breaking dense subgraphs described in the main text. To bypass this complication, we define the notion of a *mutation-edge-set* as the set of all edges corresponding to a given mutation. We further define an SHM as a mutation whose mutation-edge-set

[*]to whom correspondence should be addressed



```
read1  CAGGTGCAGCTGGTGCAGTCTGGGACTGAGCTGAAGAAGCCTGGGCGCTCGCTGAAGGTCTCCTGCAAGCTTTCTGGAGGCAC
read2  CAGGTGCGGCTGGTGCAGTCTGGGTCTGAGATGAAGAAGCCTGGGTCCTCGGTGAAGGTCTCCTGCAAGGACTCTGGAGGCAC
read3  CAGGTGCGGCTGGTGCAGTCTGGGTCTGAGATGAAGAAGCCTGGGTCCTCGGTGAAGGTCTCCTGCAAGGACTCTGGAGGCAC
read4  CAGGTGCGGCTGGTGCAGTCTGGGTCTGAGATGAAGAAGCCTGGGTCCTCGGTGAAGGTCTCCTGCAAGGACTCTGGAGGCAC
read5  CAGGTGCGGCTGGTGCAGTCTGGGTCTGAGATGAAGAAGCCTGGGTCCTCGGTGAAGGTCTCCTGCAAGGACTCTGGAGGCAC
read6  CAGGTGCCGCTGGTGCAGTCTGGGGCTGAGGTGAAGAAGCCTGGGCACTCGGTGAAGGTCTCCTGCAAGGCTTCTGGAGGCAC
read7  CAGGTGCCGCTGGTGCAGTCTGGGGCTGAGGTGAAGAAGCCTGGGCACTCGGTGAAGGTCTCCTGCAAGGCTTCTGGAGGCAC
read8  CAGGTGCCGCTGGTGCAGTCTGGGGCTGAGGTGAAGAAGCCTGGGCACTCGGTGATGGTCTCCTGCAAGGCTTCTGGAGGCAC
       *******.****************.*****.**************..**********************...***********
```

(a)

```
read1  CAGGTGGAGCTGGTGCAGTCTGGTACTGAGCTGAAGAAGCCTGGGCGCCCGCTGAAGGTCTCCTGCTAGCTTTCTGGAGGCAC
read2  CAGGTGCAGCTGGTCCAGTCTGGGACTGAGCTGCAGAAGCCTGGGCGCTCGCGGAAGGTCTCCTGCAAGCTTTCGGGAGGCAC
read3  CAGGAGCAGCTGGTGCAGTCTGCGACTGAGCTGAAGAAGCCTTGGCGCTCGCTGAAAGTCTCCTGCAAGCTTTCTGGAGGCAC
read4  CAGGTGCAGCTGGTGCAGTCTGGGACGGAGCTGAAGAAGCCTAGGCGCTCGCTGAAGGTCTACTGCAAGCTTTCTGGACGCAC
read5  CAGGTGCAGCTGGAGCAGTCTGGGACTGAGTTGAAGAAGCCTGGGCGCTCGCTGAAGGTCTCGTGCACGCTTTCTGGAGGCAC
read6  CACGTGCAGCTGGTGCAGTCTGGGACTGAGCTGAATAAGCCTGGGCGCTCGCTGATGGTCTCCTGCAAGCTTTCGTGAGGCAC
read7  CAGGTGCTGCTGGTGCAGTCTAGGACTGAGCTGAAGAAGCCTGGGCGCTCTCTGAAGGTCTCCTGCAAGCTTTCCGGAGGCAC
read8  CAGGTGCAGCTGGTGCAGACTGGGACTGAGCTGAAGAAACCTGGGCGCTCGCTGAAGGTCTCCGGCAAGCTTTCTGGATGCAC
       **.*.*..*****..***.**...**.***.**.*.**.***.*****.*.*.**..****...**..******..**.****
```

(b)

Fig. A10: (a) An alignment of reads corresponding to three similar antibodies (highlighted in blue, violet, and green). For the sake of simplicity, we show error-free reads. For comparison, (b) shows the alignment of reads originating from the same antibody, that contain randomly located errors in reads.

splits the subgraph (cluster of reads) into relatively large subclusters. Thus, IGREPERTOIRECONSTRUCTOR attempts to split each constructed dense subgraph by identifying SHMs. The split subgraphs revealed by this final step define the reads contributing to each antibody in the antibody repertoire. Below we explain how IGREPERTOIRECONSTRUCTOR splits dense subgraphs by identifying SHMs.

To design our splitting rule, we aligned reads from each dense subgraph. For each column i in the alignment, we define $count_i$ and $fraction_i$ and the count and fraction of *second* most frequent nucleotide in the i-th column. We define thresholds $count_{min}$ (the default value is 4) and $fraction_{min}$ (the default value 0.01) and limit our attention to all columns with surprisingly large fractions of the second most frequent nucleotides ($fraction_i > fraction_{min}$) among columns where this nucleotide occurs more than $count_{min}$ times. We further refer to such columns as *SHM columns* and split all dense subgraph that have SHM columns. While it may appear that the default value $fraction_{min} = 0.01$ is too small to distinguish SHMs from sequencing errors, we note that it applies to *stitched* Ig-seq reads that feature rather small error rates (0.0022 on average).

Overall, we detected $18,097$ such columns distributed over 4126 dense subgraphs constructed by IGREPERTOIRECONSTRUCTOR (25.79% percent of all dense subgraphs). Figure A12a presents the scatter plot of $(count_i, fraction_i)$. Figure A12b shows the histogram of the distribution of the $count_i$ values.

APPENDIX D EVALUATING THE CONSTRUCTED REPERTOIRES

We evaluate the constructed repertoire by checking whether the Ig-Seq reads from the same cluster exhibit variations typical for errors in reads (as expected from correctly constructed clusters) or variations typical for incorrectly constructed clusters formed by multiple antibodies. In order to analyze the pattern of variations, we align reads from each cluster and compute the distribution of positions of mismatches along the length of the reads. If this distribution is roughly uniform (as expected from sequencing errors), we conclude that the cluster is constructed correctly. However, if this distribution reveals some peaks (e.g., peaks in CDR regions), we conclude that two different antibodies were merged into a single cluster.

Fig. A13 shows a histogram of mismatch positions averaged over all clusters. Since this distribution is rather uniform (except for peaks in the beginning and end of reads typical for the error profile of Illumina reads), we conclude that most clusters correspond to a single antibody. In contrast, the distribution of variations for antibody clones (groups of multiple antibodies with the same CDR3) shows pronounced peaks in CDR1 and CDR2 regions indicating that the clones are formed by multiple antibodies (Fig. A16).

APPENDIX E BENCHMARKING IGREPERTOIRECONSTRUCTOR ON SIMULATED IMMUNOSEQUENCING DATA

In order to check accuracy of IGREPERTOIRECONSTRUCTOR, we generated small simulated immunosequencing data set using IGSIMULATOR (Safonova *et al.* (2015)) with the following parameters: *# base sequences* $= 10,000$, *# mutated sequences* $= 100,000$ and *expected repertoire size* $= 1,000,000$. The simulated repertoire contains $105,438$ clusters (size of the maximal cluster is 112, number of singletons is $10,025$). Experiments showed that

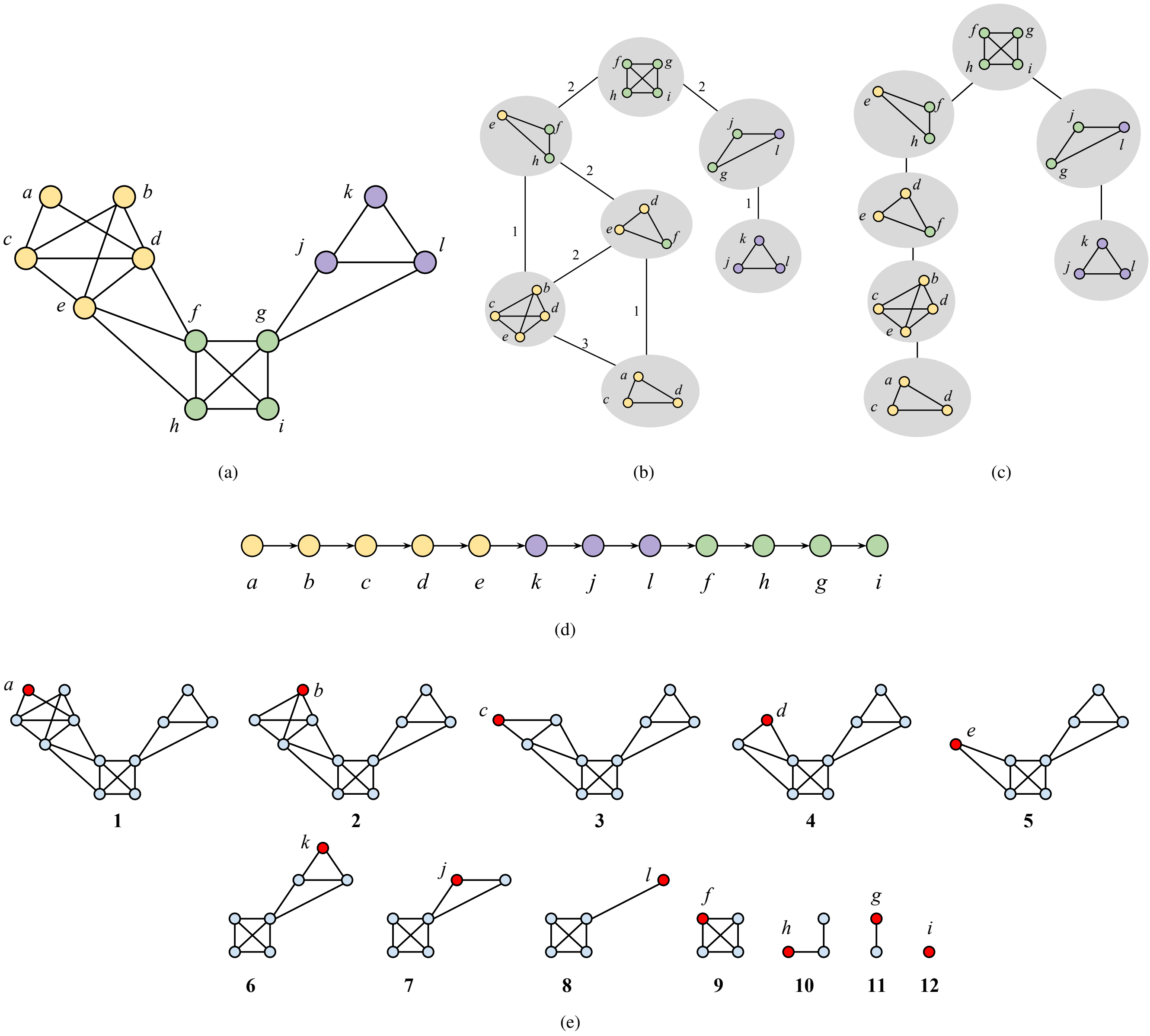


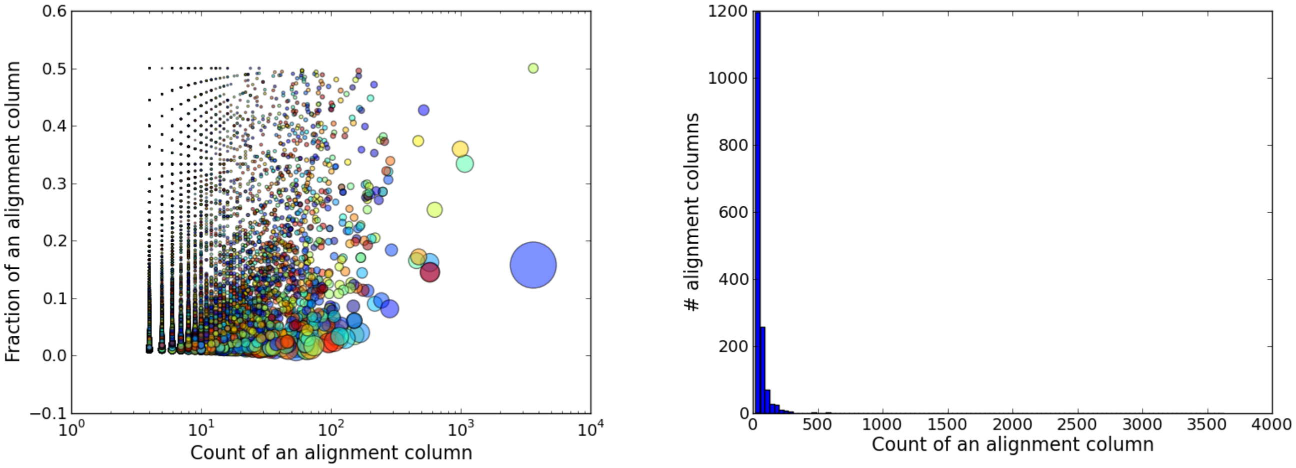
Fig. A11: A triangulated graph (a), its clique graph (b) and its clique tree (c). (d) perfect elimination order for the triangulated graph in (a). (e) the vertex elimination process for the triangulated graph in (a).

IGREPERTOIRECONSTRUCTOR accurately recovers clusters in the simulated repertoire except for several small clusters broken into singletons in the constructed repertoire.

APPENDIX F IG-SEQ DATA PREPROCESSING

Read merging. IGREPERTOIRECONSTRUCTOR works with single reads that cover the entire variable regions of antibodies. These reads are generated by merging the paired-end Illumina reads.

Since paired reads in our dataset have average insert size 366 nt (Figure A14a), they are expected to overlap by $\approx 250 + 250 - 366 = 134$ nt. After finding the overlap, we merge the two reads within a read-pair into a single merged read that is expected to cover the entire variable region of antibody. This procedure results in a significant reduction of error rates. Since the accuracy of Illumina reads drops towards the end of reads, we take advantage of the overlapping region and form its consensus by selecting the nucleotide with maximal quality value at each position in the



(a)

(b)

Fig. A12: (a) the scatter plot of pairs $(count_i, fraction_i)$ for filtered alignment columns using thresholds $count_{min} = 4$ and $fraction_{min} = 0.01$. The area of a circle is proportional to the value of $count_i$. Colors of circles are individual for each alignment column. (b) the histogram of the distribution of the $count_i$ values.

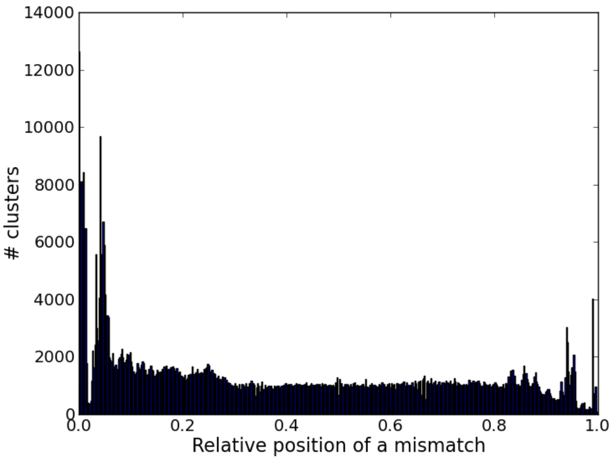


Fig. A13: Histogram of distribution of the relative mismatch positions for the constructed antibody clusters.

overlap. Figure A14b shows the difference in error rates before and after merging.

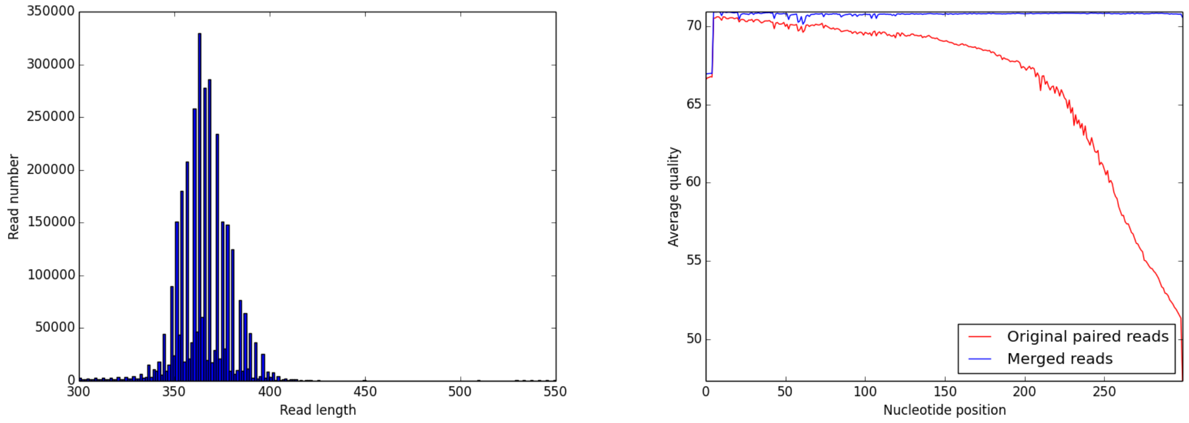
Contamination clean-up. We used IgBlast (Ye *et al.* (2013)) to align merged reads to Ig germline database and removed reads that have alignment with E-value exceeding 0.001. We further filtered reads and assembled them with SPAdes assembler (Bankevich *et al.* (2012)) resulting in 10 contigs (Table A1). Blast alignments of constructed contigs show that filtered reads were correctly classified as contamination and can be safely removed from the Ig-seq library.

ID	*Length (nt)*	*Coverage*	*Blast alignment*
1	1512	1.2	Escherichia coli genome assembly FHI92
2	1195	25.4	Homo sapiens major histocompatibility complex
3	959	1.9	Escherichia coli genome assembly FHI89
4	929	1.0	Homo sapiens protein tyrosine phosphatase
5	827	29.0	Homo sapiens O-sialoglycoprotein endopeptidase
6	868	1.3	Escherichia coli genome assembly FHI89
7	780	1.3	Homo sapiens immunoglobulin heavy locus (IGH)
8	734	1.0	Homo sapiens B lymphoid tyrosine kinase (BLK)
9	722	31.0	Homo sapiens uncharacterized LOC102725417
10	240	14.7	Homo sapiens long non-coding RNA

Table A1. Contigs assembled from reads filtered as contaminants. The table shows length, coverage and the best Blast alignment for each contig from assembly.

APPENDIX G CONTAMINATED READS ANALYSIS

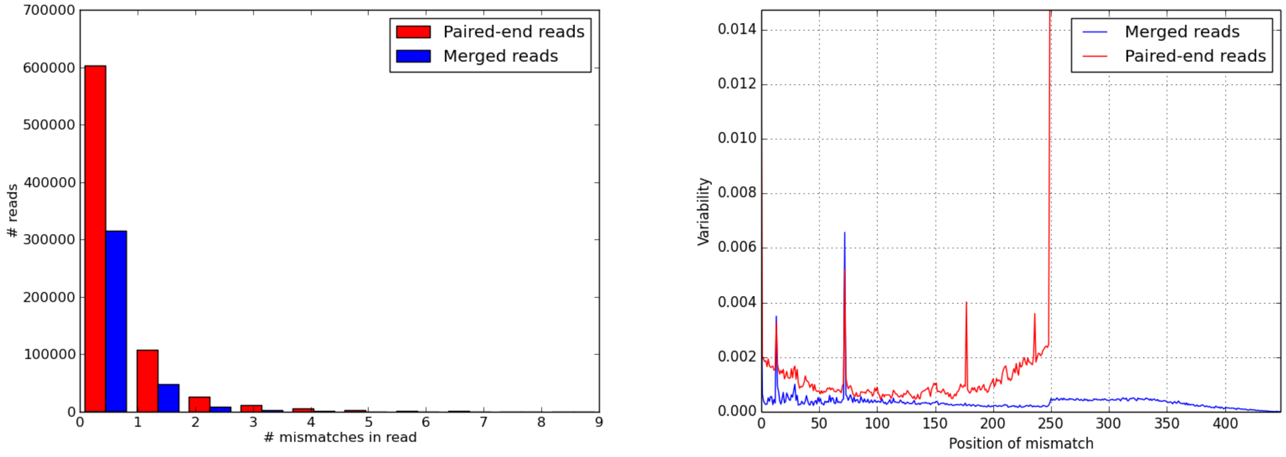
We used the fact that some our Ig-seq libraries contain contaminants to estimate the average error rate of the paired-end and merged Ig-seq reads. We identified reads corresponding to the genome of *Streptococcus sp. VT 162* and aligned both paired-end, and merged, reads to the reference genome. The average number of mismatches per read is 0.85 and 0.22 for the paired-end and merged reads, respectively. Figure A15 shows that the overlapping parts of the merged reads contain fewer errors as compared to paired-end reads.



(a)

(b)

Fig. A14: (a) shows histogram of merged read length distribution. (b) shows the average quality of reads before (red) and after (blue) merging, and illustrates that merging of overlapping reads significantly improves their quality.



(a)

(b)

Fig. A15: Analysis of the error rate of Ig-seq libraries using reads from contaminants. (a) shows the histogram of mismatches number per read distribution. (b) shows the histogram of mismatches position distribution. Since the length of merged reads is not fixed, we compute the position of the mismatch as the distance to the nearest start from the overlapping reads. Thus, positions of mismatches are normalized from 1 to 250.

APPENDIX H VALIDATING ANTIBODY REPERTOIRES

The effect from IGREPERTOIRECONSTRUCTOR is evident when comparing the peptides identified from the unique reads database and the antibody repertoire. Only 0.6% of peptides identified from the unique reads database do not appear in the antibody repertoire. This demonstrates that IGREPERTOIRECONSTRUCTOR rarely overcorrects reads implying that hardly any information is lost as the result of error correction and that the antibody repertoire is a better option for immunoproteogenomics searches than the previously used the unique reads database.

To further evaluate the constructed repertoires, we performed additional analysis of CDR3 regions. Differing antibody clusters with shared CDR3 sequence (and V region labeling) partition all antibodies into clones. We refer to the *capacity of the clone* as the number of antibodies composing it.

Since coincidence of CDR3 region of two unrelated antibodies is an unlikely event, there are two possible explanations for non-trivial clones: (i) erroneous partitioning of reads from the same antibody into multiple clusters due to insufficient error correction, and (ii) correct clustering of multiple differing antibodies into multiple clusters with the same CDR3. In the latter case, since these multiple antibodies were not exposed to diversity mechanisms (such as SHM) in their CDR3 region, we expect that variations in these antibodies dominate in CDR1 and CDR2 regions.

To test whether the case (ii) holds, we used CLUSTAL W, version 2.0 (Larkin *et al.* (2007)) to align all antibodies within a clone, and to identify the variable positions in all non-trivial clones. Figure A16 shows the histogram of the variable positions for the constructed heavy chain repertoire. Since the peaks in the histograms are located at approximate positions of CDR1 and CDR2 regions, we conclude that most non-trivial clones indeed were formed by related and diversified antibodies (rather than by errors in clustering). See Appendix J for the peptide coverage over the CDR3 region.

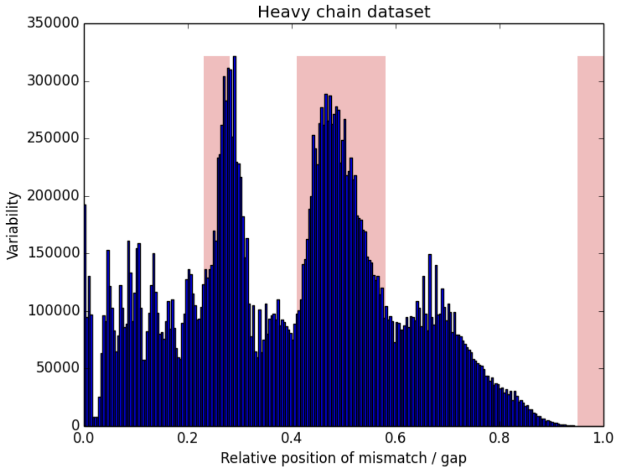


Fig. A16: Histogram of the mutated positions among non-trivial clones. Mutated positions are computed as relative positions of columns in multiple alignment of antibodies from each clone corresponding to a mismatch or an indel. The histogram was cut off at the right border of the CDR3 region. Red vertical bars correspond to positions of CDRs as specified in Murphy (2012).

APPENDIX I BLIND MODIFICATION SEARCH RESULTS

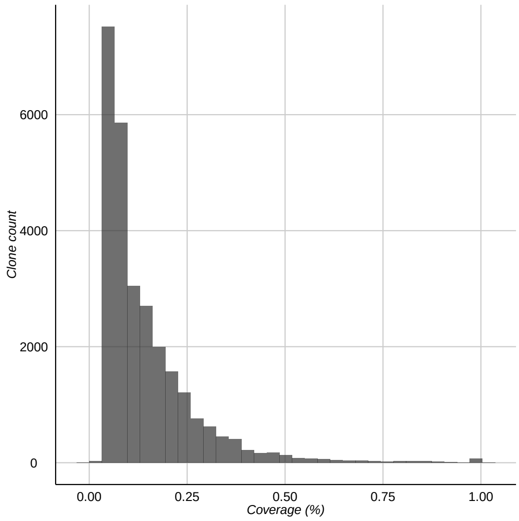
Figure A17 shows the prevalence of modifications over different peptide sets; those observed only with a modification Figure A17a and A17b, and those observed both with and without the modification Figure A17c and A17d.

MODa identified many PTMs with offsets -17/-18, +16, and +42Da. The 17Da and 18Da losses can be explained as pyroglutamic acid, occurring on Q and E, particularly on the N-terminus. The 16Da gains are nearly all located on methionine and tryptophan, consistent with oxidation. While some of the 42Da gains occur on serines consistent with acetylation, the majority occur on cystines. These are likely the result of N-isopropyl iodoacetamide (NIPCAM), since cystines are searched with a fixed +57Da offset.

The PTMs with offset +1Da identified by MODa are largely attributed to asparagine (N), which could signify a mutation to isoleucene/leucine (I/L). MODa found many modifications on tryptophan (W) centered around offsets of +32Da, +16Da, and +5Da, all of which can be attributed to oxidations of tryptophan. Additional prominent offsets were +12Da gain on glycine (G), and -9Da loss on arginine (R), both seen in Figure A17b. A 9Da loss on R can be explained as a mutation to phenylalanine (F). However, the addition to G cannot be explained with common modifications or mutations.

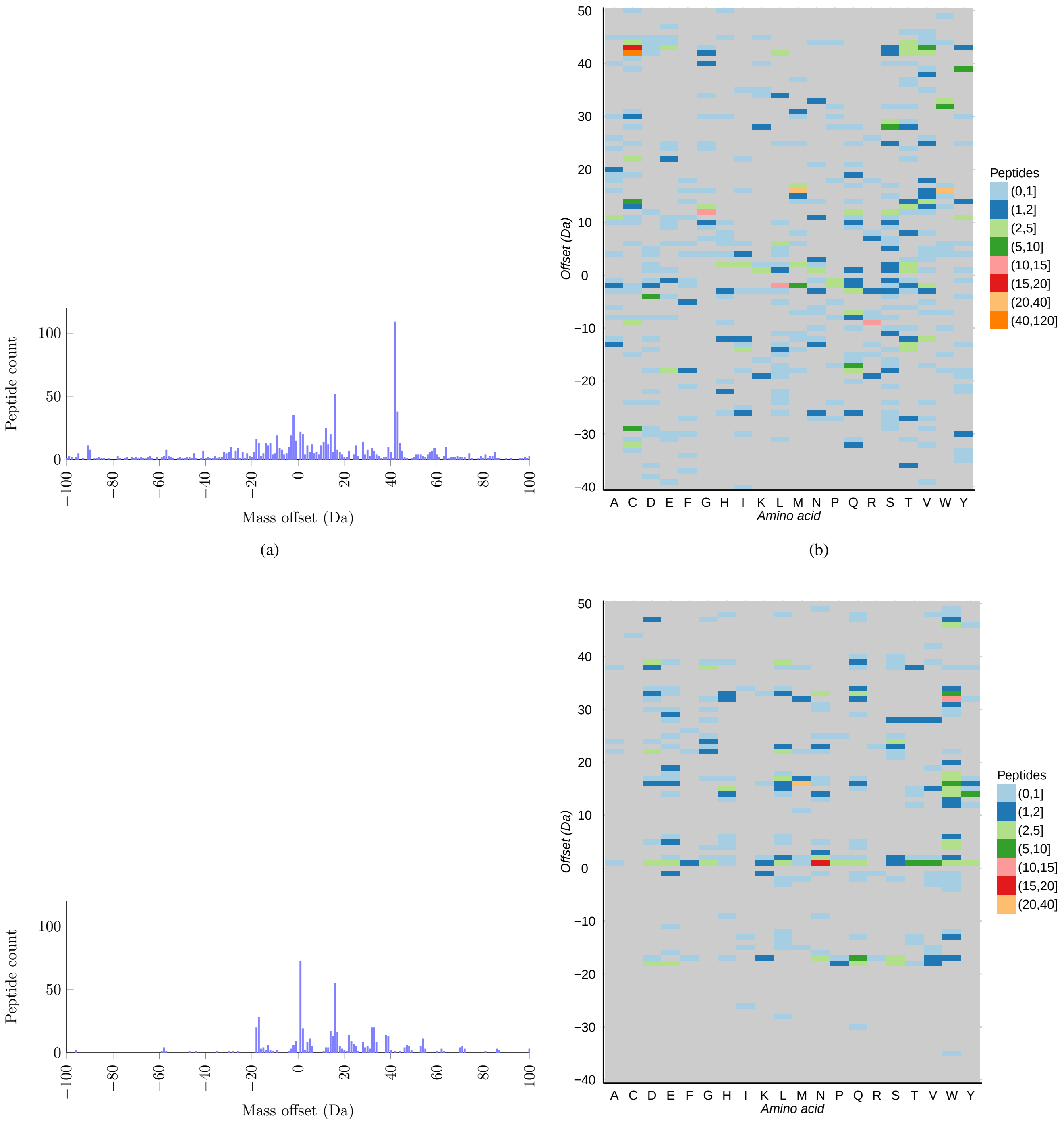
APPENDIX J COVERAGE OF CDR3 REGION BY PEPTIDES

Peptide coverage of antibodies and clones is of interest since it can provide us with direct proteomic evidence of which antibodies/clones are specific to the introduced antigen. Of particular interest is the peptide coverage over the region which defines clonality; the CDR3 region. Figure A18a shows the coverage distribution for each clone over the CDR3 region and reveals that often very few residues are being covered at the junction of CDR3 region. However, few clones have high coverage over the entire region (99 clones with 90% or more coverage). Additional representations of clone coverage are shown in Figure A19.



(a)

Fig. A18: (a) Peptide coverage distribution of CDR3.



(a)

(b)

(c)

(d)

Fig. A17: Modifications of peptides identified only with that modification, or with and without the modification. (a) Histogram of offsets over 1099 peptides with only modifications, (b) and their breakdown by residue. (c). 1051 out of these 1497 peptides were not identified in restrictive MS/MS searches. Computed on 544 peptides with observed non-modified and modified versions, (d) along with residue breakdown.

REFERENCES

Bankevich, A., Nurk, S., Antipov, D., Gurevich, A. A., Dvorkin, M., Kulikov, A. S., Lesin, V. M., Nikolenko, S. I., Pham, S., Prjibelski, A. D., Pyshkin, A. V., Sirotkin, A. V., Vyahhi, N., Tesler, G., Alekseyev, M. A., and Pevzner, P. A. (2012). SPAdes: A New Genome Assembly Algorithm and Its Applications to Single-Cell Sequencing. *Journal of Computational Biology*, **19**, 455–477.

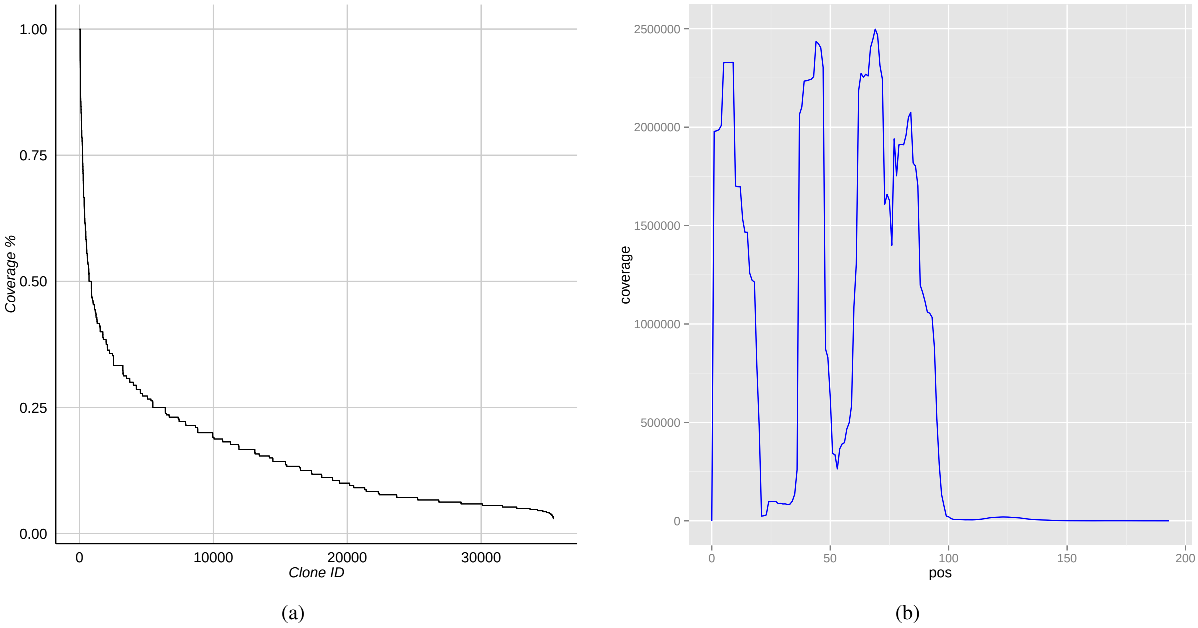
Dorner, T., Brezinschek, H. P., Brezinschek, R. I., Foster, S. J., Domiati-Saad, R., and Lipsky, P. E. (1997). Analysis of the frequency and pattern of somatic mutations within nonproductively rearranged human variable heavy chain genes. *J Immunol.*, **6**(158), 2779–89.

Larkin, M., Blackshields, G., Brown, N., Chenna, R., McGettigan, P., McWilliam, H., Valentin, F., Wallace, I., Wilm, A., Lopez, R., Thompson, J., Gibson, T., and Higgins, D. (2007). Clustal w and clustal x version 2.0. *Bioinformatics*, pages 2947–8.

Murphy, K. P. (2012). *Janeway's immunobiology*, chapter Antigen Recognition by B-cell and T-cell Receptors. Garland Science, 8th edition.

Safonova, Y., Lapidus, A., and Lill, J. (2015). IgSimulator: a versatile immunosequencing simulator. *Submitted*.

Ye, J., Ma, N., Madden, T., and Ostell, J. (2013). IgBlast: an immunoglobulin variable domain sequence analysis tool. *Nucleic Acids Res*, **41**, W34–40.



(a)

(b)

Fig. A19: (a) Clones sorted by percent coverage of the CDR3 region by peptides. (b) Peptide coverage over positions of each antibody. No normalization of coverage is performed.