UNIVERSITY OF CALIFORNIA, SAN DIEGO UCSD

BERKELEY • DAVIS • IRVINE • LOS ANGELES • MERCED • RIVERSIDE • SAN DIEGO • SAN FRANCISCO SANTA BARBARA • SANTA CRUZ

SIAVASH MIRARABBAYGI
ASSISTANT PROFESSOR OF ELECTRICAL AND COMPUTER ENGINEERING
9500 GILMAN DRIVE, MC 0407
LA JOLLA, CA 92093

PHONE: (858) 822-6245
E-MAIL: smirarabbaygi@ucsd.edu

March 30, 2017

Yana Safonova addresses two interesting and novel problems in Bioinformatics in her dissertation:

1. Assembly of highly polymorphic genomes from short reads
2. Analysis of immunogenomics medium size reads to form the antibody repertoire

Both problems are timely because new sequencing technologies make them more pressing than ever before. Neither problem had a sufficiently accurate solution in the past, and therefore, Yana's work will be the benchmark that future improvements should be compared against. For both problems, she has created new and interesting algorithms and has moved the state of the two respective areas. Her algorithms are implemented in usable software programs that have already been adopted by some in the community.

Many organisms are highly polymorphic and having an assembler that can accurately separate out the two haplomes has many applications. In my own area of research, given such assemblies, we could encode on individual as two different leaves of a phylogeny if we could separate out their haplomes.

Similarly, being able to build a clonal tree for antibodies would have not been possible without first building their repertoire, correcting for sequencing error. Yana’s work will now make the development of such algorithms possible.

The quality of Yana’s work exceeds most Ph.D. dissertations that I have seen. The papers that accompany the paper are of high quality and are published in the best venues of the Bioinformatics research area.

I enjoyed reading the brief and well-written English version of the dissertation. At several points, the dissertation could benefit from more details and further clarifications. I have pointed out

those instances in the attached file. I have also pointed out some typos that I found, but another round of proof reading can further improve the language. Beyond these, I don't have any concerns or comments about the work.

Siavash Mir Arabbaygi (Mirarab)
Assistant Professor, Electrical and Computer Engineering
UC San Diego
9500 GILMAN DRIVE, MC 0407
LA JOLLA, CA 92093
smirarab@ucsd.edu

1.3 Omics

The emergence of NGS technologies significantly enhanced genome analysis and allowed scientists to solve a number of problems related to the composition of the genome. This in turn gave a huge boost to the development of various *-omics* fields: *genomics* (study of genomes), *metagenomics* (study of complex communities and environmental samples), *transcriptomics* (study of RNA molecules, their structure and functions), *comparative genomics* (study of genome structure and function across various biological species), and *immunogenomics* (study of immune system components). As the NGS technologies continue to evolve, new ways of analyzing biological sequences are becoming available. E.g., Illumina MiSeq technology, released in 2013, is able to perform full-length scanning of adaptive immune repertoires (set of circulating antibodies and T cell receptors), making deep and accurate analysis of the immune system possible. The ability to solve immunogenomics problems opens new possibilities for the development of personalized medicine, estimation of vaccine efficiency and design of drugs for autoimmune disorders, cancer and infection diseases. However, even modern sequencing technologies, such as Illumina MiSeq, are unable to ~~ideally~~ solve the problem of reconstruction of highly polymorphic diploid genome, whose mutation rate is compatible to diversity of adaptive immune repertoires. As a result, evolving sequencing technologies leads to increasing need of more elaborate and sophisticated algorithmic approaches for processing new biological data.

comparable?

1.4 DNA sequencing and bioinformatics

During the HGP, two important computational problems were formulated: to reconstruct the raw human DNA sequence (*genome assembly*) and identify the genes encoded within (*genome annotation*). Both of them can be computationally formalized using observations on input data (sequencing reads or raw DNA sequence). Despite the fact that no one can annotate a project as well as a human expert biologist, dealing with such a vast amount of data would be extremely time consuming and require a lot of memory. Computer programs allow one to significantly increase the rate of data throughput, proving themselves to be a more adequate solution for the purpose of

genome reconstruction is called *genome assembly*. In order to be able to reconstruct a random string from short reads, one needs to find the extension of each of the reads and glue the overlapping portions into a single sequence. However, genome sequence is highly repetitive due to the duplication of genes and the presence of transposable elements (Figure 1.1a). This greatly complicates genome assembly. Typically, genome assembly does not yield a single string, but a set of contiguous fragments of the genome (*contigs*). Genome assembly algorithms attempt to compute long, non-overlapping and accurate contigs. There are several conventional approaches for assembling a genome: *the overlap-layout-consensus approach* (*OLC*) and *the de Bruijn graph approach*.

1.6.1 OLC approach

The OLC-based approaches were originally developed for the purpose of assembling Sanger reads. The OLC algorithms calculate the overlap graph, in which all vertices represent reads. Every two vertices that are adjacent in the graph correspond to a pair of overlapping reads. Contigs are then found by locating the non-branching paths in the constructed overlap graph (Figure 1.1b). The overlap-layout-consensus approach is used in such tools as: PHRAP [12], GAP [13], TIGR [14], ARACHNE [15], and CELERA [10]. In the worst case scenario, the number of edges in the overlap graph is quadratic (i.e., graph contains $O(n^2)$ edges, where n is a number of original reads) making this approach impractical for high throughput NGS reads.

1.6.2 De Bruijn graph approach

To assemble genomes from short and high throughput NGS reads algorithms using de Bruijn graph were proposed. Vertices of the de Bruijn graph are all possible k-mers (strings of length k) extracted from input reads. Two k-mers are adjacent on the de Bruijn graph if they are a prefix and a suffix of the $k+1$-mer presented in the input reads (Figure 1.1c). Computation of contigs is equivalent to finding paths that are consistent to input reads in the constructed de Bruijn graph. The de Bruijn graph

1.7 Immunogenomics

1.7.1 Adaptive immune repertoires

The subjects of immunogenomics studies are collections of circulating *antibodies* and *T-cell receptors* (or so-called *adaptive immune repertoires*). An adaptive immune repertoire can be represented by a set of sequences (each of them is characterized a unique antibody or T-cell receptor) with corresponding multiplicities (multiplicity shows the number of occurrences of the antibody/T-cell receptor in the repertoire). Adaptive immune repertoires reflect the state of an organism that make them important subjects of various biomedical studies. For example, monitoring of immune repertoire dynamics is a standard step of drug development pipeline. However, despite the importance in biology and medicine, the analysis of antibody and T-cell receptor repertoires remains poorly studied problem in immunogenomics. Until 2009, the computational analysis of antibodies had been performed using proteomics techniques [26] and did not rely on the DNA sequencing technologies. Weinstein, et al. [27] were the first to demonstrate the power of DNA sequencing for the purpose of analyzing antibody repertoires and to open an "NGS era" in antibody analysis. While many other immunosequencing studies have quickly followed in their footsteps [28–32], until 2012 there were no attempts to integrate NGS and MS approaches for antibody analysis. The absence of this integration (*immunoproteogenomics*) was a bottleneck, preventing the emergence of new approaches for analyzing large-scale and complex repertoires of antibodies and T-cell receptors.

by

Not clear how this can be "standard" if there are no existing good ways to do it.

a

Say more about why this integration is important. What is it that each of the two bring to the table?

1.7.2 Immunoproteogenomics

Cheung, et al. [33] pioneered a new immunoproteogenomics approach for identifying circulating *monoclonal antibodies* from serum that enables high-throughput antibody development. While sequencing purified monoclonal antibodies has now become routine [26;34;35], sequencing multiple antibodies from a complex sample (*polyclonal antibodies*) represents a breakthrough with great biomedical potential. The im-

portant conclusion in [33] is that antibody analysis should combine NGS and MS to infer antibodies interacting with a *specific antigen*, i.e., foreign and potentially harmful agent (see also [36–40]). In particular, Cheung, et al. [33] showed that the most well-represented transcripts in the antibody repertoire (revealed by NGS alone) may not be the most biomedically relevant. Thus, immunoproteogenomics is the key ingredient of the emerging new technology for antibody analysis. However, until 2015 no publicly available immunoproteogenomics software was available. An antibody repertoire (rather than a set of all DNA reads as in previous immunoproteogenomics studies) represents a sensible choice of a database for the follow up MS/MS searches. However, the construction of an antibody repertoire is a difficult problem since antibody genes are not directly encoded in the germline, but are diversified by somatic recombination and mutations [37]. Rapid development of sequencing technologies enabled deep full-length scanning of adaptive immune repertoires and posed new immunoinformatics challenges (see recent reviews by Georgiou, et al. [38]; Robinson [41]; Yaari and Kleinstein [42]; Grieff, et al. [43]). High accuracy of immune repertoires allows one to perform immunoproteogenomics analysis and solve such immunological problems as: analysis of clonal lineages [44; 45], statistical analysis of recombination events and secondary diversification [46; 47], analysis of immune response development [32; 48], population analysis of immunoglobulin and TCR loci [49] (Figure 1.2).

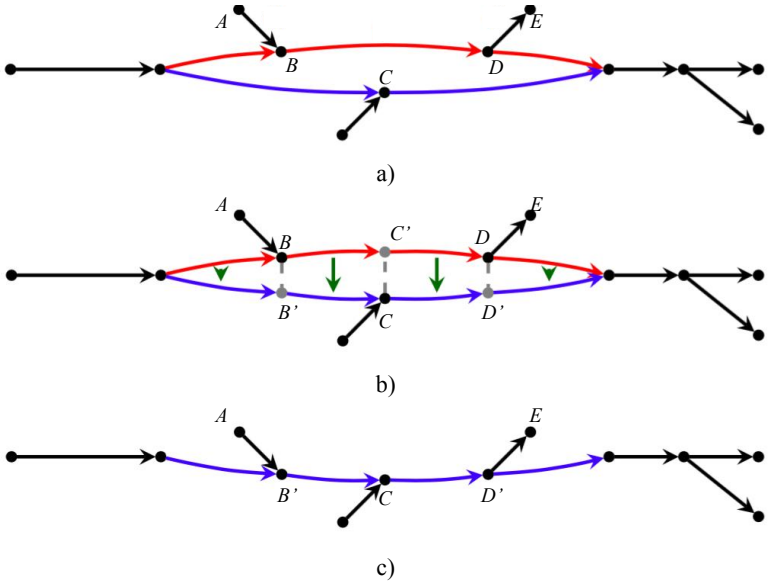
1.8 Present work

This work describes solutions of two bioinformatics problems: assembly of highly polymorphic diploid genomes and construction of adaptive immune repertoires using immunosequencing data. Both of these problems have various biological and biomedical applications.

1.8.1 Assembly of highly polymorphic diploid genomes

Is the problem only heterozygous SNPs? If not clarify here.

Assembly of highly polymorphic diploid genomes is a complicated computational problem. Existing genome assembly approaches typically report extremely frag-



Blue and red paths represent polymorphic regions corresponding to homologous positions on haplomes. a) shows structure of the Bruijn graph corresponding to HP diploid genome. Red and blue polymorphic regions correspond to a pair of alternative paths in the graph. b) shows intermediate step of the polymorphism masking algorithm that searches for masking direction and projects one of alternative paths to other one. In our case, polymorphism masking is directed from red to blue, and the red path will be masked. As a result, algorithm creates projections of intermediate vertices of the red path on the blue path. c) After polymorphism masking, red path is removed from the graph, and all edges incident to its vertices (A and E on our example) will be rerouted to created projections (B' and D') on the blue path.

Figure 2.3 — Algorithm for masking polymorphisms in the de Bruijn graph corresponding to HP diploid genome.

2.4.3 Construcing consensus and double contigs

unclear. reword

unclear

Unclear. Expand.

Haplocontigs constructed at the first step of the dipSPAdes algorithm ideally correspond to paths in the graph before polymorphism masking. So, to mask polymorphisms in haplocontigs one needs to find their paths in the de Bruijn graph with masked polymorphisms. We refer to the resulting contigs as *masked haplocontigs* and acknowledge that polymorphism masking of this type may produce a consensus sequence that belongs to neither $Genome_1$ nor $Genome_2$. After that dipSPAdes searches for overlaps in masked haplocontigs and extends them, thus, improving the quality of assembly. As a result, the algorithm reconstructs double contigs in both haplomes $Genome_1$ and $Genome_2$.

introduced SPAdes* in our benchmarking to illustrate the advantages of dipSPAdes as compared to standard assemblers run in the aggressive polymorphism collapsing mode. While dipSPAdes and HaploMerger can be applied to any set of haplocontigs, in this study we applied them to SPAdes haplocontigs. To generate results for SPAdes, we turned off the procedure of sequencing error removal and removed all edges with low coverage instead. This allows us to avoid errors of sample preparation. We expect Velvet and SPAdes to produce assemblies with total length close to double length of a haplome, and SPAdes*, dipSPAdes, and HaploMerger to produce assemblies with a total length close to the that of a haplome. Benchmarking on both simulated using two *S. commune* haplomes (Table 2) and real fungi (Table 3) datasets (with polymorphism rate varying from 0.4 to 10 percent) demonstrated that dipSPAdes significantly improves assemblies of HP genomes.

Table 2 — Joint assembly of two *Schizophyllum commune* individuals.
Average polymorphism rate is 10 %, estimated haplome size is 38.9 Mbp. HaploMerger failed to produce results on these haplocontigs since it typically requires haplocontigs with N50 exceeding tens of kbp. Columns «ETL» and «CTL» refer to expected total length and computed total length (Mbp), respectively.

	Velvet	**SPAdes**	**HaploMerger**	**SPAdes***	**dipSPAdes**
ETL (Mbp)	77.8	77.8	38.9	38.9	38.9
CTL (Mbp)	39.15	60.33	N/A	45.91	38.36
# contigs	34,406	26,820	N/A	5721	3764
Largest contig	37,580	44,596	N/A	231,443	239,371
N50	1219	3598	N/A	24,931	27,245
N75	761	1694	N/A	8477	11,330

Table 3 — Assembly of diploid genome of *Candida albicans*.
Average polymorphism rate is 0.4 %, estimated haplome size is 14.5 Mbp.

	Velvet	**SPAdes**	**HaploMerger**	**SPAdes***	**dipSPAdes**
ETL (Mbp)	29.0	29.0	14.5	14.5	14.5
CTL (Mbp)	11.28	17.37	2.84	14.85	13.16
# contigs	6731	4007	337	1540	1119
Largest contig	34,870	112,388	92,126	116,985	116,985
N50	2276	8788	23,529	25,691	28,039
N75	1155	3300	8115	10,639	12,809

Chapter 3. Problems of computational immunogenomics

3.1 Adaptive immune system and antibody repertoire

The adaptive immune system is a subsystem of the overall immune system that is composed of highly specialized cells and processes meant to eliminate or prevent pathogen growth. After an initial response adaptive immunity creates an immunological memory to the specific pathogen, which that leads to a fast response to future encounters with that pathogen. Unlike *the innate immune system*, the adaptive immune system is highly specific to the particular pathogens. Adaptive immunity can also provide long-lasting protection: someone who has recovered from measles is protected against this disease for the rest of his or her life. In other cases, however, it does not provide a lifetime worth of protection. An example of this case would be: for example, chickenpox. The adaptive system response destroys the invading pathogens and any toxic molecules they produce. Sometimes the adaptive system is unable to accurately identify foreign molecules, as a result we get such afflictions as hay fever, asthma or other allergies. *Antigens* are any substances that elicit an adaptive immune response. The cells that carry out the adaptive immune response are white blood cells known as *lymphocytes*. The two main broad classes, *antibody responses* and *cell mediated response*, are also provided by two different lymphocytes (*B lymphocytes*/*B cells* and *T lymphocytes*/*T cells*, respectively). In antibody responses, B cells are activated to secrete *antibodies*, proteins that are also known as *immunoglobulins* (antibody structure is shown in Figure 3.1). Antibodies travel through the bloodstream and bind to the foreign antigens causing them to deactivate. Further we will describe all the algorithms in terms of antibodies since they present the most complicated case due to the presence of somatic hypermutagenesis. However, all proposed algorithms can be extended for repertoires of *T cell receptors* (*TCRs*), antigen recognizing products of T cells. The set of all antibodies in the human organism is called an *antibody repertoire*. The diversity of the antibody repertoire is achieved via mechanisms for antibody generation and diversification: V(D)J recombination, heavy and light chains pairing and somatic hypermutagenesis [60–62]. As a result of clonal selection, antibodies have an extremely uneven distribution of multiplicities. For example, several highly abundant antibodies are typical for repertoires that were taken during the immune response.

3.2 Sequencing adaptive immune repertoires

Length of variable region of an antibody (the most interesting part of an antibody for bioinformatics analysis) varies from 350 to 400 nucleotides (Figure 3.1). Complexity of antibody structure is a result of V(D)J recombination that does not allow one to apply genome assembly algorithms for reconstruction of full-length antibody sequences in an antibody repertoire. Thus, in all immunogenomics studies, sequencing technologies allowing unambiguously link each read with antibody were used. However, until recently, there were few attempts to construct full-length antibody repertoires (*full-length classification*) due to limitation of available sequencing technologies. For example, 454 technology used in early studies produces reads covering entire variable region of antibodies, but has extremely high error rate (Table 1). This makes 454 technology inefficient for immunogenomics studies. Moreover, total number of reads produced in a single 454 run is too small for comprehensive repertoire analysis. As a result, 454 reads were mainly used for *V(D)J classification*: finding the closest V, D, and J segments from the database which combination produce variable sequences for a given antibody (Figure 3.2a). Later released Illumina technology had high accuracy, but produced short reads ($75-100$ nt) that were not able to cover variable regions of antibodies. Illumina short reads were used for *CDR3 classification*. CDR3 classification is focused on short region of variable region of antibody: *CDR3* or *third complementarity determining region* (Figure 3.1). CDR3 classification is the more detailed representation of antibody repertoire compared to V(D)J classification. However, CDR3 classification does not allow to distinguish between antibodies that share CDR3s, but differ in other positions of the variable region (Figure 3.2b). Construction of full-length repertoires in turn provides an ideal representation of variable regions of antibodies (Figure 3.2c) and differs from the well-studied VDJ classification [23; 47; 63; 64] and CRD3 classification [65–68] problems. In fact, VDJ classification, CDR3 classification and full-length classification are three different clustering problems with increasing granularity of partition into clusters (from very rough to ideal) and different biological applications (Figure 3.2d). Emergence of MiSeq Illumina sequencing machine in 2013 opened horizons of adaptive immune repertoires investigation using highly accurate 250×2 Illumina reads. Availability of Illumina MiSeq reads raised interest of bioinformaticians to the problem and in 2015 three new tools for solving the issue of repertoire construction were released: IgRepertoireConstructor [3], MiXCR [69] and IMSEQ [70].

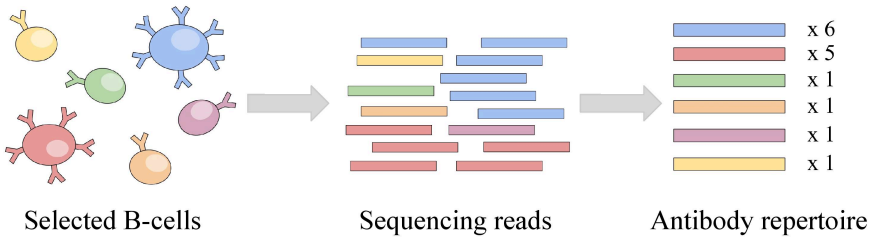
3.3 Repertoire construction problem and IgRepertoireConstructor tool

If we view an antibody as a center of a cluster formed by reads derived from this antibody, then construction of a repertoire corresponds to a difficult clustering problem with many closely located centers so that the radius of a cluster may exceed the distance from one cluster to another one. Since the standard clustering techniques (like k-means clustering) are not applicable to such problems, we have designed IgRepertoireConstructor, a novel algorithm for constructing antibody repertoires.

3.3.1 Representation of full-length antibody repertoire

Each unique antibody in a repertoire is characterized by its sequence and multiplicity; estimated by the number of reads derived from this antibody (Figure 3.3). The complexity of the antibody repertoire mirrors the complexity of the immune system. E.g., clonal selection leads to a highly uneven distribution of multiplicities of antibodies [71]. Abundant antibodies mutate and yield new antibodies that share the same VDJ recombination pattern, but differ only by somatic hypermutations. As a result, the antibody repertoire contains a mixture of closely related antibodies with differing multiplicities. The multiplicities of ≈ 2.3 million antibodies in dataset sequenced after vaccination vary from 1 to $\approx 33{,}000$ with the most abundant antibody representing ≈ 1 % of all reads. The major challenge in constructing antibody repertoires is the identification of all reads that are derived from a single antibody. If the reads were free of errors, we would simply group all the identical (up to small shifts) reads together into unique reads to generate an antibody repertoire. In reality, reads are error-prone, which means that an error-correction step is required before any analysis can be carried out. IgRepertoireConstructor error-corrects reads; partitions them into clusters; and computes the consensus sequence and multiplicity of each antibody.

If free of error, why would you still tolerate shifts? Clarify.



Sequencing process of antibody repertoire includes selection of B cells and transcripts corresponding to antibodies (left) and sequencing variable regions of immunoglobulin RNA molecules (center). As a result, each sequenced unique antibody corresponds to a number of reads in immunosequencing sample (number depends on characteristics of a repertoire and a sample preparation process). An antibody repertoire is constructed as a clustering of reads corresponding to variable regions of antibodies (right). An antibody repertoire is characterized by a set of pairs (s,n). Each pair (s,n) characterizes an unique antibody with sequence of variable region s presented by n reads (*multiplicity* of an antibody). The antibody repertoire provided on the example contains 6 antibodies with different multiplicities. For example, multiplicity of the red antibody is equal to 5.

Figure 3.3 — Sequencing and construcing antibody repertoire.

3.4 IgRepertoireConstructor algorithm

3.4.1 Motivation for IgRepertoireConstructor development

IgRepertoireConstructor uses the idea of the *Hamming graph* proposed in Hammer and BayesHammer tools [72; 73] for error-correction of Illumina genomics reads. *The Hamming distance* $d(s_1,s_2)$ between sequences s_1 and s_2 of equal length is defined as the number of positions where the symbol in s_1 differs from a symbol in s_2. Ideas of algorithms implemented in Hammer and BayesHammer tools are to extract all k-mers from input reads and cluster them using information about Hamming distance between k-mers. Hammer and BayesHammer expect that, for small k-mer size, each genome position will be covered by error-free k-mer. Thus, finding groups of k-mers with relatively small Hamming distance, computation of an error-free k-mer in each group, and correction of remaining k-mers provide one with error-correction of k-mers and, thus, reads. Features of immunoglobulins do not allow us to use Hammer and BayesHammer tools for correction of errors in immunosequencing reads. More specifically, correction

rephrase

position of immunosequencing reads into groups corresponding to identical or highly similar antibodies (Figure 3.4b). The Corrupted Cliques Problem is NP-hard [74] and thus no polynomial algorithm for its solution was proposed. However, the problem has many applications in analysis of biological and social networks, and there exists a number of efficient heuristic algorithms for solving the Corrupted Cliques Problem (such as CAST [75]). But, unfortunately, most of them are not applicable for the Bounded Hamming Graphs with many vertices. Moreover, we would like to use some features of input immunosequencing reads, for example, to derive groups of identical reads with high multiplicity. We, thus, developed a different approach for analyzing the Bounded Hamming Graph that is based on transforming it into a *triangulated graph* (i.e., a graph where every cycle of length longer than three has a chord) using the *Minimum Fill-in Problem* [74] rather than a clique graph. Minimum Fill-in Problem often arises during solving large systems of linear equations and thus has efficient heuristic solutions (e.g., METIS [76]). Finding maximal cliques in a triangulated graph (in our algorithm, constructed by METIS) can be done in a linear time [77]. Maximal cliques of a triangulated graph can be transformed into dense subgraph of the original Bounded Hamming graph. Computed dense subgraphs are turned into antibody clusters by calculating sets of consensus sequences between reads corresponding to the same dense subgraph (Figure 3.5).

Why? Just Size? Specify explicitly.

specify complexity

3.4.3 Immunoproteogenomics search

We use the constructed repertoire as a database for identifying mass spectra that allow us to validate antibody sequences constructed from NGS reads and compute similarity between sequencing and mass spectra data. We acknowledge that immunoproteogenomics searches require new algorithmic and statistical approaches since the standard peptide identification algorithms were not designed for searches in large and highly repetitive immunoproteogenomics databases. We also argue that yet another key difference between the standard and the immunoproteogenomics searches is that, in the latter case, after constructing the antibody repertoire, we have information about antibody abundances. Since higher-abundance antibodies are more promising candidates for spectral searches than lower-abundance antibodies (despite limited correlation between genomics- and proteomics-derived abundances), we partition all antibodies into

multi-layer approach, the controversy about the statistical foundations of the two-stage approach [78] does not extend to our multi-layer approach. It brings the total percentage of identified spectra to ≈ 22.6 % at 1 % FDR (compare to 6 % of spectra at 2 % FDR identified by [33]).

This is not clear or obvious. More needs to be said.

1- With multi-layer approach, do you still guarantee a 1% FDR? I guess not, but I am may be wrong. Specify

2- You are effectively doing four 1% controlled FDRs. Do you not need a meta-correction for the four sets of tests?

3.4.4 Comparison of genomics and proteomics multiplicities

To compare the relation between peptides and their Ig-seq counterparts, we introduce the notion of *total peptide abundance*. Total peptide abundance is the total abundance of antibodies that encode this peptide. Figure 3.6a shows the relation of the total abundance for each peptide to its spectral count (number of PSMs). The remarkable lack of correlations between genomics-based and proteomics-based abundances further amplifies the concern first expressed in [33]. Figure 3.6b shows the correlations between clone abundances measured by MS and immunosequencing data. These plots show the difference when considering the unit of a repertoire (antibody), and the unit of antibody evolution (*the clone*) raising the concern that immunosequencing data do not adequately represent antibody abundances. For the sake of simplicity, we consider that clone can be computed as a set of antibodies sharing CDR3. When considering only the antibodies, there is no correlation with the mass-spectrometry evidence, as previously reported by [33]. However, when considering the amalgam of antibodies forming each clone, a moderate correlation emerges (Pearson's correlation for clone $= 0.5687614$ vs Pearson's correlation $= 0.1724002$ for total peptide abundance). One possible explanation is that certain antibodies, within highly expressed clones, are not captured by mass-spectrometry.

3.5 IgSimulator

The ability to perform full-length and deep scanning of adaptive immune repertoires using next generation sequencing (NGS) have resulted in a growing number of immunoinformatics tools for antibody repertoire analysis [27; 80; 81]. Benchmarking these newly emerging tools, however, remains problematic since the gold standard