12-15-2016

Computational Methods for Sequencing and Analysis of Heterogeneous RNA Populations

Olga Glebova

COMPUTATIONAL METHODS FOR SEQUENCING AND ANALYSIS OF HETEROGENEOUS RNA POPULATIONS

by

OLGA GLEBOVA

Under the Direction of Alexander Zelikovsky, PhD

ABSTRACT

Next-generation sequencing (NGS) and mass spectrometry technologies bring unprecedented throughput, scalability and speed, facilitating the studies of biological systems. These technologies allow to sequence and analyze heterogeneous RNA populations rather than single sequences. In particular, they provide the opportunity to implement massive viral surveillance and transcriptome quantification. However, in order to fully exploit the capabilities of NGS technology we need to develop computational methods able to analyze billions of reads for assembly and characterization of sampled RNA pop-

ulations.

In this work we present novel computational methods for cost- and time-effective analysis of sequencing data from viral and RNA samples. In particular, we describe: i) computational methods for transcriptome reconstruction and quantification; ii) method for mass spectrometry data analysis; iii) combinatorial pooling method; iv) computational methods for analysis of intra-host viral populations.

INDEX WORDS: Next-Generation Sequencing, RNA-sequencing, Transcriptome quantification and reconstruction, Mass spectrometry, Combinatorial pooling, Genetic relatedness, Molecular surveillance, Viral transmission networks

COMPUTATIONAL METHODS FOR SEQUENCING AND ANALYSIS OF HETEROGENEOUS RNA POPULATIONS

by

OLGA GLEBOVA

A Dissertation Submitted in Partial Fulfillment of the Requirements for the Degree of

Doctor of Philosophy

in the College of Arts and Sciences

Georgia State University

2016

COMPUTATIONAL METHODS FOR SEQUENCING AND ANALYSIS OF HETEROGENEOUS RNA POPULATIONS

by

OLGA GLEBOVA

Committee Chair: Alexander Zelikovsky

Committee: Zhipeng Cai
Robert Harrison
Yury Khudyakov

Electronic Version Approved:

Office of Graduate Studies
College of Arts and Sciences
Georgia State University
December 2016

DEDICATION

To my husband Pavel for his encouragement and advice, and to my father Vladimir for his never-endless believing in me and my dreams and supporting my academic pursuits.

ACKNOWLEDGEMENTS

I am very grateful to my advisor Dr. Alex Zelikovsky for his constant support and guidance. He always has time for discussing research, no matter time or day of the week.

I would also like to thank Dr. Yury Khudyakov and his lab at CDC for giving me the opportunity to intern there and to learn technologies, processes, and more importantly people behind it. All that helped me to establish long lasting fruitful collaborations.

Thank you to my professors Dr. Cai, Dr. Harrison, Dr. Song, Dr. Weber for all time and work they do behind-the-scenes to prepare for classes and to make them true learning experiences. The more I go into teaching path myself now the more I remember your classes and appreciate teachers' work and such thoughtful, sometimes sneaky, ways to help students succeed.

I am thankful for all help and support I received from Dr. Raj Sunderraman. I felt a part of GSU family immediately knowing whom to ask if I got lost in all procedures, and for that I am also grateful to Adrienne, Celena, Tammie, and Venette.

Special thanks to Dr. King whom I got to know in his role of advisor to student chapter of ACM at GSU. His calm and gentlemanly ways of handling all kinds of organizational disasters taught me a lot.

Heartfelt thanks to all of my friends in the department who made this experience so much more rewarding: Adrian, Andrew, Andrii, Anjuli, Bassam, Blanche, Daniel, Debraj, Dhara, Guoliang, Igor, Katia, Maryam, Melinda, Meng, Michael, Nick, Lei, Sasha, Sergey, Yanjun, Yuan and Zhiyi.

I thank my family for their loving support throughout my life. I could not have done all this work without it.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

- NGS - Next-Generation Sequencing
- RNA-Seq - RNA-Sequencing
- MALDI-TOF - Matrix-Assisted Laser Desorption/Ionization Time Of Flight
- PPV - Positive Predictive Value
- EM - Expectation Maximization
- DE - Differential Expression
- FPKM - Fragment Per Kilobase of gene length per Million reads
- KEGG - Kyoto Encyclopedia of Genes and Genomes

PART 1

INTRODUCTION

In this work we study algorithmic problems associated with RNA sequencing (RNA-Seq). RNA-Seq is an increasingly popular approach to transcriptome profiling that uses the capabilities of next generation sequencing (NGS) technologies and provides better measurement of levels of transcripts and their frequencies. The algorithmic problems that arise in RNA-Seq are conceptually similar to the problems that are associated with RNA viruses. Indeed, in genomics studies each sequenced sample contains a single genetic variant, whereas in metagenomics studies each sample may contain several substantially different variants. However, in transcriptomics studies most often the intermediate problem arises: it is highly desirable to reconstruct the whole transcriptome, i.e. the set of genetically related and very similar but not identical transcriptome variants. In this work we propose novel algorithms for effective and accurate transcriptome reconstruction and quantification, using integer programming approach for reconstruction and simulated regression method for quantification problem.

In the recent decades molecular biology has been revolutionized by the advent of NGS which delivers many orders of magnitude higher throughput compared to classic Sanger sequencing [5, 6]. Continued advances in NGS technologies now provide the opportunity to implement massive molecular surveillance of viral diseases that will allow to characterize viral strains in tens of thousands of infected individuals. Availability of such large-scale datasets would result in unprecedented progress in our understanding of virus evolution and structures of transmission networks, enabling the development of more effective prevention strategies based on the applications of vaccines and antiviral therapeutics.

In this work we mostly deal with RNA viruses, which include such highly impor-

tant for public health research viruses as HIV and HCV. Due to error prone replication, RNA viruses mutate at average rates estimated to be as high as 10^{-3} substitutions per nucleotide per replication cycle [7]. Since mutations are generally well tolerated, many RNA viruses infecting a host exist as highly heterogeneous populations of closely related sequences commonly referred by virologists as quasispecies [8–12]. Extremely high genetic heterogeneity of intra-host viral populations has major biological implications, contributing to the efficiency of virus transmission, tissue tropism, virulence, disease progression, and emergence of drug/vaccine resistant variants [13–17]. NGS allows sampling viral quasispecies at a great depth [18], and has enabled, e.g., identification of extremely low frequency variants in human patients chronically infected with HIV or HCV [19–24]. The most preferable way of assessment of intra-host viral populations in each sample is analysis of whole-genome sequences. However, NGS usually generates short reads, which should be assembled into whole-genome sequences. Assembly of viral quasispecies and estimation of their frequencies is extremely complex task, and currently even most advanced computational tools for whole-genome quasispecies reconstruction often only allow inference of most prevalent intra-host variants, with minority variants being frequently undetectable [25–29]. Alternatively, genetic viral variants can be detected using highly variable subgenomic regions that can be easily amplified and sequenced. Although genetic information presented in such regions does not allow for identification of all viral variants, it is usually sufficient for inferring transmission networks [30–32], detecting drug-resistant variants, predicting therapy outcome [33–35], and studying intra-host viral evolution [36–38]. In our work we concentrate on analysis of sequences of highly variable genomic regions.

1.1 Sequencing heterogeneous RNA populations

1.1.1 Computational methods for transcriptome quantification and reconstruction

Massively parallel whole transcriptome sequencing and its ability to generate full transcriptome data at the single transcript level provides a powerful tool with multiple interrelated applications, including transcriptome reconstruction [39–42], gene/isoform expression estimation [41, 43–45], also known as transcriptome quantification, studying trans- and cis-regulatory effect [46], studying parent-of origin effect [46–48], and calling expressed variants [49]. As a result, whole transcriptome sequencing has become the technology of choice for performing transcriptome analysis, rapidly replacing array-based technologies [50].

The most commonly used transcriptome sequencing protocol, referred to as RNA-Seq, generates short (single or paired) sequencing tags from the ends of randomly generated cDNA fragments. Using transcriptome sequencing data, most current research employs methods that depend on existing transcriptome annotations. Unfortunately, as shown by recent studies [51], existing transcript libraries still miss large numbers of transcripts. The incompleteness of annotation libraries poses a serious limitation to using this powerful technology since accurate normalization of data critically requires knowledge of expressed transcript sequences [43–45, 52]. Another challenge in transcriptomic analysis comes from the ambiguities in read/tag mapping to the reference. Our research focuses on two main problems in transcriptome data analysis, namely, transcriptome reconstruction and quantification, and we show how these challenges are handled. Transcriptome reconstruction, also referred to as novel isoform discovery, is the problem of reconstructing the transcript sequences from the sequencing data. Reconstruction can be done de novo or it can be assisted by existing genome and transcriptome annotations. Transcriptome quantification refers to the problem of estimating the expression level of each transcript.

1.1.2 Computational inference of genetic relatedness from mass spectrometry viral data

Mass spectrometry (MS) of DNA fragments generated by base-specific cleavage of PCR products is a cost-effective and robust alternative to DNA sequencing. MS is cheaper and less labor-intensive than most of the next-generation sequencing technologies, and also is not prone to the errors characteristic for these technologies. It is is based on matrix-assisted laser desorption/ ionization time-of-flight (MALDI-TOF) analysis of complete base-specific cleavage reactions of a target RNA obtained from PCR fragments [53, 54]. RNA transcripts generated from both strands of PCR fragment are cleaved by RNase A at either U or C, thus querying for every of the 4 nucleotides (A, C, U and G) in separate reactions. Cleavage at any one nucleotide; e.g. U, generates a number of short fragments corresponding to the number of U's in the transcript. The mass and size of the fragments differ based on the number of A, C and G nucleotides residing between the U's that flank each short fragment. The fragments are resolved by MALDI-TOF-MS, resulting in mass spectral profiles, where each peak defines a specific mass measured in Daltons and has intensity that corresponds to the number of molecules of identical masses.

Unlike sequencing, MS is not readily applicable to reconstruction of the genetic composition of DNA/RNA populations. Algorithms for reconstruction of sequences from MS data were proposed [55]; but, owing to technological and computational limitations, none is widely used. Nevertheless, MS has been successfully applied to the reference-guided single nucleotide polymorphism (SNP) discovery [54, 56, 57], genotyping [53, 58], viral transmission detection [59], identification of pathogens and disease susceptibility genes [60, 61], DNA sequence analysis [62], analysis of DNA methylation [63], simultaneous detection of bacteria [64] and viruses [65, 66].

In the case of molecular surveillance of viral diseases MS may serve as a rich source of information about the population structure and the genetic relations among populations without sequences reconstruction. One of the most important applications of sequences is phylogeny. However, construction of phylogenetic trees requires knowledge of genetic

distances among species rather than sequences, with sequences being merely used to estimate the distances. Comparison of MS profiles may also accurately approximate genetic distances. The problem of calculating the distance between two MS samples is known as spectral alignment problem [67,68]. It is usually formulated as follows: match the masses from two MS profiles in such a way that some predefined objective function is maximized or minimized.

We developed a new algorithm for alignment of the base-specific cleavage MS profiles (MS-Al) that is based on the reduction of the problem to the network flow problem. MS-Al allows *de novo* comparison of sampled populations and may be used for phylogenetic analysis and viral transmissions detection.

1.1.3 NGS of large cohorts of viral samples using combinatorial pooling

Although NGS offers a significant increase in throughput, sequencing of a large number of viral samples still is prohibitively expensive and extremely time consuming. Therefore, massive molecular surveillance requires development of a strategy for simple, rapid and cost-effective sequencing of microbial populations from a large number of specimens.

Cost of sequencing of multiple viral samples can be reduced using multiplexing through barcoding. Although this is probably the simplest approach to a simultaneous sequencing of large number of specimen, it requires individual handling of each sample starting from nucleic acid extraction to PCR and library preparation, which increases the sequencing costs [69, 70]. Additionally, bias in amplification of different viral variants using PCR primers with different barcodes may affect distribution of reads [70, 71]. Moreover, maintaining a large library of barcodes is daunting [69, 70].

Combinatorial pooling provides an alternative approach to sequencing costs reduction. Applications of pooling to diagnostic testing goes back to the 1940s [72]. Commonly, it is used for tests producing binary results; e.g., positive or negative, as in group testing [73–76]. Recently, several pooling strategies were proposed for more complex assays based on DNA sequencing, SNP calling and a rare alleles detection [77–82]. In particu-

lar, recent application of combinatorial pooling protocol to selective genome sequencing using NGS [69] should be mentioned.

Pooing strategies for heterogeneous intra-host viral populations sequencing are fundamentally different from other existing pooling protocols. Those protocols assume that a single sequence must be reconstructed for each sample. In contrast, the goal of viral quasispecies sequencing is to reconstruct the whole *quasispecies spectra* that includes multiple sequence variants and their frequencies, including low-frequency variants. It makes the problems of viral quasispecies pool design and pool deconvolution challenging. In particular, the assessment of intra-host viral populations can be distorted by PCR or sampling biases. Thus mixing of a large number of specimens or specimens with significant differences in viral titers may contribute to underrepresentation of viral variants from some samples in pools, suggesting that size and composition of pools should be carefully designed. Stochastic sampling from genetically diverse intra-host viral populations usually produces variability in compositions of sets of variants in different pools obtained from the same sample. Additionally, mixing specimens may differentially bias PCR amplification, contributing to mismatching between viral variants sampled from the same host in two pools with different specimen compositions. Therefore, straightforward approaches cannot be used for samples deconvolution, indicating that a more complex approach based on clustering techniques is needed. To increase the effectiveness of cluster-based deconvolution and minimize possible clustering errors, it is important to minimize mixing of genetically close samples as can be expected in epidemiologically related samples and samples collected from a small geographic region.

We developed a combinatorial pooling pipeline for NGS of viral quasispecies. Our pipeline includes the following steps (Fig. 4.1): (i) mixing samples in a specially designed set of pools so that the identity of each sample is encoded in the composition of pools; (ii) sequencing pools; (iii) pools deconvolution; i.e., assignment of viral variants from the pools to individual samples. This approach allows to significantly reduce the number of PCR and NGS runs, reducing the cost of testing and hands-on time. Our pipeline was

validated using simulated and experimental HCV data.

1.1.4 Computational inference of genetic relatedness, transmission clusters and sources of outbreaks from NGS viral data

Sequencing has already been used for transmission networks inference and outbreaks investigations for Influenza A [83], HIV [84–87], Hepatitis A virus [88,89], Hepatitis B virus [90,91] and HCV [92–94]. However, contribution of sequencing technologies to molecular surveillance of viral infections was not significant so far, being mainly hindered by the lack of reliable computational methods for the inference of transmission networks directly from sequence data, without the need of expert analysis by trained molecular epidemiologist.

Currently transmissions are usually detected either by phylogenetic analysis carried out visually by a humane expert [84–86,88,90–94] or by applying a cutoff on genetic distances between sequences from infected individuals [87]; i.e., two individuals are considered linked by transmission if the genetic distance between the corresponding consensus viral sequences does not exceed a certain value. Although they work well in some cases, such approaches have a number of disadvantages. In particular, it is known that minor variants are often responsible for transmission of HCV infections [95,96]. Transmission of low-frequency variants is most probably associated with the fact that dominant variants in a chronically infected host are highly adapted to the intra-host environment developed during the course of infection, which potentially results in a lower viability in the naive host environment [97]. Such transmissions may not be effectively detected using consensuses sequences. Moreover, distance cutoffs are often either arbitrary or derived from analysis of limited or incomplete experimental data. Cutoffs are highly data- and situation-specific. Different viruses or even different genomic regions of the same virus can be analyzed only using specifically established cutoffs. Moreover, cutoffs tailored to outbreak settings with high prevalence of transmissions may be too strict for surveillance where the detection rate of cases linked by direct transmission is low. In addition

to that, analysis of consensus sequences and genetic-distance cutoff-based methods (even for intra-host populations) does not allow for detecting the direction of transmissions, which is crucial for the identification of outbreak sources and superspreaders. Finally, evolutionary history of viral quasispecies in the host contains important information on viral transmissions. However, phylogenetic trees may not represent intra-host evolution of highly mutable RNA viruses as accurate as network-based approaches reconstructing viral evolution from sets of founders [98].

We developed novel methods for identification of genetic relatedness, transmission clusters and sources of outbreaks, which resolve the aforementioned limitations. Our algorithms address the following problems:

1) Detection of possible transmission links and their directions.

2) Identification of transmission clusters and sources of outbreaks.

1.2 Contributions

We present a novel annotation-guided method for transcriptome discovery and reconstruction in partially annotated genomes and compare it with existing annotation-guided and genome-guided transcriptome assembly methods. Our method, referred as "Discovery and Reconstruction of Unannotated Transcripts" (DRUT) [2], can be used to enhance existing transcriptome assemblers, such as Cufflinks [39]. It was shown that Cufflinks enhanced by DRUT has superior quality of reconstruction and frequency estimation of transcripts. To solve transcriptome reconstruction problem assisted by existing genome annotations

We propose a novel method called "Transcriptome Reconstruction using Integer Programming" (TRIP) [42]. The method incorporates information about fragment length distribution of RNA-Seq paired-end reads to reconstruct novel transcripts.

To estimate isoform frequencies from RNA-Seq data we propose a simulated regression based method (SimReg) [1]. Experiments demonstrate improved frequency estima-

tion accuracy of SimReg comparatively to that of the existing tools which tend to skew the estimated frequency toward super-transcripts.

To assess the genetic relatedness among RNA populations we propose several methods. First, we use mass spectrometry (MS) data which enables an accurate comparison of MS profiles and provides a direct evaluation of genetic distances between RNA molecules without invoking sequences. MS alignments (MSA) may serve as accurately as sequence alignments to facilitate phylogenetic analysis and, as such, has numerous applications in basic research, clinical and public health settings. We formulate and solve MSA as network flow problem.

We propose a cost-effective and reliable protocol for sequencing of viral samples, that combines NGS using both barcoding and pooling and a bioinformatical framework including novel algorithms for optimal virus-specific design of pools and deconvolution of individual samples from sequenced pools. It allows our framework to be readily applicable to highly mutable RNA viruses' data.

1.3 Roadmap

The rest of the dissertation proposal is organized as follows. Chapter 2 presents novel algorithms for transcriptome reconstruction and quantification. In Chapter 3 we describe a framework for measuring genetic distances using mass spectrometry profiles. Chapter 4 presents pooling strategies and the motivation behind it. We first present the state of the art in pooling methods, then we introduce our novel pooling technique optimized for large number of viral samples, and we finish by describing the experimental setup and results. In Chapter 5 we present algorithms for effective detection of viral transmissions and outbreak source identification.

1.4 Related publications

Journal Papers and Book Chapters

1. Serghei Mangul, Adrian Caciula, **Olga Glebova**, Ion Mandoiu, and Alex Zelikovsky, "Improved transcriptome quantification and reconstruction from RNA-Seq reads using partial annotations", In silico biology, Volume: 11, Issue: 5-6, pp. 251-261, 2011.

2. Pavel Skums, Alexander Artyomenko, **Olga Glebova**, Sumathi Ramachandran, Ion Mandoiu, David S Campo, Zoya Dimitrova, Alex Zelikovsky, and Yury Khudyakov, "Computational framework for next-generation sequencing of heterogeneous viral populations using combinatorial pooling", Bioinformatics, issue 31(5), pp. 682-690, 2015.

3. Pavel Skums, **Olga Glebova**, Alex Zelikovsky, Zoya Dimitrova, David Stiven Campo Rendon, Lilia Ganova-Raeva, and Yury Khudyakov,"Alignment of DNA Mass-Spectral Profiles Using Network Flows", In Bioinformatics Research and Applications: 9th International Symposium, ISBRA 2013, Charlotte, NC, USA, May 20-22, 2013, Proceedings, vol. 7875, pp.149 - 160. Springer, 2013.

4. **Olga Glebova**, Yvette Temate-Tiagueu, Adrian Caciula, Sahar Al Seesi, Alexander Artyomenko, Serghei Mangul, James Lindsay, Ion I. Măndoiu and Alex Zelikovsky, "Transcriptome Quantification and Differential Expression from NGS Data", ed. A.Zelikovsky and I. Măndoiu, Wiley, pp. 301-327, 2016.

5. Pavel Skums, Alexander Artyomenko, **Olga Glebova**, Sumathi Ramachandran, David S. Campo, Zoya Dimitrova, Ion Măndoiu, Alex Zelikovsky and Yury Khudyakov, "Pooling Strategy for Massive Viral Sequencing", ed. A.Zelikovsky and I. Măndoiu, Wiley, pp.57-83, 2016.

6. Pavel Skums, Alexander Artyomenko, **Olga Glebova**, David S. Campo, Zoya Dimitrova, Alex Zelikovsky, and Yury Khudyakov, "Error Correction of NGS Reads from Viral Populations", ed. A.Zelikovsky and I. Măndoiu, Wiley, pp.329-353, 2016.

7. **Olga Glebova**, Sergey Knyazev, Andrew Melnick, Alexander Artyomenko, Pavel

Skums and Alex Zelikovsky, "Inference of Genetic Relatedness between Viral Populations", invited paper submitted to Special Issue of BMC Genomics devoted to ISBRA 2016.

Conference Papers

1. Serghei Mangul, Adrian Caciula, Nicholas Mancuso, **Olga Glebova**, Ion Măndoiu, and Alex Zelikovsky, "Short Abstract: An Integer Programming Approach to Novel Transcript Reconstruction from Paired-End RNA-Seq Reads", Proc. of 8th International Symposium on Bioinformatics Research and Applications (ISBRA), May 21-23, 2012, University of Texas at Dallas, Dallas, TX.

2. Pavel Skums, **Olga Glebova**, Alex Zelikovsky, Zoya Dimitrova, David Stiven Campo Rendon, Lilia Ganova-Raeva, and Yury Khudyakov, "Alignment of DNA mass-spectral profiles using network flows", Proceedings of International Symposium on Bioiv nformatics Research and Applications (ISBRA), May 20-23, 2013, Charlotte, NC.

3. Pavel Skums, **Olga Glebova**, Alexander Zelikovsky, Ion Măndoiu, and Yury Khudyakov, "Optimizing pooling strategies for the massive next-generation sequencing of viral samples", Proc. of IEEE International Conference on Computational Advances in Bio and Medical Sciences (ICCABS), 3rd Workshop on Computational Advances for Next Generation Sequencing (CANGS 2013), New Orleans, LA, USA, June 12-14, 2013.

4. Pavel Skums, Alexander Artyomenko, **Olga Glebova**, Alex Zelikovsky, David S Campo, Zoya Dimitrova, and Yury Khudyakov, "Detection of genetic relatedness between viral samples using EM-based clustering of next-generation sequencing data", 2014 IEEE 4th International Conference on Computational Advances in Bio and Medical Sciences (ICCABS), Miami, FL, June 2-4, 2014.

5. Adrian Caciula, **Olga Glebova**, Alexander Artyomenko, Serghei Mangul, James

Lindsay, Ion I Măndoiu, Alex Zelikovsky, "Deterministic regression algorithm for transcriptome frequency estimation", 2014 IEEE 4th International Conference on Computational Advances in Bio and Medical Sciences (ICCABS), Miami, FL, June 2-4, 2014.

6. Adrian Caciula, **Olga Glebova**, Alexander Artyomenko, Serghei Mangul, James Lindsay, Ion I Măndoiu, and Alex Zelikovsky, "Simulated Regression Algorithm for Transcriptome Quantification", Proceedings of Bioinformatics Research and Applications: 10th International Symposium, ISBRA 2014, Zhangjiajie, China, June 28-30, 2014.

7. Yvette Temate-Tiagueu, Meril Mathew, Igor Mandric, Qiong Cheng, **Olga Glebova**, Nicole Beth Lopanik, Ion Măndoiu, and Alex Zelikovsky, "Metabolic pathway activity estimation from RNA-Seq data", short abstract in Proc. of Bioinformatics Research and Applications: 11th International Symposium (ISBRA 2015), Norfolk, VA, June 7-10, 2015.

8. **Olga Glebova**, Sergey Knyazev, Alexander Artyomenko and Alex Zelikovsky, "Simulation-based inference of genetic relatedness between viral populations", 2016 IEEE 6th International Conference on Computational Advances in Bio and Medical Sciences (ICCABS), Atlanta, GA, October 13-15, 2016, submitted.

Posters presentation

1. **Olga Glebova**, Pavel Skums, Zoya Dimitrova, David Stiven Campo Rendon, Lilia Ganova-Raeva, Yury Khudyakov and Alex Zelikovsky, "Estimation of Editing Distance between DNA Sequences Using Network Flows and Mass Spectrometry", The 2013 MBD Graduate Molecular Basis of Disease Graduate Research Day, June 21, 2013, Georgia State University, Atlanta GA ("Best poster" award).

2. Yvette Temate-Tiagueu, Meril Mathew, Adrian Caciula, Sahar Al Seesi, **Olga Glebova**, Nicole Beth Lopanik, Ion Măndoiu and Alex Zelikovsky "Bioinformatics

Analysis of RNA-Seq Data for Bugula neritina", 4th IEEE International Conference on Computational Advances in Bio and Medical Sciences (CANGS 2014).

3. Adrian Caciula, **Olga Glebova**, Alexander Artyomenko, Serghei Mangul, James Lindsay, Ion I Măndoiu, and Alex Zelikovsky, "Deterministic Regression Algorithm for Transcriptome Frequency Estimation" at 2014 Molecular Basis of Disease Research Day, June 13th, 2014.

PART 2

TRANSCRIPTOME RECONSTRUCTION AND QUANTIFICATION FROM NGS DATA

2.1 RNA-Seq protocol

RNA sequencing (RNA-Seq) is a widely used cost-efficient technology with several medical and biological applications. This technology, however, presents scholars with a number of computational challenges. RNA-Seq protocol provides full transcriptome data at a single transcript level.

RNA-Seq is an increasingly popular approach to transcriptome profiling that uses the capabilities of next generation sequencing (NGS) technologies and provides better measurement of levels of transcripts and their isoforms. One issue plaguing RNA-Seq experiments is reproducibility. This is a central problem in bioinformatics in general. It is not easy to benchmark the entire RNA-seq process [99], and the fact that there are fundamentally different ways of analyzing the data (assembly, feature counting, etc) make it more difficult. Nevertheless RNA-Seq offers huge advantages over microarrays since there is no limit on the numbers of genes surveyed, no need to select what genes to target, and no requirements for probes or primers and it is the tool of choice for metagenomics studies. Also, RNA-seq has the ability to quantify a large dynamic range of expression levels, this lead to transcriptomics and metatranscriptomics.

Rapid advances in NGS have enabled shotgun sequencing of total DNA and RNA extracted from complex microbial communities, ushering the new fields of metagenomics and metatranscriptomics. Depending on surrounding conditions e.g. food availability, stress or physical parameters, the gene expression of organisms can vary widely. The aim of transcriptomics is to capture the gene activity. Transcriptomics helps perform gene expression profiling to unravel gene functions. It can tell us, which metabolic pathways are

in use under the respective conditions and how the organisms interact with the environment. Hence, it can be applied for environmental monitoring and for the identification of key genes. Transcriptomics also play a role in clinical diagnosis and in screening for drug targets or for genes, enzymes and metabolites relevant for biotechnology [100–102].

While transcriptomics deals with the gene expression of single species, metatranscriptomics covers the gene activity profile of the whole microbial community. Metatranscriptomics studies changes in the the function and structure of complex microbial communities as it adapts to environments such as soil and seawater. Unfortunately, as in all "meta" approaches, only a small percentage of the vast number of ecologically important genes has been correctly annotated [103].

Here, we apply RNA-Seq protocol and transcriptome quantification to estimate gene expression and differential gene expression analysis.

RNA-Seq, or deep sequencing of RNAs, is a cost-efficient high-coverage powerful technology for transcriptome analysis. There are various tools and algorithms for RNA-Seq data analysis devoted to different computational challenges, among them transcriptome quantification and reconstruction. We focus on the problem of transcriptome quantification, i.e. on the estimation the expression level of each transcript.

2.2 Transcriptome reconstruction from RNA-seq reads

2.2.1 Related work

RNA-Seq analyses typically start by mapping sequencing reads onto the reference genome, reference annotations, exon-exon junction libraries, or combinations thereof. In case of mapping reads onto the reference genome one needs to use spliced alignment tools, such as TopHat [104] or SpliceMap [105].

Identifying of all transcripts expressed in a particular sample require the assembly of reads into transcription units. This process is collectively called transcriptome reconstruction. A number of recent works have addressed the problem of transcriptome reconstruc-

tion from RNA-Seq reads. These methods fall into three categories: “genome-guided", “genome-independent" and “annotation-guided" methods [106]. Genome-independent methods such as Trinity [107] or transAbyss [108] directly assemble reads into transcripts. A commonly used approach for such methods is de Brujin graph [109] utilizing "k-mers". The use of genome-independent methods becomes essential when there is no trusted genome reference that can be used to guide reconstruction. On the other end of the spectrum, annotation guided methods [110] make use of available information in existing transcript annotations to aid in the discovery of novel transcripts. RNA-Seq reads can be mapped onto reference genome, reference annotations, exon-exon junction libraries, or combinations thereof, and the resulting alignments are used to reconstruct transcripts.

Many transcriptome reconstruction methods fall in the genome-guided category. They typically start by mapping sequencing reads onto the reference genome,using spliced alignment tools, such as TopHat [104] or SpliceMap [105]. The spliced alignments are used to identify exons and transcripts that explain the alignments. While some methods aim to achieve the highest sensitivity, others work to predict the smallest set of transcripts explaining the given input reads. Furthermore, some methods aim to reconstruct the set of transcripts that would insure the highest quantification accuracy. Scripture [40] construct a splicing graph from the mapped reads and reconstructs isoforms corresponding to all possible paths in this graph. It then uses paired-end information to filter out some transcripts. Although scripture achieves very high sensitivity, it may predict a lot of incorrect isoforms. The method of Trapnell et al. [39, 111], referred to as Cufflinks, constructs a read overlap graph and generates candidate transcripts by finding a minimal size path cover via a reduction to maximum matching in a weighted bipartite graph. Cufflinks and Scripture do not target the quantification accuracy. IsoLasso [41] uses the LASSO [112] algorithm, and it aims to achieve a balance between quantification accuracy and predicting the minimum number of isoforms. It formulates the problem as a quadratic programming one, with additional constraints to ensure that all exons and junctions supported by the reads are included in the predicted isoforms. CLIIQ [113] uses

Table (2.1) Classification of transcriptome reconstruction methods

Method	Support paired-end reads	Consider fragment lenght distribution	Require annotation
TRIP	Yes	Yes	No
IsoLasso	Yes	No	No
IsoInfer	No	No	TES/TSS
Cufflinks	Yes	Yes	No
CLIQ	No	No	No
Scripture	Yes	No	No
SLIDE	Yes	No	gene/exon boundaries

an integer linear programming solution that minimizes the number of predicted isoforms explaining the RNA-Seq reads while minimizing the difference between estimated and observed expression levels of exons and junctions within the predicted isoforms.

Table 2.1 includes classification of the available methods for genome-guided transcriptome reconstruction based on supported parameters and underlying algorithms.

2.2.2 An integer programming approach to novel transcript reconstruction from paired-end RNA-seq reads

The common applications of RNA-seq are gene expression level estimation (GE), transcript expression level estimation (IE) [3] and novel transcript reconstruction (TR). A variety of new methods and tools have been recently developed to tackle these problems. In this work, we propose a novel statistical "genome-guided" method called "Transcriptome Reconstruction using Integer Programming" (TRIP) that incorporates fragment length distribution into novel transcript reconstruction from paired-end RNA-Seq reads. To reconstruct novel transcripts, we create a splice graph based on exact annotation of exon boundaries and RNA-Seq reads. A splice graph is a directed acyclic graph (DAG), whose vertices represent exons and edges represent splicing events. We enumerate all maximal paths in the splice graph using a depth-first-search (DFS) algorithm. These paths correspond to putative transcripts and are the input for the TRIP algorithm.

2.2.3 Transcriptome quantification and reconstruction using partial annotations

In this section, we propose a novel annotation-guided algorithm called "**D**iscovery and **R**econstruction of **U**nannotated **T**ranscripts"(DRUT) [114] for transcriptome discovery, reconstruction and quantification in partially annotated genomes. DRUT incorporates VTEM algorithm to detect overexpressed segments corresponding to the unannotated transcripts and to estimate transcriptome frequencies. In case rVTEM algorithm is used, segments represent reads corresponding to unannotated transcripts. eVTEM algorithm requires one additional step, to select reads corresponding to overexpressed exons. Henceforth we will refer to these reads as overexpressed reads. Spliced read is selected only in the case when it entirely belongs to the “overexpressed" exons.

In this way we add the mapped reads to a new read alignment file (e.g., sam file) that represents a subset of original reads. This subset of reads is merged with reads that failed to map to annotated transcripts. Only reads that failed to map to annotated transcripts are now mapped to the reference genome using spliced alignment tools, e.g. TopHat [104] (see Fig. 2.1c). Merged subsets of reads are used as an input for transcriptome assembler. For DRUT framework we chose Cufflinks [39] as ab initio transcriptome reconstruction tool. Assembled transcripts are merged with annotated transcripts and the resulting set of transcripts is filtered to remove duplicates (see Fig. 2.1d). Finally DRUT reports full set of transcripts and maximum likelihood frequencies of transcripts that the best explain reads.

2.2.4 Experimental Results.

Our validation of DRUT includes three experiments over human RNA-seq data, two experiments on transcriptome quantification and one experiment on transcriptome discovery and reconstruction. Below we describe the transcriptome data and read simulation and then give the settings for the each experiment and analyze the obtained experimental results.

2.2.4.1 Simulated human RNA-Seq data. The human genome data (hg19, NCBI build 36) was downloaded from UCSC [115] and CCDS [116], together with the coordinates of the transcripts in the KnownGenes table. The UCSC database contains a total of 66, 803 transcripts pertaining to 19, 372 genes, and CCDS database contains 20, 829 transcripts from 17, 373 genes. The transcript length distribution and the number of transcripts per genes for UCSC are shown in Fig. 2.2. Genes were defined as clusters of known transcripts as in GNFAtlas2 table, such that CCDS data set can be identified with the subset of UCSC data set. 30 millions single reads of length 25bp were randomly generated by sampling fragments of transcripts from UCSC data set. Each transcript was assigned a true frequency based on the abundance reported for the corresponding gene in the first human tissue of the GNFAtlas2 table, and a probability distribution over the transcripts inside a gene cluster [45]. We simulate datasets with geometric (p=0.5) distributions for the transcripts in each gene.

Single error-free reads of length 25bp, 50bp, 100bp and 200bp were randomly generated by sampling fragments of transcripts from UCSC data set. As shown in the [45] for transcriptome quantification purposes it is more beneficial to have shorter reads if the throughput is fixed. At the same time, for transcriptome reconstruction is quite beneficial to have longer reads. Read length of 100bp is the best available option for such next generation sequencing platform as Illumina™ [117]. Current Ion Torrent™technology is capable of producing reads of length more than 200bp. Ion Torrent™next generation sequencing technology utilizes integrated circuits capable of detection ions produced by the template-directed DNA polymerase synthesis for sequencing genomes [118].

2.2.4.2 Accuracy Estimation Transcriptome Quantification Accuracy was assessed using *error fraction (EF)* and *median percent error (MPE)* measures used in [119]. However, accuracy was computed against true frequencies, not against estimates derived from the true counts as in [119]. If $\hat{f}_i$ is the frequency estimate for an transcript with true frequency f_i, the *relative error* is defined as $|\hat{f}_i - f_i|/f_i$ if $f_i \neq 0$, 0 if $\hat{f}_i = f_i = 0$, and ∞ if $\hat{f}_i > f_i = 0$.

The error fraction with threshold τ, denoted EF_τ is defined as the percentage of transcripts with relative error greater or equal to τ. The median percent error, denoted MPE, is defined as the threshold τ for which $EF_\tau = 50\%$.

To estimate transcriptome reconstruction accuracy all assembled transcripts (referred to as "candidate transcripts") are matched against annotated transcripts. Two transcripts match if and only if they include the same set of exons. Two single-exon transcripts match if and only if the overlapping area is at least 50% the length of each transcript.

Following [26], we use sensitivity and Positive Predictive Value (PPV) to evaluate the performance of different methods. Sensitivity is defined as portion of the annotated transcript sequences being captured by candidate transcript sequences as follows:

$$Sensitivity = \frac{TP}{TP + FN}$$

PPV is defined portion of annotated transcript sequences among candidate sequences as follows:

$$PPV = \frac{TP}{TP + FP}$$

2.2.4.3 Comparison on partially annotated UCSC data set. We assumed that in every gene 25% of transcripts are not annotated. In order to create such an instance we assign to the transcripts inside the gene a geometric distribution (p=0.5), assuming a priori that number of transcripts inside the gene is less or equal to 3, we will refer to this experiment as Experiment 1. This way we removed transcripts with frequency 0.25. As a result 11, 339 genes were filtered out, number of transcripts was reduced to 24, 099. Note that in our data set unannotated transcripts do not have unique exon-exon junctions that can emit reads indicating that certain transcripts are not annotated.

We first check how well VTEM estimates the volume of missing transcripts. Although the frequencies of all missing transcripts are the same (25%), the volumes significantly differ because they have different lengths. Therefore, the quality can be measured by correlation between actual unannotated volumes and predicted missing vol-

umes, which represent volumes of virtual transcripts. In this experiment the quality is 61% which is sufficiently high to give an idea which genes have unannotated transcripts in the database.

Table 2.2 reports the median percent error (MPE) and .15 error fraction $EF_{.15}$ for the isoform expression levels inferred from 30 millions reads of length 25bp, computed over groups of isoforms with various expression levels.

Figure 2.3 gives the error fraction at different thresholds ranging between 0 and 1. Clearly the best performance is achieved when the genome is completely annotated, in which case IsoEM and VTEM (rVTEM and eVTEM) show similar results. This happens due to the fact that the frequency of virtual transcript is not increasing over iterations of VTEM. In case of partial annotated genome using virtual transcript allows rVTEM to achieve better results comparative to IsoEM. eVTEM has worse performance than other methods, the reason is that it uses simplified model based on exons rather than on reads, as is done in IsoEM and rVTEM.

Table (2.2) Median percent error (MPE) and 15% error fraction ($EF_{.15}$) for isoform expression levels in Experiment 1.

	Expression range	0	$(0, 10^{-6}]$	$(10^{-6}, 10^{-5}]$	$(10^{-5}, 10^{-4}]$	$(10^{-4}, 10^{-3}]$	$(10^{-3}, 10^{-2}]$	All
	Complete annotations:							
	IsoEM, rVTEM, eVTEM	0.0	61.7	22.0	8.0	3.2	2.1	10.3
MPE	**Partial annotations:**							
	IsoEM	0.0	59.3	41.3	24.8	19.7	5.9	33.7
	rVTEM	0.0	47.2	33.1	20.7	16.4	8.5	26.9
	eVTEM	0.0	60.5	45.1	25.2	22.1	9.1	35.3
	Complete annotations:							
	IsoEM, rVTEM, eVTEM	0.0	81.9	61.3	28.7	7.5	8.5	38.8
$EF_{.15}$	**Partial annotations:**							
	IsoEM	0.0	81.7	72.4	61.4	56.7	42.1	67.6
	rVTEM	0.0	77.2	68.2	57.6	53.0	36.8	63.6
	eVTEM	0.0	82.8	75.6	64.7	59.2	44.4	70.1

2.2.4.4 Comparison on on CCDS data set. In this experiment, referred as Experiment 2, UCSC data set represents the complete set of transcripts and CCDS data set

represents the partially annotated set of transcripts. Reads were generated from UCSC annotations, while only frequencies of the known transcripts from the CCDS database were estimated. In contrast to Experiment 1, we do not control the frequency of unannotated transcripts (i.e. transcripts from UCSC which are absent in CCDS). Therefore, one cannot expect as good improvements as in Experiment 1.

Table 2.3 reports the median percent error (MPE) and .15 error fraction $EF_{.15}$ for isoform expression levels inferred from 30 millions reads of length 25bp, computed over groups of isoforms with various expression levels. We do not report the number of transcripts since they are different for UCSC and CCDS panels. Anyway, one can see a reasonable improvement in frequency estimation of rVTEM over IsoEM.

Table (2.3) Median percent error (MPE) and 15% error fraction ($EF_{.15}$) for isoform expression levels in Experiment 2.

	Expression range	0	$(0, 10^{-6}]$	$(10^{-6}, 10^{-5}]$	$(10^{-5}, 10^{-4}]$	$(10^{-4}, 10^{-3}]$	$(10^{-3}, 10^{-2}]$	All
	Complete annotations:							
	IsoEM, rVTEM, eVTEM	0.0	100	22.7	7.3	3.5	2.5	11.8
MPE	**Partial annotations:**							
	IsoEM	0.0	100	45.5	29.4	28.5	28.7	31.8
	rVTEM	0.0	100	43.2	27.1	25.7	14.3	29.6
	eVTEM	0.0	100	46.3	32.2	33.2	32.1	34.6
	Complete annotations:							
	IsoEM, rVTEM, eVTEM	5.1	91.2	62.8	29.3	15.8	7.6	45.5
$EF_{.15}$	**Partial annotations:**							
	IsoEM	18.6	95.6	85.6	83.3	89.2	86.7	80.0
	rVTEM	17.6	91.8	81.3	77.9	80.3	75.5	75.2
	eVTEM	19.5	97.4	89.2	87.7	88.3	87.9	82.3

2.2.4.5 Comparison Between DRUT, RABT and Cufflinks. In order to simulate a partially annotated genome we removed from every gene exactly one transcript. As a result all 19, 372 genes become partially annotated, and number of transcripts was reduced to 47, 431. In this section, we use the sensitivity and PPV defined above to compare our DRUT method to the most recent version of Cufflinks and RABT (version 1.3.0 of Cufflinks and RABT downloaded from website http://cufflinks.cbcb.umd.edu/). Due to the

fact that results on 100bp and 200bp are very similar, we decided to present comparison on reads of length 100bp. TopHap [104] is used for Cufflinks and RABT to map simulated reads to the reference genome. For DRUT we used Bowtie [120] to map reads to the set of annotated transcripts. For our simulation setup we assume perfect mapping of simulated reads to the genome in case of Cufflinks and to the annotated transcripts in case of DRUT.

Intuitively, it seems more difficult to predict the transcripts in genes with more transcripts. Following [121] we group all the genes by their number of transcripts and calculate the sensitivity and PPV of the methods on genes with certain number of transcripts as shown in Fig. 2.4.

Next we want to define the portion of known transcripts that is input for annotation-guided methods as “existing annotations". Please note that sensitivity of annotation-guided methods needs to be compared to the “existing annotations" ratio unlike regular reconstruction methods that do not have any a priori information about annotated transcripts. In our simulation setup “existing annotations" ratio increases as the number of transcripts in genes become larger.

Fig. 2.4(a) shows that for genes with more transcripts it is more difficult to correctly reconstruct all the transcripts. As a result Cufflinks performs better on genes with few transcripts since annotations are not used in it standard settings. DRUT has higher sensitivity on genes with 2 and 3 transcripts, but RABT is better on gene with 4 transcripts. For genes with more than 4 transcripts performance of annotation-guided methods is equal to "existing annotations ratio", which means these methods are unable to reconstruct unannotated transcripts.

We compared PPV for all 3 methods (Fig. 2.4(b)), all methods show high PPV for genes with 2 transcripts. DRUT outperforms all methods on genes with more then 3 transcripts and shows comparable performance on gene with 2 and 3 transcripts.

TRIP is a novel “genome-guided" method that incorporates fragment length distribution into novel transcript reconstruction from paired-end RNA-Seq reads. The method starts from a set of maximal paths corresponding to putative transcripts and selects the

subset of candidate transcript with the highest support from the RNA-Seq reads. We formulate this problem as an integer program. The objective is to select the smallest set of putative transcripts that yields a good statistical fit between the fragment length distribution empirically determined during library preparation and fragment lengths implied by mapping read pairs to selected transcripts.

2.2.4.6 Construction of Splice Graph and Enumeration of Putative Transcripts. Typically, alternative variants occurs due alternative transcriptional events and alternative splicing events [122]. Transcriptional events include: alternative first exon, alternative last exon. Splicing events include: exon skipping, intron retention, alternative 5′ splice site(A5SS), and alternative 3′ splice site (A3SS). Transcriptional events may consist only of non-overlapping exons. If exons partially overlap and both serve as a first or last exons we will refer to such event as A5SS or A3SS respectively.

To represent such alternative variants we suggest to process the gene as a set of so called “pseudo-exons" based on alternative variants obtained from aligned RNA-seq reads. A *pseudo-exon* is a region of a gene between consecutive transcriptional or splicing events, i.e. starting or ending of an exon, as shown in Figure 2.5. Hence every gene has a set of non-overlapping pseudo-exons, from which it is possible to reconstruct a set of putative transcripts.

The notations used in Figure 2.5 represents the following:

e_i : exon i ;

pse_j : pseudo-exon j ;

S_{pse_j} : start position of pseudo-exon j, $1 \leq j \leq 2n$;

E_{pse_j} : end position of pseudo-exon j, $1 \leq j \leq 2n$;

Tr_i : transcript i ;

A splice graph is a directed acyclic graph (see Fig. 2.6), whose vertices represent pseudo-exons and edges represent pairs of pseudo-exons immediately following one another in at least one transcript (which is witnessed by at least one (spliced) read). We enumerate all maximal paths in the splice graph using a depth-first-search algorithm. These

paths correspond to putative transcripts and are the input for the TRIP algorithm. A gene with n pseudo-exons may have $2^n - 1$ possible candidate transcripts, each composed of a subset of the n pseudo-exons.

Next we will introduces an integer program producing minimal number of transcripts sufficiently well covering observed paired reads.

2.2.4.7 Integer Program Formulation. The following notations are used in the Integer Program (IP) formulation :

N	Total number of reads ;
J_l	l-th splice junction;
p_j	paired-end read, $1 \leq j \leq N$;
t_k	k-th candidate transcript , $1 \leq k \leq K$;
s_i	Expected portion of reads mapped within i standard deviations ($s_1 \approx 68\%$, $s_2 \approx 95\%$, $s_3 \approx 99.7\%$);
ϵ	allowed deviation from the rule ($\epsilon = 0.05$)
$T_i(p_j)$	Set of candidate transcripts where p can be mapped with a fragment length between $i-1$ and i standard deviations, $1 \leq i \leq 3$;
$T_4(p_j)$	Set of candidates transcripts where p_j can be mapped with a fragment length within more than 3 standard deviations;

For a given instance of the transcriptome reconstruction problem, we formulate the integer program.

$$\sum_{t_k \in T} y(t) \rightarrow min$$

where the boolean variables are:

$y(t_k) =$	1 if candidate transcript t_k is selected, and 0 otherwise;
$x_i(p_j) =$	1 if the read p_j is mapped between $i-1$ and i standard deviations, and 0 otherwise;

The IP objective is to minimize the number of candidate transcripts subject to the constraints (1) through (4).

Subject to

(1) $\sum_{t_k \in T_i(p)} y(t) \geq x_i(p), \forall p, i = \overline{1,4}$

(2) $N(s_i - \epsilon) \leq \sum_j x_i(p_j) \leq N(s_i + \epsilon), i = \overline{1,4}$

(3) $\sum_i x_i(p) \leq 1, \forall p$

(4) $\sum_{t_k \in J_l} y(t) \geq 1, \forall J_l$

Constraint (1) implies that for each paired-end read $p \in n(s_i)$, at least one transcript $t \in T_i(p_j)$ is selected. Constraint (2) restricts the number of paired-end reads mapped within every category of standard deviation. Constraint (3) ensures that each paired-end read p_j is mapped no more than with one category of standard deviation. Finally, constraint (4) requires that every splice junction to be present in the set of selected transcripts at least once.

2.2.4.8 Maximum Likelihood Integer Programming Solution. Here we introduce 2-step approach for novel transcript reconstruction from single-end RNA-Seq reads. First, we introduce the integer program (IP) formulation, which has an objective to minimize number of transcripts sufficiently well covering observed reads. Since such formulation can lead to many identical optimal solutions we will use the additional step to select maximum likelihood solution based on deviation between observed and expected read frequencies. As with many RNA-Seq analyses, the preliminary step of our approach is to map the reads. We map reads onto the genome reference using any of the available splice alignment tools (we use TopHat [104] with default parameters in our experiments).

1st step : Integer Program Formulation:

We will use the following notations in our IP formulation:

N total number of candidate ;

R total number of reads ;

J_l l-th spliced junction;

P_l l-th poly-A site(PAS);

r single-read, $1 \leq j \leq R$;

t candidate transcript , $1 \leq k \leq K$;

T set of candidate transcripts

$T(r)$ set of candidate transcripts where read r can be mapped

For a given instance of the transcriptome reconstruction problem, we formulate the IP. The boolean variables used in IP formulation are:

$x(r \rightarrow t)$ 1 iff read r is mapped into transcript t and 0 otherwise;

$y(t)$ 1 if candidate transcript t is selected, and 0 otherwise;

$x(r)$ 1 if the read r is mapped , and 0 otherwise;

The IP objective is to minimize the number of candidate transcripts subject to the constraints (1)-(5):

$$\sum_{t \in T} y(t) \rightarrow min$$

Subject to:

(1) For any r, at least one transcript t is selected: $y(t) \geq x(r \rightarrow t), \forall r, \forall t$

(2) Read r can be mapped only to one transcript: $\sum_{t \in T(r)} x(r \rightarrow t) = x(r), \forall r$

(3) Selected transcripts cover almost all reads: $\sum_{r \in R} x(r) \geq N(1 - \epsilon)$

(4) Each junction is covered by at least one selected transcript: $\sum_{t \in J_l} y(t_k) \geq 1, \forall J_l$

(5) Each PAS is covered by at least one selected transcript: $\sum_{t_k \in P_l} y(t_k) \geq 1, \forall P_l$

We use CPLEX [123] to solve the IP, the rest of implementation is done using Boost C++ Libraries and bash scripting language.

2nd step : Maximum Likelihood Solution:

In the second step we enumerate all possible subsets of candidate transcripts of size N, where N is determined by solving transcriptome reconstruction IP, that satisfy the following condition: every spliced junction and PAS to be present in the subset of transcripts at least once. Further, for every such subset we estimate the most likely transcript frequencies and corresponding expected read frequencies. The algorithm chooses subset with the smallest deviation between observed and expected read frequencies.

The model is represented by bipartite graph $G = \{T \bigcup R, E\}$ in which each transcript is represented as a vertex $t \in T$, and each read is represented as a vertex $r \in R$. With each vertex $t \in T$, we associate frequency f of the transcript. And with each vertex $r \in R$, we associate observed read frequency o_r. Then for each pair t, r, we add an edge (t, r) weighted by probability of transcript t to emit read r.

Given the model we will estimate maximum likelihood frequencies of the transcripts using our previous approach, refer as IsoEM [45]. Regardless of initial conditions IsoEM algorithm always converge to maximum likelihood solution (see [124]).The algorithm starts with the set of T transcripts. After uniform initialization of frequencies $f_t, t \in T$, the algorithm repeatedly performs the next two steps until convergence:

- E-step: Compute the expected number $n(t_k)$ of reads that come from transcript t_k under the assumption that transcript frequencies $f(t)$ are correct, based on weights h_{t_k, r_j}
- M-step: For each t_k, set the new value of f_t to the portion of reads being originated by transcript t among all observed reads in the sample

We suggest to measure the model quality, i.e. how well the model explains the reads,

by the deviation between expected and observed read frequencies as follows:

$$D = \frac{\sum_j |o_j - e_j|}{|R|}, \tag{2.1}$$

where $|R|$ is number of reads, o_j is the observed read frequency of the read r_j and e_j is the expected read frequencies of the read r_j calculated as follows:

$$e_j = \sum_{r_j} \frac{h_{t_k,r_j}}{\sum_{r_j} h_{t_k,r_j}} f_t^{ML} \tag{2.2}$$

where h_{t_k,r_j} is weighted match based on mapping of read r_j to the transcript t_k and f_t^{ML} is the maximum-likelihood frequency of the transcript t_k.

The flowchart of MLIP is depicted in figure 2.7.

Figure 2.8 illustrates how MLIP works on a given synthetic gene with 3 transcripts and 7 different exons (see figure 2.8-A). First we use mapped reads to construct the splice graph from which we generate T possible candidate transcripts, as shown in figure 2.8-B. Further we run our IP approach to obtain N minimum number of transcripts that explain all reads. We enumerate N feasible subsets of candidate transcripts.The subsets which doesn't cover all junctions will be excluded from consideration. The subset with the smallest deviation between expected and observed read frequencies is selected by the MLIP algorithm.

2.2.4.9 Stringency of Reconstruction. Different level of stringency corresponds to different strategies of transcriptome reconstruction. High stringency has the goal to optimize precision of reconstruction, with some loss in sensitivity. On the other hand, low stringency corresponds to increase in sensitivity and some decrease in prediction. Medium stringency strikes balance between sensitivity and precision of reconstruction. The medium stringency is chosen as a default setting for the proposed MLIP method.

Below, we will describe how different stringency levels are computed. For the default medium level we will use the subset of candidate transcripts selected based on the

smallest deviation between observed and expected read frequency. For the low stringency level, our method selects the subset of transcripts that will correspond to the union of the solution obtained by solving the IP and the solution supported by the smallest deviation. High stringency level will correspond to the intersection of above solutions.

Influence of Sequencing Parameters. Although high-throughput technologies allow users to make trade-offs between read length and the number of generated reads, very little has been done to determine optimal parameters for fragment length. Additionally, novel Next Generation Sequencing (NGS) technologies such as Ion Torrent may allow to learn exact fragment length. For the case when fragment length is known, we have modified TRIP's IP referring to this new method as TRIP-L.

In this section we compare methods TRIP-L, TRIP and Cufflinks for the mean fragment length 500bp and variance of either 50bp or 500bp, to check how the variance affects the prediction quality. Figures 2.9(a)-2.9(c) compare sensitivity, PPV and F-score of five methods (TRIP-L 500,500; TRIP-L 500,50; TRIP 500,50; Cufflinks 500,500; Cufflinks 500,50) on simulated data. The results show that as before TRIP has a better sensitivity and F-score while TRIP-L further improves them. Also higher variation in fragment length actually improves performance of all methods.

Results on Real RNA-Seq Data. We tested TRIP on real RNA-Seq data that we sequenced from a CD1 mouse retina RNA samples. We selected a specific gene that has 33 annotated transcripts in Ensembl. The gene was picked and validated experimentally due to interest in its biological function. We plan to have experimental validation at a larger scale in the future. The read alignments falling within the genomic locus of the selected gene were used to construct a splicing graph; then candidate transcripts were selected using TRIP. The dataset used consists of 46906 alignments for 22346 read pairs with read length of 68. TRIP was able to infer 5 out of 10 transcripts that we confirmed using qPCR. For comparison, we ran the same experiment using cufflinks, and it was able to infer 3 out of 10.

In order to explore influence of coverage on precision and sensitivity of reconstruc-

Table (2.4) Transcriptome reconstruction results

Coverage	Read Length	Fragment Length	Methods	Number of reconstructed transcripts	Number of identified annotated transcripts	Sensitivity (%)	Precision (%)	F-Score (%)
20X	100	250	Cufflinks	21803	16519	66.77	75.76	70.98
			MLIP	23351	18412	74.46	78.85	76.59
			IsoLasso	21021	15209	60.66	72.35	65.99
	400	450	Cufflinks	20958	16443	59.78	78.46	67.86
			MLIP	25592	20069	75.39	78.42	76.88
			IsoLasso	13241	9684	37.32	73.14	49.42
100X	100	250	Cufflinks	17981	14073	69.30	78.27	73.51
			MLIP	19405	15539	76.72	80.08	78.36
			IsoLasso	16864	12802	62.60	75.91	68.62
	400	450	Cufflinks	18582	12909	51.06	69.47	58.86
			MLIP	23706	18698	76.69	78.87	77.77
			IsoLasso	21441	15693	63.52	73.19	68.02

tion we simulated 2 datasets with 100X and 20X coverage. Table 2.4 shows how accuracy of transcriptome reconstruction depends on the coverage. For all methods higher coverage (100X vs. 20X) doesn't provide significant improvement in precision and sensitivity.

2.3 Transcriptome quantification

In this chapter we focus on the transcriptome quantification problem, which is to estimate the expression level of each transcript. Transcriptome quantification analysis is crucial to determine similar transcripts or unraveling gene functions and transcription regulation mechanisms. We propose a novel simulated regression based method for isoform frequency estimation from RNA-Seq reads. We present SimReg [1] – a novel regression based algorithm for transcriptome quantification. Simulated data experiments demonstrate superior frequency estimation accuracy of SimReg comparatively to that of the existing tools which tend to skew the estimated frequency toward super-transcripts.

Recent review of computational methods for transcriptome quantification from RNA-Seq data reports several problems with the current state of transcriptome quantifica-

tion, among them a significant variation in distributions of expressions level throughout transcriptome reconstruction and quantification tools [125]. Transcriptome quantification from RNA-Seq data highly depends on read depth. Due to the sparse read support at some loci, many tools fail to report all/some of the exons or exon-intron junctions.

Improving isoform frequency estimation error rate is critical for detecting similar transcripts or unraveling gene functions and transcription regulation mechanisms, especially in those cases when one isoform is a subset of another. Figure 2.10 shows a gene with sub-transcripts from human genome (hg19).

2.3.1 State-of the-art transcriptome quantification methods

From optimization point of view, the variety of approaches to transcriptome quantification is very wide. The most popular approach is maximizing likelihood using different variants of expectation-maximization (EM) [45, 126, 127], integer linear program (LP) based methods [42, 113], min-cost flow [128], and regression [129].

RNA-Seq by Expectation Maximization (RSEM) is an Expectation-Maximization (EM) algorithm that works on the isoform level. The initial version of RSEM only handled single-end reads, however, the latest version [126] has been extended to support paired-end reads, variable-length reads, and incorporates fragment length distribution and quality scores in its modeling. In addition to the maximum likelihood estimates of isoform expressions, RSEM also calculates 95% confidence intervals and posterior mean estimates. RSEM is the best algorithm presented so far, so we compare our tool SimReg to RSEM in Results and Discussion section.

The main limitation of statistically-sound EM approach is that it does not include uniformity of transcript coverage, i.e., it is not clear how to make sure that a solution with more uniform coverage of transcripts will be preferred to the one where coverage is volatile. LP and integer LP based methods overcome this limitation but cannot handle many isoforms simultaneously.

More recently, the authors of [127] proposed a quasi-multinomial model with a single

parameter to capture positional, sequence and mapping biases. Tomescu et al. [130] proposed a method based on network flows for a multiassembly problem arising from transcript identification and quantification with RNA-Seq. This approach is good at keeping overall uniformity coverage but is not suitable for likelihood maximization.

Regression based approaches are the most related to the proposed method. The most representative of these is IsoLasso approach [129]. IsoLasso mathematically model a gene partitions into segments (a segment is a consecutive exon region while a subexon is a non-spliced region).

IsoLasso approach also assumes reads being uniformly sampled from transcripts. The Poisson distribution [131] then used to approximate the binomial distribution for the number of reads falling into each segment or subexon. The following quadratic program [129] is well-known as a LASSO approach [112]:

$$\begin{aligned} &\text{minimize:} && \sum_{i=1}^{M} \left(\frac{r_i}{l_i} - \sum_{j=1}^{N} a_{ji} x_j \right)^2 \\ &\text{subject to:} && x_j \geq 0 \, , \, 1 \leq j \leq N \, , \, \sum_{j=1}^{N} x_j \leq \lambda \, , \, \forall t = 1...|T| \end{aligned} \tag{2.3}$$

and two more “completeness" constraints (namely that each segment or junction with mapped reads is covered by at least one isoform; and the sum of expression levels of all isoforms that contain this segment or junction should be strictly positive [129]) were added to this program in IsoLasso. The main over-simplification is an assumption that each segment receives from containing transcripts the number of reads proportional to its length. For example, it is not clear how to handle very short subexons and take in account position of a subexon in a transcript. Fragment length distribution also can discriminate one subexon from another. Especially difficult to accurately estimate portions of pair-end reads emitted from each subexon since in fact such reads are frequently emitted by multiple subexons collectively. Furthermore, mapping of the reads into transcripts is

frequently ambiguous which is consciously ignored in [129].

In this chapter we propose to apply a more accurate simulation of read emission. Our novel algorithm falls into the category of regression based methods: namely, SimReg is a Monte-Carlo based regression method.

In general, one of the main goals of differential expression (DE) analysis is to identify the differentially expressed genes between two or more conditions. Such genes are selected based on a combination of expression level threshold and expression score cut-off, which is usually based on p-values generated by statistical modeling. The expression level of each RNA unit is measured by the number of sequenced fragments that map to the transcript, which is expected to correlate directly with its abundance level [132].

The outcome of DE analysis is influenced by the way primary analysis (mapping, mapping parameters, counting) is conducted [132]. In addition, the overall library preparation protocol and quality is also an important factor of bias [133–135]. As described in the next chapters, DE analysis methods differ in how to deal with these pre-analysis phases. Furthermore, RNA Seq experiments tend to be underpowered (too few replicates) and we need methods to perform DE under these circumstances.

2.3.2 Simulated regression method for transcriptome quantification

The proposed method for estimating frequencies of transcripts is based on the novel approach for estimating expected read frequencies. First we describe the essence of our approach and contrast it with IsoLasso.

As discussed above, it is very difficult (if at all possible) to accurately estimate portions of pair-end reads emitted from each subexon. Instead, rather than distinguishing reads by their gene position, we partition reads into *classes* each consisting of reads consistent with each element of a particular subset of transcripts. In other words, two reads are assigned to the same class if they are consistent with exactly the same transcripts. Our second innovation is to use Monte-Carlo simulations instead of attempting to formally estimate contributions of each transcript to each read class. For any particular read class R,

the expected frequency is estimated based on the frequencies of contributing transcripts as well as portions of reads that fall into the class R. Finally, using the standard regression method, we estimate transcript frequencies by minimizing deviation between expected and observed read class frequencies.

The general description of the proposed simulated regression algorithm (SimReg) consists of four steps and is described below.

2.3.2.1 Splitting the transcripts and reads into independent connected components. We assume that alignment of a read to transcript is valid if the fragment length deviates from the mean by less than 4 standard deviations. Our simulations show that the Monte-Carlo estimates become accurate enough only when simulated coverage is sufficiently high, i.e., approaching 1000x. Such high coverage is time consuming since each simulated read needs to be aligned with each possible transcript. In order to reduce runtime, we split transcripts into small related subsets roughly corresponding to sets of overlapping genes. First, we build the matching graph $M = (\mathcal{T} \cup \mathcal{R}, E)$, where $\mathcal{T}$ and $\mathcal{R}$ are the sets of all transcripts and reads, respectively, and each edge $e = (r, T) \in E$ corresponds to a valid alignment of a read r to a transcript $T \in \mathcal{T}$. Transcript frequencies within each connected component of M do not depend on transcript frequencies within other connected components and can be estimated separately. A significant portion of connected components contains just a single transcript for which the next step is trivial. Finally, the observed reads are partitioned into read classes each consisting of reads mapped to the same transcripts (see Figure 2.11).

2.3.2.2 Estimating transcript frequencies within each connected component. As discussed above, in each connected component C we simulate reads with 1000x coverage for each transcript (see Figure 2.11). Thus for a transcript T with the length $|T|$ we generate $N_T = 1000 l_T$ reads, where $l_T = |T| - \mu + 1$ is the adjusted length of T. Similar to observed reads, we allow only alignments with fragment length less than 4σ away from μ. The

reads that belong to exactly the same transcripts are collapsed into a single read class. Let $\mathcal{R} = \{R\}$ be all read classes found in the connected component C and let R_T be the number of reads simulated from the transcript T that fall in the read class R. The first inner loop outputs the set of coefficients $D_{\mathcal{R},\mathcal{T}} = \{d_{R,T}\}$, where $d_{R,T}$ is the portion of reads generated from T belonging to R

$$D_{\mathcal{R},\mathcal{T}} = \left\{ \frac{|R_T|}{N_T} \right\}$$

Let $F'_{\mathcal{T}} = \{f'_T\}$ be the *crude* transcript frequency, i.e., the portions of reads emitted by transcripts in the connected component C. Then the expected read class frequency $E_{\mathcal{R}}$ can be estimated as

$$E_{\mathcal{R}} = D_{\mathcal{R},\mathcal{T}} \times F'_{\mathcal{T}} \tag{2.4}$$

Regression-based estimation of f''_ts minimizes squared deviation

$$(D_{\mathcal{R},\mathcal{T}} \times F'_{\mathcal{T}} - O_{\mathcal{R}})^2 = \sum_{R \in \mathcal{R}} (e_R - o_R)^2 \tag{2.5}$$

between expected read class frequencies e_R's and observed read class frequencies o_R's. Minimizing (2.5) is equivalent to the following quadratic program that can be solved with any constrained quadratic programming solver.

$$\begin{aligned} &\text{minimize:} && \sum_{R \in \mathcal{R}} \left(\sum_{T \in C} d_{R,T} f'_T - o_R \right)^2 \\ &\text{subject to:} && \sum_{T \in C} f'_T = 1 \text{ and } f'_T \geq 0 \, , \forall T \in C \end{aligned} \tag{2.6}$$

2.3.2.3 Update initial estimates of transcript frequencies. The obtained crude transcript frequency estimation $F'_{\mathcal{T}}$ can deviate from the true crude frequency since the minimization of deviation is done uniformly. Indeed, the deviation in frequency is minimized on the same scale for each read class while different read classes have different size,

as well as contribute to different subsets of transcripts. Instead of estimating unknown coefficients, we propose to directly obtain $F'_{\mathcal{T}}$ for which simulated read class frequencies $S_{\mathcal{R}} = \{s_R\}$ match the observed frequencies $O_{\mathcal{R}}$ accurately enough as follows.

Until the deviation between simulated and observed read class frequencies is small enough, we repeatedly

- simulate reads according to $F'_{\mathcal{R}}$,
- find deviation between simulated and observed reads, $\Delta_R = S_R - O_R$,
- obtain read frequencies $C_R = O_R - \Delta_R/2$ corrected half-way in the direction opposite to the deviation
- update estimated crude transcript frequencies $F'_{\mathcal{T}}$ based on corrected read class frequencies $\{C_R\}$

Finally, the transcript frequencies f_T's can be obtained from crude frequencies f'_T's as follows

$$f_T = \frac{f'_T/l_T}{\sum_{T' \in C} f'_{T'}/l_{T'}} \tag{2.7}$$

2.3.2.4 Combining transcript frequency estimates from all connected components. Finally, we combine together individual solutions for each connected component. Let f_T^{glob} and f_T^{loc} be the global frequency of the transcript T and local frequency of the transcript T in its connected component C. Then the global frequency can be computed as follows

$$f_T^{glob} = f_T^{loc} \times \frac{|R_C| / \sum_{T' \in C} f_{T'}^{loc} l_{T'}}{\sum_{C' \in \mathcal{C}} \frac{|R_{C'}|}{\sum_{T' \in C'} f_{T'}^{loc} l_{T'}}} \tag{2.8}$$

where $\mathcal{C}$ is the set of all connected components in the graph M, $|R_C|$ is the number of reads emitted by the transcripts contained in the connected component C.

2.3.3 Experimental results

2.3.3.1 Results on simulated data. We tested [1] SimReg on several test cases using simulated human RNA-Seq data. The RNA-Seq data was simulated from UCSC annotation (hg18 Build 36.1) using Grinder read simulator (version 0.5.0) [136], with a uniform 0.1% error rate. Experiments on synthetic RNA-seq datasets show that the proposed method improves transcriptome quantification accuracy compared to previous methods.

The following three test cases have been used to validate SimReg:

Case 1: consists of a single gene with 21 transcripts extracted from chromosome 1 (see Figure 2.12). From this gene we have simulated around 3000 (coverage $100\times$) paired-end reads of length 100bp and mean fragment length $\mu = 300$.

Case 2: we have randomly chosen 100 genes from which we have simulated reads using same parameters as in case 1.

Case 3: we have run our tool on the entire chromosome 1 which contains a total of 5509 transcripts (from 1990 genes) from where we have simulated 10M paired-end reads of length 100bp.

We have compared our results with RSEM, one of the best tool for transcriptome quantification. Frequency estimation accuracy was assessed using r^2 and the comparison results are presented in Table 1.The results show better correlation compared with RSEM especially because of those cases of sub-transcripts where RSEM skewed the estimated frequency toward super-transcripts.

2.3.3.2 Results on real data. For the real dataset we assayed sets of human genes using MicroArray Quality Control (MAQC) Human Brain Reference (HBR) sample [137] and NanoString nCounter amplification-free detection system [125].

For MAQC we have correlated [1] our results using the Taqman qRT-PCR values while for NanoString we have used the probe counts provided in [125]. Since Taqman qRT-PCR and NanoString counts only measure the expression levels of genes and probes, respectively, we only compare gene (probe) abundance estimations. The expression level

of a gene (probe) is obtained by summing up the frequencies of all transcripts in the gene (probe). For both datasets we have used the Ensembl Homo sapiens genome sequence indexes (GRCH37) provided by Illumina.

There are three 2×50bp paired-end datasets for Human Brain in SRA in MAQC dataset . The average insert size is about 200bp and the standard deviation about 30bp. In NanoString data set we have paired-end reads of length 75bp and similar characteristics as in MAQC (more details can be found in [125, 138])

In order to compute the 95% confidence interval (CI), we performed bootstrapping procedure by randomly choosing reads from the given set, and returning chosen samples back to the pool. As a result, our chosen subsample may contain several copies of the same reads, whereas some reads are never chosen. We repeat subsampling procedure 200 times. For each sample we compute MPE and r^2 for Cufflinks (v2.2.0), RSEM (v1.2.19), and SimReg and we count how many times our estimates are better than RSEM (since RSEM shows best performance compared to the other tools).

Table (2.5) Median Percent Error (MPE) and r^2 together with 95% CI for Transcriptome Quantification on MAQC and NanoString datasets [1]

Dataset: MAQC [137]				
Algorithm	MPE	[95% CI]	r^2	[95% CI]
SimReg	**77.2%**	76.0 - 79.7%	85.7%	80.2 - 89.0%
RSEM	78.0%	77.4 - 80.1%	86.4%	81.1 - 89.3%
Cufflinks	81.3%	79.5 - 85.2%	82.5%	78.9 - 85.1%
Dataset: NanoString [138]				
Algorithm	MPE	[95% CI]	r^2	[95% CI]
SimReg	**57.0%**	55.2 - 59.7%	82.0%	80.2 - 89.0%
RSEM	65.8%	61.3 - 68.2%	82.6%	78.7 - 85.4%
Cufflinks	67.9%	62.5 - 70.1%	79.9%	75.3 - 82.4%

The results presented in table 2.5 [1] show that SimReg has accuracy comparable to that of RSEM on the MAQC data, but outperforms RSEM in both MPE and r^2 on the Nanostring dataset. Mean Percentage Error of SimReg is less than that of RSEM in 90.5% of cases.

All experiments were conducted on a Dell PowerEdge R815 server with quad 2.5GHz 16-core AMD Opteron 6380 processors and 256Gb RAM running under Ubuntu 12.04 LTS.

SimReg is freely available at http://alan.cs.gsu.edu/NGS/?q=adrian/simreg

2.4 Software packages

Our software tools are available online and may be freely used for all non-commercial purposes.

2.4.1 TRIP

Novel transcript reconstruction from paired-end RNA-Seq reads.
http://grid.cs.gsu.edu/ serghei/?q=trip

2.4.2 DRUT

Discovery and reconstruction of unannotated transcripts in partially annotated genomes from high-throughput RNA-Seq data.
http://www.cs.gsu.edu/ serghei/?q=drut

2.4.3 SimReg

A simulated regression based algorithm for transcriptome quantification.
http://alan.cs.gsu.edu/NGS/?q=adrian/simreg

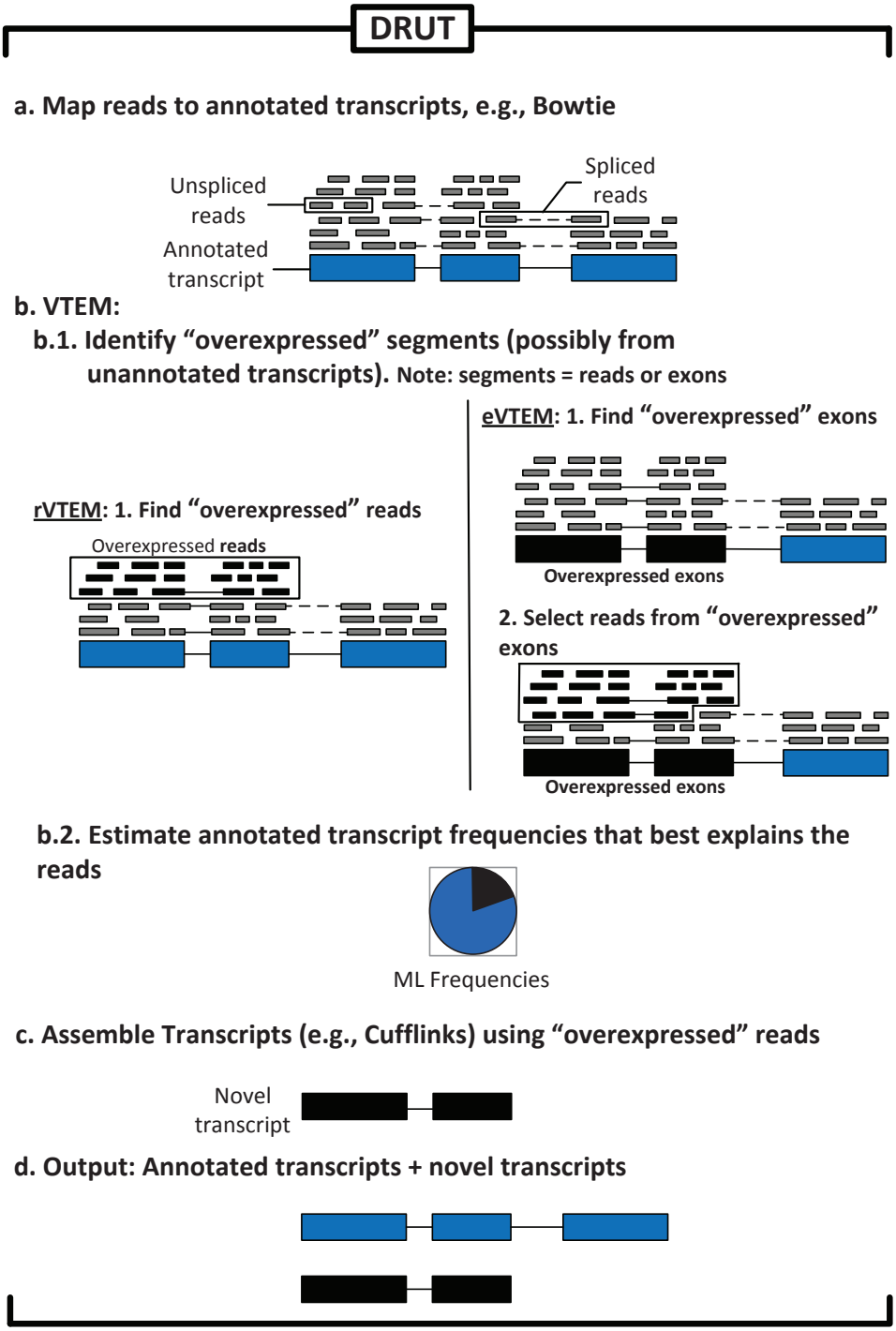
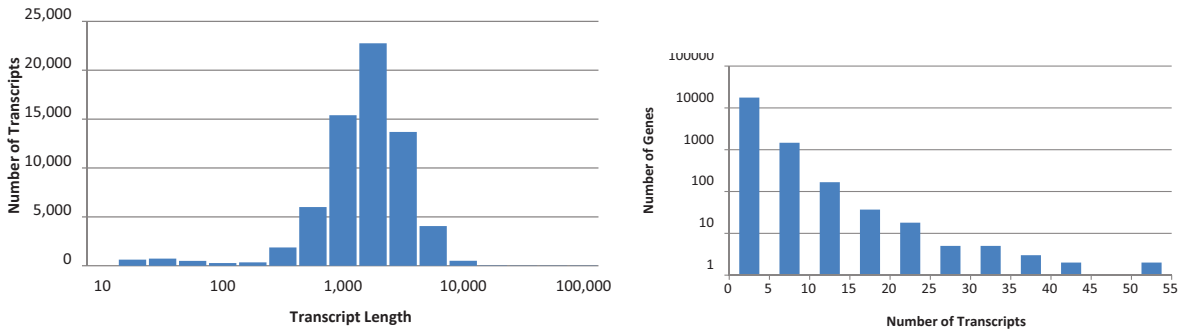


Figure (2.1) Flowchart for DRUT [2].



(a) (b)

Figure (2.2) Distribution of transcript lengths (a) and gene cluster sizes (b) in the UCSC dataset

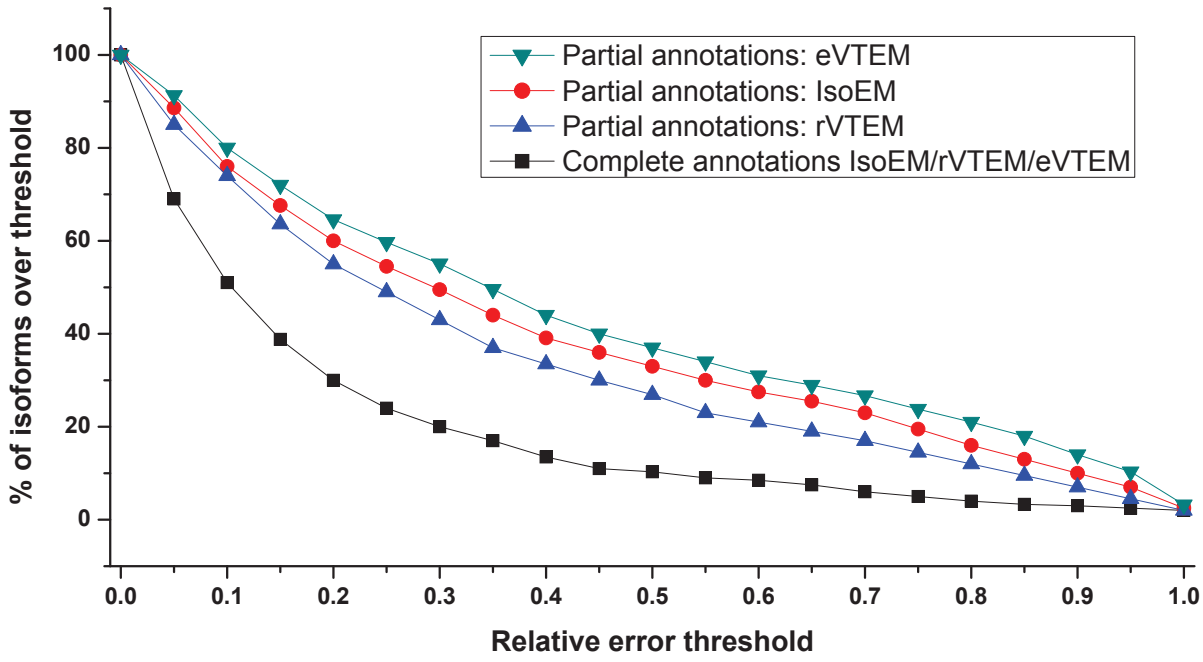
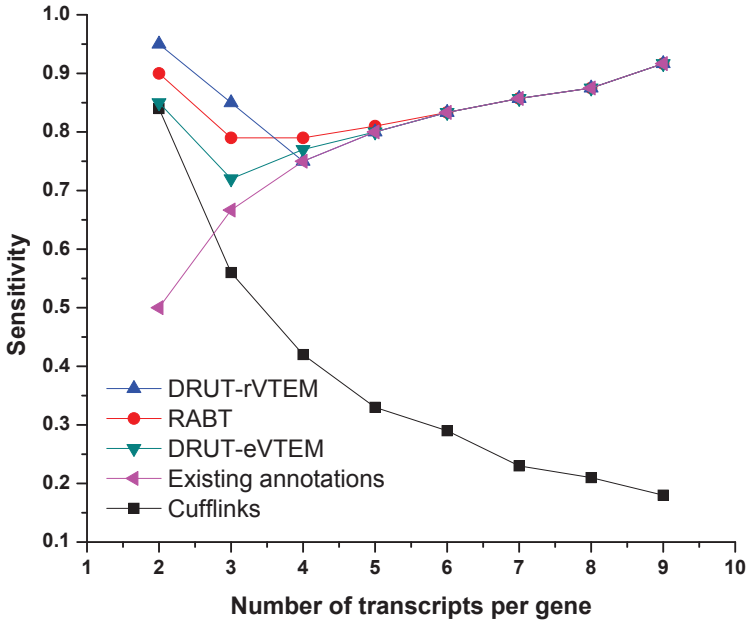
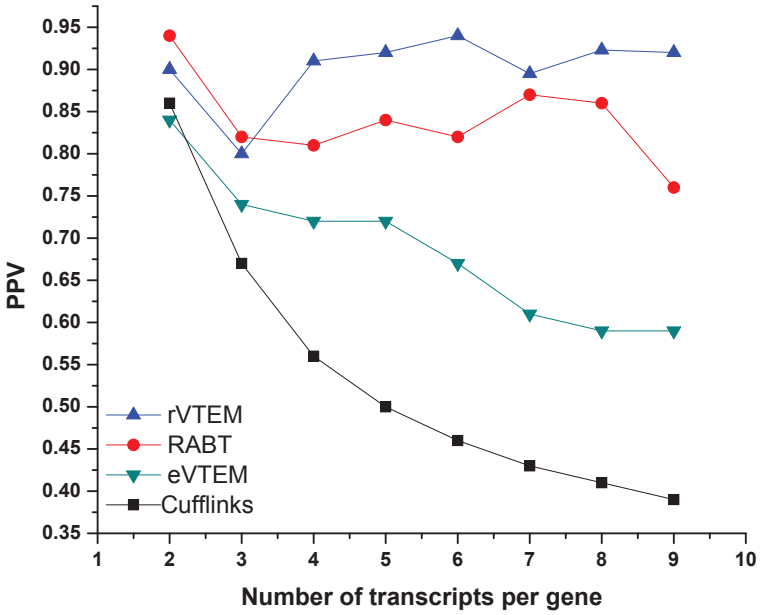


Figure (2.3) Error fraction at different thresholds for isoform expression levels inferred from 30 millions reads of length 25bp simulated assuming geometric isoform expression. Black line corresponds to IsoEM/VTEM with the complete panel, red line is IsoEM with the incomplete panel, blue line is rVTEM and the green line is eVTEM.



(a) Sensitivity



(b) PPV

Figure (2.4) Comparison between DRUT, RABT, Cufflinks for groups of genes with n transcripts (n=1,...,9) : (a) Sensitivity (b) Positive Predictive Value (PPV)

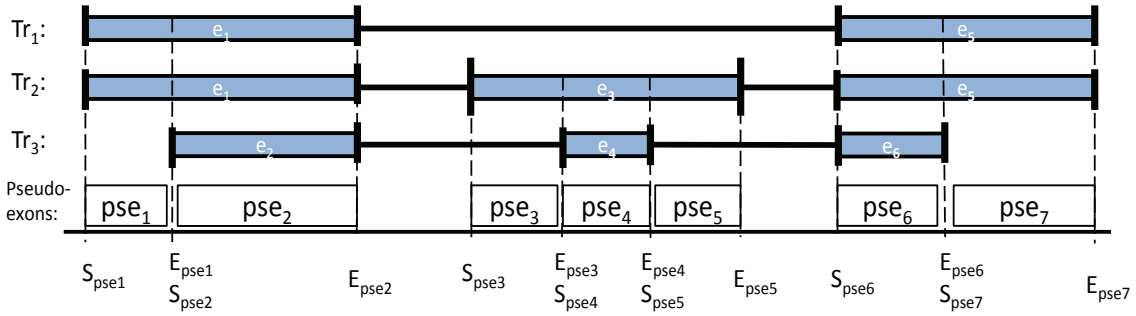


Figure (2.5) Pseudo-exons(white boxes) : regions of a gene between consecutive transcriptional or splicing events. An example of three transcripts $Tr_i, i = 1, 2, 3$ each sharing exons(blue boxes). S_{psej} and E_{psej} represent the starting and ending position of pseudo-exon j, respectively.

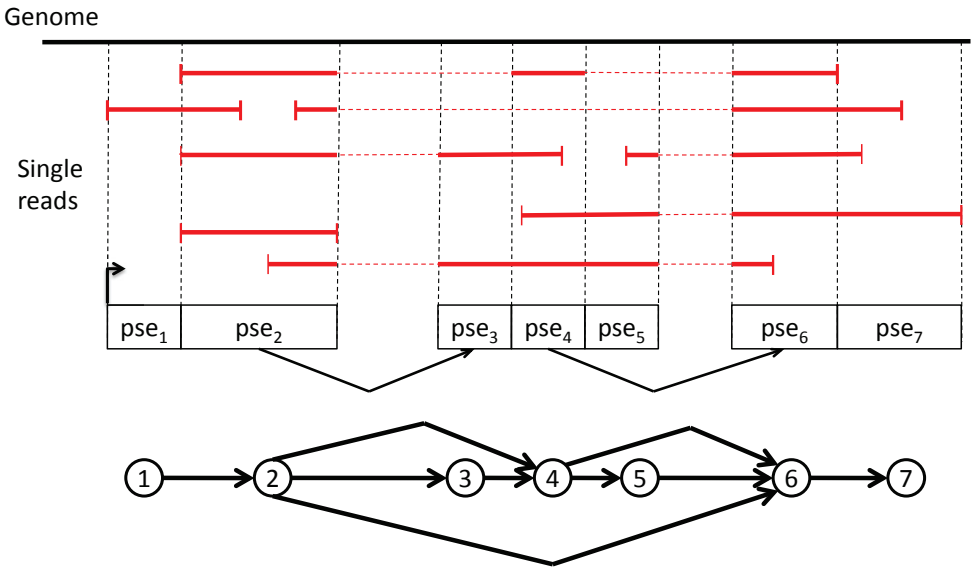


Figure (2.6) Splice graph. The red horizontal lines represent single reads. Reads interrupted by dashed lines are spliced reads. Each vertex of the splice graph corresponds to a pseudo-exon and each directed edge corresponds to a (splice) junction between two pseudo-exons.

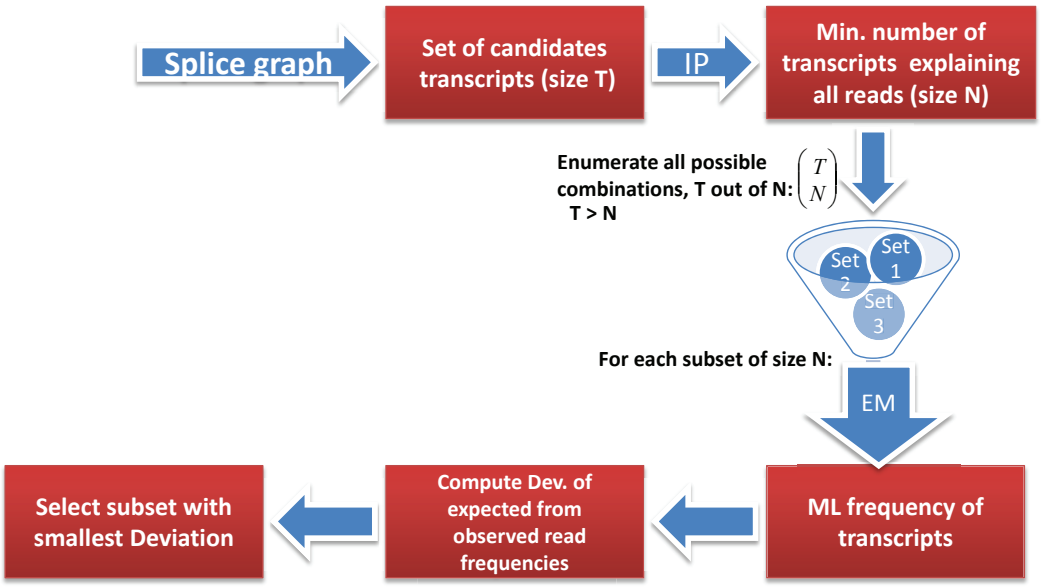
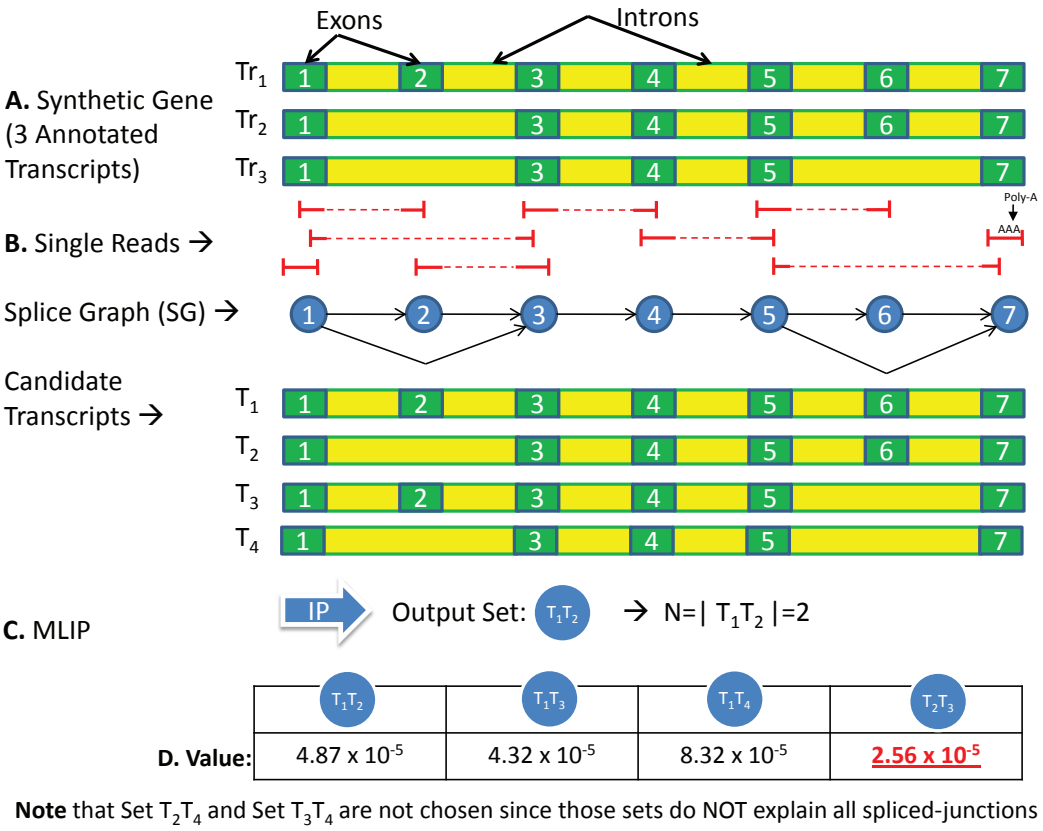
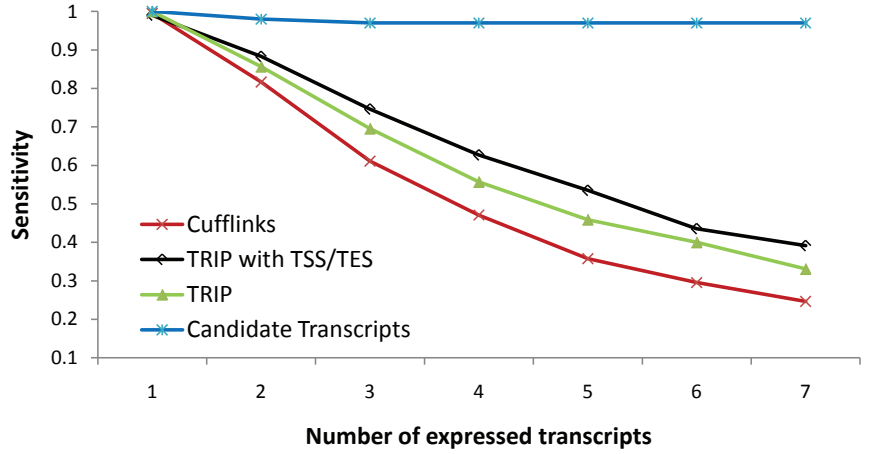


Figure (2.7) Flowchart for MLIP. Input : Splice graph. Output: subset of candidate transcripts with the smallest deviation between observed and expected read frequencies.

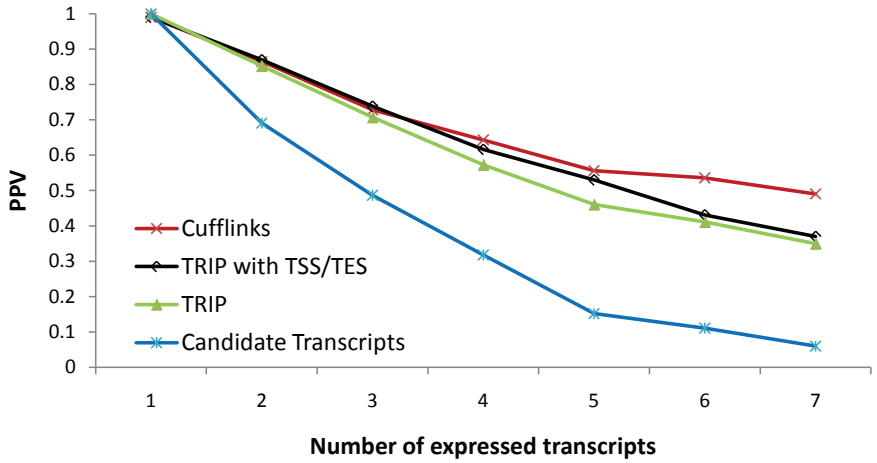


Note that Set T_2T_4 and Set T_3T_4 are not chosen since those sets do NOT explain all spliced-junctions

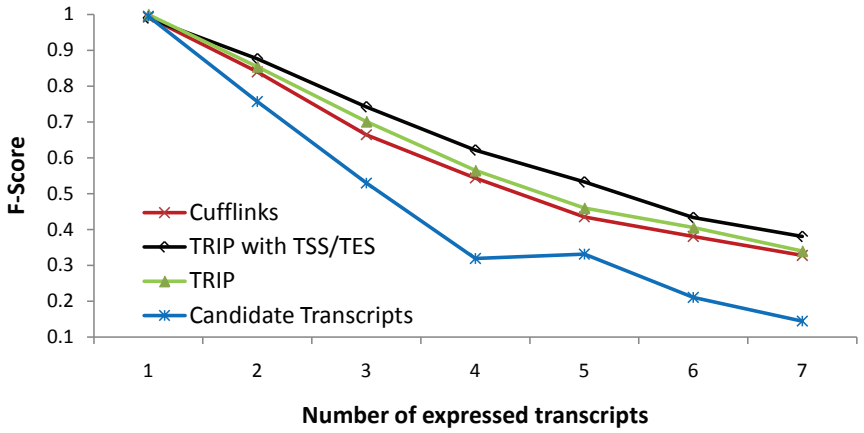
Figure (2.8) A. Synthetic gene with 3 transcripts and 7 different exons. B. Mapped reads are used to construct the splice graph from which we generate T possible candidate transcripts. C. MLIP. Run IP approach to obtain N minimum number of transcripts that explain all reads. We enumerate N feasible subsets of candidate transcripts.The subsets which doesn't cover all junctions and MLIP will be excluded from consideration. The subset with the smallest deviation between expected and observed read frequencies is selected by the MLIP algorithm.



(a)



(b)



(c)

Figure (2.9) Comparison between methods for groups of genes with n transcripts (n=1,...,7) on simulated dataset with mean fragment length 500, standard deviation 50 and read length of 100x2: (a) Sensitivity (b) Positive Predictive Value (PPV) and (c) F-Score.

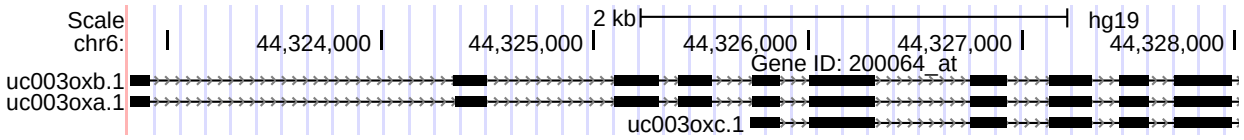


Figure (2.10) Screenshot from Genome browser [3]

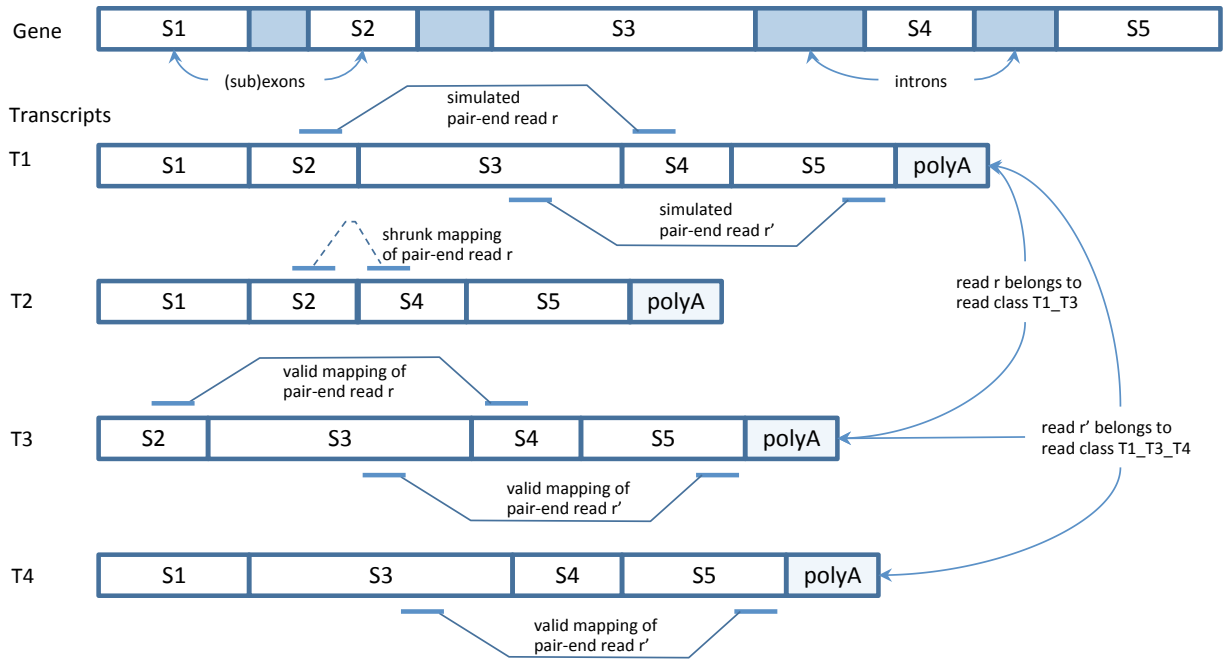


Figure (2.11) Paired reads r and r′ are simulated from the transcript T1. Each read is mapped to all other transcripts (T2, T3, T4). Mapping of the read r into the transcript T2 is not valid since the fragment length is 4 standard deviations away from the mean. Then each read is assigned to the corresponding read class – the read r is placed in the read class T1_T3 and the read r′ is placed in the read class T1_T3_T4.

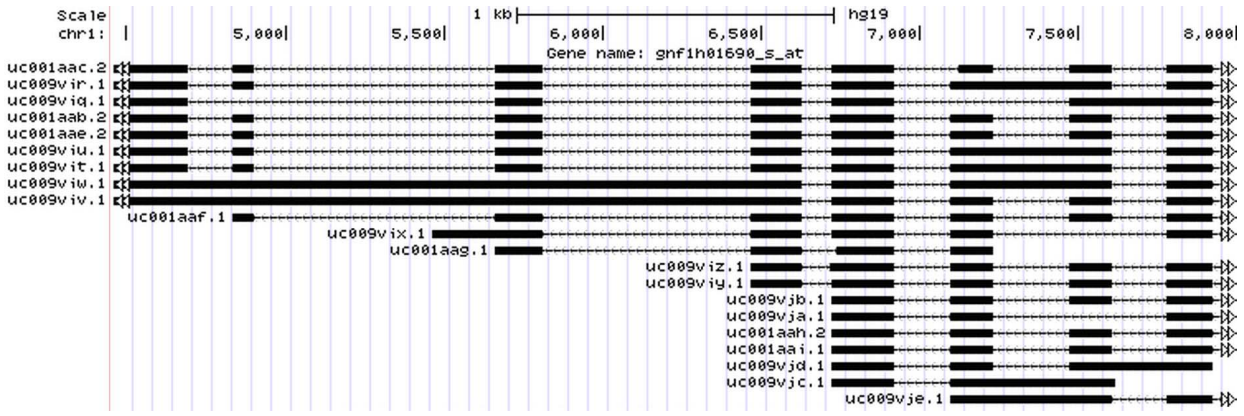


Figure (2.12) Screenshot from Genome browser [3] of a gene with 21 sub-transcripts

PART 3

ALIGNMENT OF DNA MASS-SPECTRAL PROFILES USING NETWORK FLOWS

3.1 Mass spectrometry technology

Mass spectrometry (MS) of DNA fragments generated by base-specific cleavage of PCR products emerges as a cost-effective and robust alternative to DNA sequencing.

MS is cheaper and less labor-intensive than most of the next-generation sequencing technologies [59,139], and also is not prone to the errors characteristic for these technologies. MS has been successfully applied to the reference-guided single nucleotide polymorphism (SNP) discovery [54,56,57], genotyping [53,58], viral transmission detection [59], identification of pathogens and disease susceptibility genes [60,61], DNA sequence analysis [62], analysis of DNA methylation [63], simultaneous detection of bacteria [64] and viruses [65,66].

MS technology is based on matrix-assisted laser desorption/ ionization time-of-flight (MALDI-TOF) analysis of complete base-specific cleavage reactions of a target RNA obtained from PCR fragments [53,54]. RNA transcripts generated from both strands of PCR fragment are cleaved by RNaseA at either U or C, thus querying for every of the 4 nucleotides (A, C, U and G) in separate reactions. Cleavage at any one nucleotide; e.g. U, generates a number of short fragments corresponding to the number of U's in the transcript. The mass and size of the fragments differ based on the number of A, C and G nucleotides residing between the U's that flank each short fragment. The fragments are resolved by MALDI-TOF-MS, resulting in mass spectral profiles, where each peak defines a specific mass measured in Daltons and has intensity that corresponds to the number of molecules of identical masses.

It should be noted that in MALDI-TOF-MS technology all molecules are equally singly charged, so the actual molecular weights could be obtained simply by subtract-

ing the mass of a single hydrogen from every mass from MS profile. Therefore, in the paper, we assume that MS profiles reflect molecular weights of the corresponding DNA molecules.

Unlike sequencing, MS is not readily applicable to reconstruction of the genetic composition of DNA/RNA populations. Algorithms for reconstruction of sequences from MS data were proposed [55]; but, owing to technological and computational limitations, none is widely used.

MS may serve as a rich source of information about the population structure and the genetic relations among populations without sequences reconstruction. One of the most important applications of sequences is to phylogenetic reconstructions. However, construction of phylogenetic trees requires knowledge of genetic distances among species rather than sequences, with sequences being merely used to estimate the distances. Comparison of MS profiles may also accurately approximate genetic distances. The problem of calculating the distance between two MS samples is known as spectral alignment problem [67, 68]. It is usually formulated as follows: match the masses from two MS profiles in such a way that some predefined objective function is maximized or minimized. We discuss the most common objective functions and methods for solving the spectral alignment problem in the next section.

Spectral alignment is crucial for the most applications of MS based on the matching of the sample and reference spectra, with the reference MS spectrum generated *in silico*. Spectral alignments are also used for MS data of proteins [140], but the protein technology and, therefore, the problem formulation and algorithm for its solution are completely different.

Here we propose a new formulation of the problem of aligning of the base-specific cleavage MS profiles (MS-Al) and present a method for its finding. The method is based on the reduction of the problem to the network flow problem. MS-Al allows *de novo* comparison of sampled populations and may be used for phylogenetic analysis and viral transmission detection. For conserved genomes (such as human genome) it allows

accurate estimation of actual genetic distance between DNA sequences.

3.2 Mass-spectral profiles alignment problem

MS profile $P = \{p_1, ..., p_n\}$ consists of n peaks, where each peak $p_i = (m(p_i), f(p_i))$ is represented by a mass $m(p_i)$ and intensity $f(p_i)$. Further without loss of generality we assume that $f(p_i)$ is an integer proportional to the number of occurrences of the mass $m(p_i)$ in the sample. In the simplest version, the spectral alignment problem could be formulated as follows [67]:

Problem 1.

Input: Two MS profiles $P^1 = \{p_1^1, ..., p_{n_1}^1\}$ and $P^2 = \{p_1^2, ..., p_{n_2}^2\}$

Find: Two subsets $P_*^1 \subseteq P^1$ and $P_*^2 \subseteq P^2$ of matched peaks and a bijection $\pi : P_*^1 \to P_*^2$ such that the following objective function is maximized:

$$score(P_*^1, P_*^2, \pi) - \sum_{p_i^1 \in P^1 \setminus P_*^1} pen(p_i^1) - \sum_{p_i^2 \in P^2 \setminus P_*^2} pen(p_i^2) \tag{3.1}$$

Here $score$ is a matching score function and pen is a mismatch penalty function. Usually it is assumed [67] that the function score is additive, which means that matches between different peaks are independent:

$$score(P_*^1, P_*^2, \pi) = \sum_{p_i^1 \in P_*^1} score(p_i^1, \pi(p_i^1)) \tag{3.2}$$

Most of known score functions are based on matches of peaks with close masses. In the simplest case we can put $pen \equiv 0$ and

$$score(p_i^1, p_j^2) = \begin{cases} 1, \ |m_i^1 - m_j^2| < \epsilon; \\ 0, \ \text{otherwise.} \end{cases} \tag{3.3}$$

Using these functions and a greedy algorithm for solving Problem 1, authors of [59, 139] accurately identified HCV transmission clusters.

In general, Problem 1 with a score function (3.2) could be efficiently solved using dynamic programming [67, 68]. However, it assumes that matches between different peaks are independent. In some cases this is not true, and taking into account dependencies between peak matches may significantly improve the quality of an alignment. One such case is MS based on a complete base-specific cleavage. Further we formulate spectral alignment problem in that case.

Let $\Sigma = \{\sigma_1, ..., \sigma_4\} = \{C, A, G, T\}$ be an alphabet, and let Σ^* be the set of strings over Σ. We assume that Σ^* contains the empty string o. Let $s = (s_1, ..., s_n) \in \Sigma^*$ and let $\Sigma_k = \Sigma \setminus \{\sigma_k\}$, $k = 1, ..., 4$. For each $\sigma_k \in \Sigma$ define $s(\sigma_k) = s(k)$ as

$$s(k) = \begin{cases} \{s\}, \ s_i \neq \sigma_k \text{ for every } i = 1, ..., n; \\ \{x \in \Sigma_k^* : s \in \{x\sigma_k y, z\sigma_k x, z\sigma_k x\sigma_k y\} \text{ for some } y, z \in \Sigma^*\}, \text{ otherwise.} \end{cases} \tag{3.4}$$

(see [55]). In other words, $s(k)$ is the set of all maximal substrings of s, which does not contain σ_k. For $s^1, s^2 \in \Sigma^*$ denote by $r_{s^1}(s^2)$ the number of substrings of s^1 equal to s^2.

Let $m(\sigma_k)$, $k = 1, ..., 4$ be the mass of the nucleotide σ_k and $m(s) = \sum_{i=1}^{n} m(s_i)$ be the mass of molecule represented by a sequence s.

Suppose that $S = \{s^1, ...s^m\}$, $s^j \in \Sigma^*$, is a sample tested using MS with base-specific cleavage. Let $S(k) = \bigcup_{j=1}^{m} s^j(k)$. MS profile P of S is partitioned into four subprofiles: $P = P(A) \cup P(G) \cup P(C) \cup P(T)$, where

$$P(\sigma_k) = \{p_i^{\sigma_k} = (m, f) : m \in \{m(s) : s \in S(k)\}, f = \sum_{\substack{s \in S(k): \\ m(s) = m}} \sum_{j=1}^{m} r_{s^j}(s)\} \tag{3.5}$$

Example 1. Let $S = \{s\}$ and $R = \{r\}$ be two samples each containing one sequence, s =AAGCTAGTTCA, r =AAGCTCGTTCA. Then

$$s(C) = \{\text{AAG,TAGTT,A}\}, s(A) = \{\text{GCT,GTTC}\},$$

$$s(G) = \{AA,CTA,TTCA\}, s(T) = \{AAGC,AG,CA\}$$

$$r(C) = \{AAG,T,GTT,A\}, r(A) = \{GCTCGTTC\},$$

$$r(G) = \{AA,CTC,TTCA\}, r(T) = \{AAGC,CG,CA\}$$

If $P_S = P_S(C) \cup P_S(A) \cup P_S(G) \cup P_S(T)$ and $Q_R = Q_R(C) \cup Q_R(A) \cup Q_R(G) \cup Q_R(T)$ are MS profiles of S and R, respectively, then they have the following form:

$P_S(C)$	$Q_R(C)$
p_1^C=(2m(A)+m(G),1)	q_1^C=(2m(A)+m(G),1)
p_2^C=(3m(T)+m(A)+m(G),1)	q_2^C=(m(T),1)
p_3^C=(m(A),1)	q_3^C=(2m(T)+m(G),1)
	q_4^C=(m(A),1)
$P_S(A)$	$Q_R(A)$
p_1^A=(m(G)+m(C)+m(T),1)	q_1^A=(3m(T)+3m(C)+2m(G),1)
p_2^A=(2m(T)+m(G)+m(C),1)	
$P_S(G)$	$Q_R(G)$
p_1^G=(2m(A),1)	q_1^G=(2m(A),1)
p_2^G=(m(C)+m(T)+m(A),1)	q_2^G=(2m(C)+m(T),1)
p_3^G=(2m(T)+m(C)+m(A),1)	q_3^G=(2m(T)+m(C)+m(A),1)
$P_S(T)$	$Q_R(T)$
p_1^T=(2m(A)+m(G)+m(C),1)	q_1^T=(2m(A)+m(G)+m(C),1)
p_2^T = (m(A)+m(G),1)	q_2^T = (m(C)+m(G),1)
p_3^T=(m(C)+m(A),1)	q_3^T=(m(C)+m(A),1)

6 of 11 peaks from P_S could be matched by the equal masses and the cleavage base with peaks from Q_R (p_1^C and q_1^C, p_3^C and q_4^C, p_1^G and q_1^G, p_3^G and q_3^G, p_1^T and q_1^T, p_3^T and q_3^T). However, it is easy to see that a single A-C SNP at position 6 between s and r causes the following relations between masses of remaining peaks:

$$m(p_2^C) = m(q_2^C) + m(q_3^C) + m(A) \tag{3.6}$$

$$m(p_1^A) + m(p_2^A) + m(C) = m(q_1^A) \tag{3.7}$$

$$m(p_2^G) - m(A) = m(q_2^G) - m(C) \tag{3.8}$$

$$m(p_2^T) - m(A) = m(q_2^T) - m(C) \tag{3.9}$$

If peaks and pairs of peaks are matched according to the relations (3.6)-(3.9) (p_2^C and (q_2^C, q_3^C), (p_1^A, p_2^A) and q_1^A, p_2^G and q_2^G, p_2^T and q_2^T), then all peaks from P_S and Q_R will be matched. Moreover, masses of single nucleotides and subprofiles involved in (3.6)-(3.9) allow to guess the corresponding SNP between s and r and in some cases the number of such type of matches allows to estimate the number of SNP's (in this example 1 SNP).

In general, the relations analogous to (3.6)-(3.9) have the following form:

$$m(p_i^{\sigma_{k_1}}) = m(q_{i_1}^{\sigma_{k_1}}) + m(q_{i_2}^{\sigma_{k_1}}) + m(\sigma_{k_2}) \tag{3.10}$$

$$m(p_{j_1}^{\sigma_{k_2}}) + m(p_{j_2}^{\sigma_{k_2}}) + m(\sigma_{k_1}) = m(q_j^{\sigma_{k_2}}) \tag{3.11}$$

$$m(p_{h_1}^{\sigma_{k_3}}) - m(\sigma_{k_2}) = m(q_{h_2}^{\sigma_{k_3}}) - m(\sigma_{k_1}) \tag{3.12}$$

$$m(p_{l_1}^{\sigma_{k_4}}) - m(\sigma_{k_2}) = m(q_{l_2}^{\sigma_{k_4}}) - m(\sigma_{k_1}) \tag{3.13}$$

Usually there are many possible alternative matches between peaks according to (3.10)-(3.13). The goal is to choose the optimal assignments such that the alignment score is maximized. Therefore the problem could be formulated as follows. Let $P_{(2)}$ be a set of all 2-element subsets of a set P. For $p \in P$ denote by $P_{(2)}(p)$ the set of all 2-subsets containing p. If P is a MS-profile, add to P an auxiliary empty peak $p_\epsilon = (0, \infty)$ with 0 mass

and unbounded intensity. We will call such profile an extended MS profile. We assume without loss of generality that all other peaks have intensity 1 (otherwise, if peak p_i has intensity $f(p_i) > 1$ replace it with $f(p_i)$ peaks of intensity 1). Further, extend an alphabet Σ by addition of an auxiliary empty symbol ϵ with $m(\epsilon) = 0$. Those additional objects are needed to include insertions, deletions and mutations in homopolymers (i.e. sequences of identical nucleotides) in the model.

Problem 2.

Input: Two extended MS profiles $P^1 = \{p_1^1, ..., p_{n_1}^1\} = P^1(C) \cup P^1(A) \cup P^1(G) \cup P^1(T) \cup \{p_\epsilon\}$ and $P^2 = \{p_1^2, ..., p_{n_2}^2\} = P^2(C) \cup P^2(A) \cup P^2(G) \cup P^2(T) \cup \{p_\epsilon\}$

Find: Two subsets $P_*^1 \subseteq P^1 \cup P_{(2)}^1$ and $P_*^2 \subseteq P^2 \cup P_{(2)}^2$ of matched peaks and pairs of peaks and a bijection $\pi : P_*^1 \to P_*^2$ such that the following conditions hold:

(i) $|P_*^j \cap (P_{(2)}(p_l^j) \cup \{p_l^j\})| \leq 1$ for every $p_l^j \in P^j \setminus \{p_\epsilon^j\}$, $j = 1, 2$ (every peak is matched at most once either as a singleton or as a member of a pair)

(ii) $\pi(\{p_i^1, p_j^1\}) \in P^2$ for every pair $\{p_i^1, p_j^1\} \in P_{(2)}^1$ (pair of peaks should be matched to a single peak);

(iii) there exists a bijection $\psi : P_*^1 \cap P_{(2)}^1 \to P_*^2 \cap P_{(2)}^2$ (matchings of pairs of peaks go in pairs)

and the objective function (3.1) is maximized. The objective function should be defined in such a way that

a) a pair of peaks is matched to a peak and vise versa only if (3.10) and (3.11) holds for them; the bijection ψ maps pairs which are conjugate by (3.10) and (3.11);

b) the number of matches involving pairs is as small as possible. Each such match potentially corresponds to an insertion, deletion or replacement and we are trying to align MS profiles with the smallest number of involved mismatches as possible - analogously to alignment of sequences using edit distance.

In the next section we show how to define such a function and present an algorithm for its calculation. This is a new approach, which, as Example 1 shows, is more accurate than the approaches based on the direct peak matching, and, moreover, in many cases allows to estimate the actual number and types of SNPs.

Note that (3.10)-(3.13) holds for a certain SNP, if it is isolated, which means that substrings between it and the closest SNPs contain all four nucleotides. For the conserved genomes this is a reasonable assumption: it was shown in [56] that the overwhelming majority of SNPs in human genome are isolated (for the data analyzed in [56] the average and minimal distance between two neighbor SNPs is 231bp and 14bp, respectively). Therefore for such genomes a solution of Problem 2 provides a reliable estimation for the number and types of SNPs. If two mutations happen in close proximity, then the relation between peaks caused by them is more complex than (3.10)-(3.13). Moreover, if sample contains more than one unknown sequence, it is usually impossible to assign peaks to each sequence. Therefore for a highly mutable genomes, such as viral genomes, solution of Problem 2 provides a distance, which specifies and generalizes the most commonly used distance with the score function (3.3), instead of direct estimation of the number of mismatches.

3.3 Network flow method for spectral alignment

For a directed graph (or network) N with a vertex set V, an arcs set A, pair of source and sink $s, t \in V$, arcs capacities cap and possibly arc costs $cost$ a network flow is a mapping $f : A \to \mathbb{R}_+$ such that $f(a) \leq cap(a)$ for every $a \in A$ (capacity constraints) and $\sum_{uv \in A} f(uv) - \sum_{vw \in A} f(vw) = 0$ for every $v \in V \setminus \{s, t\}$ (flow conservation constraints). The value of flow is $|f| = \sum_{sv \in A} f(sv)$. The classical network flow problem either searches for a flow of maximum value (Maximum Flow Problem) or for a flow with a given value of a minimum cost (Minimum-cost Flow Problem)

It is well-known that in discrete optimization many matching-related problems (such as Maximum Bipartite Matching Problem, Assignment problem, Minimum Cost Bipartite

Perfect Matching Problem, Linear Assignment Problem, etc.) could be solved using either network flows or shortest path - based algorithms. It suggests that a similar approach could be used for Problem 2. However, the formulation of Problem 2 is more complex that of the above-mentioned problems, so the reduction of Problem 2 to the network flow-based problem appeared to be rather complex. Below we present that reduction.

Let $P^1 = \{p_1^1, ..., p_{n_1}^1\} = P^1(C) \cup P^1(A) \cup P^1(G) \cup P^1(T) \cup \{p_\epsilon\}$ and $P^2 = \{p_1^2, ..., p_{n_2}^2\} = P^2(C) \cup P^2(A) \cup P^2(G) \cup P^2(T) \cup \{p_\epsilon\}$ be extended MS profiles. Let also $\delta \in \mathbb{R}_+$ be the mass precision, $g \in \mathbb{R}_+$ be the mismatch penalty and $p, q \in \mathbb{R}_+$ be the mutation (i.e. replacement, insertion, deletion) penalties corresponding to pairs of relations (3.10),(3.11) and (3.12),(3.13), respectively. Construct the network

$$N = (V, A, l, m, cost, cap) \tag{3.14}$$

where $l : V \to \Sigma^*$ is a vertices labels function, $m : V \to \mathbb{R}_+$ is vertices weights function, $cost : A \to \mathbb{R}_+$ and $cap : A \to \mathbb{R}_+$ are cost and capacity functions of arcs, respectively. Vertex set

$$V = \{s, t\} \cup V_1 \cup V_2 \cup V_{p_1} \cup V_{p_2} \cup V_{a_1} \cup V_{a_2} \cup V_{d_1} \cup V_{d_2}$$

and arc set A are constructed as follows:

1) s and t are the source and sink, respectively.

2) for each peak $p_i^j \in P^j(\sigma)$, $j = 1, 2$, $i = 1, ..., n_j$, $\sigma \in \Sigma$ the set V_j contains $f(p_j^i)$ vertices $v_j^i(1), ..., v_j^i(f(p_j^i))$. For each $v_j^i(k)$ $l(v_j^i(k)) = \sigma$, $m(v_j^i(k)) = m(p_i^j)$. For an empty peak $p_\epsilon \in P^j$, $j = 1, 2$, the set V_j contain the unique vertex v_ϵ^j with $l(v_\epsilon^j) = o$ and $m(v_\epsilon^j) = 0$.

3) For each $v \in V_1 \setminus \{v_\epsilon^1\}$ the set A contains an arc sv with $cost(sv) = 0$ and $cap(sv) = 1$. For each $v \in V_2 \setminus \{v_\epsilon^2\}$ A contains an arc vt with $cost(vt) = 0$ and $cap(vt) = 1$. There are also arcs sv_ϵ^1 and $v_\epsilon^2 t$ with $cost(sv_\epsilon^1) = cost(v_\epsilon^2 t) = 0$ and $cap(sv_\epsilon^1) = cap(v_\epsilon^2 t) = \infty$.

4) $uv \in A$ for each $u \in V_1$, $v \in V_2$ such that $|m(u) - m(v)| < \delta$ and $l(u) = l(v)$; $cost(uv) = 0$, $cap(uv) = 1$.

5) For every $u, v \in V_1$ and $w \in V_2$ such that

   a) $l(u) = l(v) = l(w)$,

   b) there exists $\sigma \in \Sigma$ such that $|m(u) + m(v) + m(\sigma) - m(w)| < \delta$,

   the vertex set V contains vertices $y \in V_{p_1}$ and $z \in V_{a_1}$ with $m(y) = m(z) = 0$, $l(y) = o$, $l(z) = l(u)\sigma$. The set A contains arcs uy, vy, yz, zw with $cost(uy) = cost(vy) = cost(yz) = cost(zw) = 0$, $cap(uy) = cap(vy) = cap(zw) = 1$, $cap(yz) = 2$. See Figure 1. The subgraph $N[u, v, w, y, z]$ induced by vertices u, v, w, y, z will be referred as left fork.

6) Analogously, for every $a \in V_1$ and $b, c \in V_2$ such that

   a) $l(a) = l(b) = l(c)$,

   b) there exists $\sigma \in \Sigma$ such that $|m(a) - m(b) - m(c) - m(\sigma)| < \delta$,

   the set V contains vertices $d \in V_{a_2}$ and $e \in V_{p_2}$ with $m(d) = m(e) = 0$, $l(e) = o$, $l(d) = \sigma l(b)$. The set A contains arcs ad, de, eb, ec with $cost(ad) = cost(de) = cost(eb) = cost(ec) = 0$, $cap(ad) = cap(eb) = cap(ec) = 1$, $cap(de) = 2$. See Figure 1. Further the subgraph $N[a, b, c, d, e]$ will be referred as right fork.

7) For vertices $u \in V_{a_1}$, $v \in V_{a_2}$ the set A contains an arc uv with $cost(uv) = p$ and $cap(uv) = 1$, if $l(u) = l(v)$. See Figure 1.

8) For every $u \in V_1$ and $v \in V_2$ such that

   a) $l(u) = l(v)$,

   b) there exists $\sigma_1, \sigma_2 \in \Sigma$ such that $|m(u) - m(\sigma_1) - m(v) + m(\sigma_2)| < \delta$,

the set V contains vertices $y \in V_{d_1}$ and $z \in V_{d_2}$ with $m(y) = m(z) = 0$, $l(y) = l(z) = \sigma_1\sigma_2$. The set A contains arcs uy, yz, zv with $cost(uy) = cost(yz) = cost(zv) = 0$, $cap(uy) = cap(zv) = 1$, $cap(yz) = 0$. See Figure 2.

9) for all distinct vertices $y, a \in V_{d_1}$, $z, b \in V_{d_2}$ such that $yz, ab \in A$, $cap(yz) = cap(ab) = 0$ and $l(y) = l(b)$, the set A contains arcs yb,az with $cost(yb) = cost(az) = \frac{g}{2}$, $cap(yb) = cap(az) = 1$. See Figure 2.

10) For every $v \in V_1$ there exists an arc vs with $cost(vs) = g$ and $cap(vs) = 1$.

Let $x : A \to \mathbb{N}, a \mapsto x_a$ is a flow in the network N. Problem 2 could be formulated as the following variant of the network flow problem:

$$\text{minimize} \sum_{a \in A} cost(a)x_a \tag{3.15}$$

subject to

$$\sum_{uv \in A} x_{uv} - \sum_{vw \in A} x_{vw} = 0, \quad v \in V \setminus \{s, t\}; \tag{3.16}$$

$$\sum_{sv \in A, v \neq v_e} x_{sv} = |V_1| - 1; \tag{3.17}$$

$$x_{uy} - x_{vy} = 0, \quad y \in V_{p_1}; \tag{3.18}$$

$$x_{eb} - x_{ec} = 0, \quad e \in V_{p_2}; \tag{3.19}$$

$$x_{uy} - x_{zv} = 0; \;\; yz \in A, cap(yz) = cost(uy) = cost(zv) = 0 \tag{3.20}$$

$$0 \leq x_a \leq cap(a), \;\; a \in A. \tag{3.21}$$

This formulation differs from the classical network flow problem formulation by additional constraints which require flow to be equal on some prescribed pairs of arcs.

Arcs from 4) provide the possibility of match between peaks with close masses with 0 penalty. Vertices and arcs from 5)-7) and constraints (3.18)-(3.19) allow to match peaks with pairs of peaks according to relations (3.10),(3.11). The capacities of arcs defined in 5)-7) are chosen in such a way that if flow goes through the left fork, then it should also go through the right fork indicating the same mutation, thus forcing a fulfillment of requirement (iii) of Problem 2. Moreover, if flow goes through some pair of forks, exactly one arc of cost p between those forks is involved, thus forcing penalty for mutation. Vertices and arcs from 8)-9) and constraints (3.20) play the same role for relations (3.12),(3.13). Constraint (3.17) for total size of the flow ensures that every peak is either matched or penalized for mismatch, which is encoded by arcs from 10). Moreover, arcs from 10) ensure that the problem (3.15)-(3.21) always has a feasible solution. (3.16) and (3.21) are standard flow conservation and capacity constraints.

If P^1 and P^2 are samples of single genomes with isolated SNPs, then the number of SNPs could be estimated as $|\{a \in A : x_a > 0, cost(a) = p\}|$.

3.4 Experimental results

The algorithm was tested on simulated data. For this, 80 pairs of sequences of lengths 40-60bp with 2-4 isolated SNPs were randomly generated. For each position one of possible symbols was chosen with equal probability to generate first sequence, and then random mutations were introduced on the prescribed positions to generate the second sequence. MS profiles of generated sequences were simulated using masses $m(A) = 329.21$ DA, $m(T) = 306.17$ DA, $m(G) = 345.21$ DA, $m(C) = 305.18$ DA. The ILP formulation (3.15)-(3.21) was solved using GNU Linear Programming Kit (GLPK) (http://www.gnu.org/software/glpk/) on a computer with two 2.67GHz processors and 12 GB RAM. Since ILP solution is usually time-consuming, the time limit 30 seconds per problem was established. For 90% (72 of 80) of test instances ILP was solved within the

time limit. For all that instances the numbers of SNPs were estimated correctly. Running times for ILP solution in that cases varies from 0.491 seconds in average with the standard deviation 0.968 seconds for 40bp sequences to 3.434 seconds with the standard deviation 5.824 seconds for 60bp sequences.

Thus the proposed approach enables an accurate comparison of MS profiles and provides a direct evaluation of genetic distances between DNA molecules without invoking sequences. It is potentially more accurate than the approaches based on the direct peak matching, and, moreover, in many cases allows to estimate the actual number and types of SNPs.

The proposed spectral alignment method is expected to be highly effective in evaluating genetic relatedness among viral samples and identifying transmission clusters in viral outbreaks. The reasons behind this presumption is based on the fact, that simple Hamming distance between samples could be calculated using a special case of our model with $p = q = \infty$. Hamming distance (which corresponds to the score function (3.3)) was shown to effectively separate transmission clusters [59, 139]. Thus, the developed model allows for generating a large spectrum of distances in addition to the special case and as such offers a more general framework for measuring genetic distances using MS profiles.

The ILP-based approach to solving the problem (3.15)-(3.21) is time-consuming. Therefore more computationally effective approaches may be required to handle larger samples. It is expected that direct applications of network flow-based methods, Lagrangian relaxations or other methods should dramatically increase performance of the algorithm. The generalizations of relations (3.10)-(3.13) in order to obtain a model allowing for estimation of the actual number of mutations in highly heterogeneous samples is an important direction for the future research.

PART 4

POOLING STRATEGIES FOR VIRAL MASSIVE SEQUENCING

4.1 Introduction

In this chapter we describe our novel framework for a cost-effective next-generation sequencing of heterogeneous viral populations, which combines barcoding and pooling recently proposed in [4]. This framework includes the following steps (Fig. 4.1):

(i) mixing samples in a specially designed set of pools in such a way that the identity of each sample is encoded in the composition of pools;

(ii) sequencing pools using barcoding;

(iii) deconvolution of samples; i.e., assignment of viral variants from the pools to individual samples.

This approach significantly decreases the number of PCR and NGS runs, thus reducing the cost of testing and hands-on time. As an additional benefit, pooling provides opportunity for PCR amplification of viral variants from each sample in different mixtures of samples generated in each pool, thus introducing variation in amplification biases and contributing to sequencing of a more representative set of viral variants from each sample. In difference to most pooling methods and algorithms for human samples, which aim at SNP calling (i.e. the identification of positions in the sequenced region which differ from the reference), this approach allows for finding the whole *viral quasispecies spectra*, i.e. viral sequences and their frequencies. However, application of the approach requires a careful designing of pools and significantly increases complexity of deconvolution of pools into individual samples, with the last task being especially demanding when applied to highly heterogeneous viral populations.

Sequence analysis of highly mutable RNA viruses is particularly difficult because of the complexity of their intra-host populations, the assessment of which can be dis-

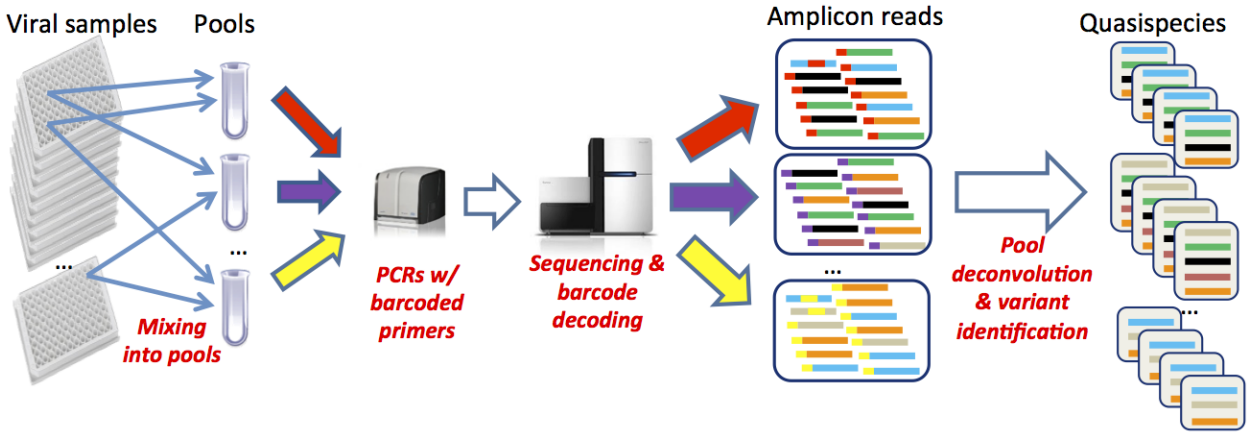


Figure (4.1) Combinatorial pooling strategy for viral samples sequencing [4].

torted by PCR or sampling biases, presenting additional challenges for application of the pool-based sequencing to these viruses. The complex nature of viral samples imposes restrictions on the pool design and deconvolution. It is essential to detect not only major but also minor viral intra-host variants from pools, since minor variants may have important clinical implications and in many cases may define outcomes of therapeutic treatment [33, 141, 142]. Mixing of a large number of specimens or specimens with significant differences in viral titers may contribute to under-representation of viral variants from some patients in pools, suggesting that size and composition of pools should be carefully designed.

Stochastic sampling from genetically diverse intra-host viral populations usually produces variability in compositions of sets of variants in different pools obtained from a single patient. Additionally, mixing specimens may differentially bias PCR amplification, contributing to mismatching between viral variants sampled from the same host in two pools with different specimen compositions. Thus, straightforward set-theoretical intersections among pools cannot be used for samples deconvolution, indicating that a more complex approach based on clustering techniques is needed. To increase the effectiveness of cluster-based deconvolution and minimize possible clustering errors, it is important to minimize mixing of genetically close samples as can be expected in epidemiologically

related samples and samples collected from a small geographic region.

4.2 Combinatorial pooling

The basic idea of the overlapping pools strategy for sequencing n samples is to generate m pools (i.e. mixtures of samples) with $m \ll n$ in such a way that every sample is uniquely identified by the pools to which it belongs [77]. Then, after sequencing of pools the obtained amplicon reads can be assigned to samples by the sequence of set-theoretic intersections and differences of pools. Below we two examples showing that a small number of pools can be used to uniquely identify larger number of samples.

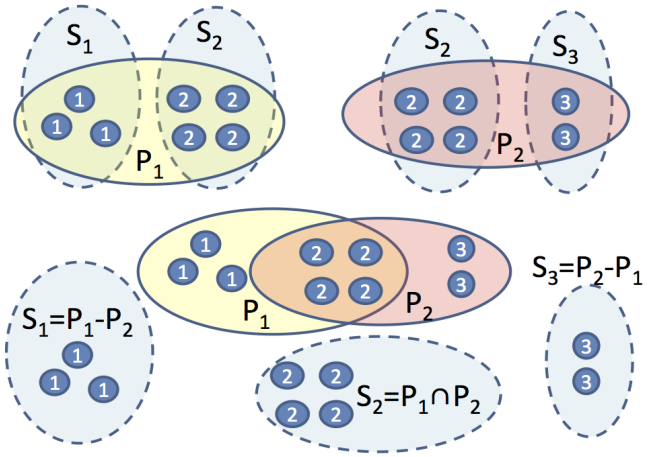


Figure (4.2) 2 pools for 3 samples: S_1 has 3, S_2 has 4 and S_3 has 2 variants. All 3 samples can be reconstructed from these 2 pools by pool intersection and subtraction [4].

Example 1. Consider 3 samples S_1, S_2, S_3 and 2 pools $P_1 = S_1 \cup S_2$, $P_2 = S_2 \cup S_3$ (see Figure 4.2). These pools satisfy the separation requirement, and, therefore, each sample can be recovered, e.g., $S_2 = P_1 \cap P_2$, $S_1 = P_1 \setminus P_2$, and $S_3 = P_2 \setminus P_1$. Thus, pooling sequencing of all 3 samples requires 2 sequencing runs.

Example 2. As a more complex example, consider 8 samples $S_1, ..., S_8$ and 4 pools $P_1, ..., P_4$ defined as follows: $P_1 = S_1 \cup S_2 \cup S_3 \cup S_4$, $P_2 = S_5 \cup S_6 \cup S_7 \cup S_8$, $P_3 = S_1 \cup S_2 \cup S_5 \cup S_6$, $P_4 = S_1 \cup S_3 \cup S_5 \cup S_7$. These pools satisfy the separation requirement, and therefore each

sample could be recovered by the sequence of intersections and differences of pools. For instance, $S_1 = P_1 \cap P_3 \cap P_4$, $S_2 = (P_1 \cap P_3) \setminus P_4$,...,$S_8 = (P_2 \setminus P_3) \setminus P_4$. Therefore, sequencing of all 8 samples may require 4 sequencing runs instead of 8.

The unique identification is possible if and only if for any two samples there is a pool *separating* them, i.e., containing exactly one of the samples. Indeed, if any two samples are separated by a pool, then the intersection of all pools containing sample S minus the union of all pools not containing S coincides with S. On the other hand, if two samples S_1 and S_2 are not separated by any pool, then it is impossible to distinguish them from each other by set-theoretical operations. This fact leads to an efficient pool design method described below.

Theorem [77, 143]. *If any subset of samples can form a single pool, then* n *samples can be reconstructed using* $m = \lceil \log(n) \rceil + 1$ *pools.*

Proof. Assume for simplicity that n is a power of 2, i.e. $n = 2^k$ (the proof is analogous for any n). Then apply induction by k. If $k = 1$, then $\mathcal{P} = \{\{S_1\}, \{S_2\}\}$ clearly is a valid pool design with $m = 2$. Suppose that $\mathcal{P}' = \{P'_1, ..., P'_{m'}\}$ is a valid pool design with $\mathcal{S}' = \{S'_1, ..., S'_{n'}\}$, $n' = n/2 = 2^{k-1}$, $m' = \log(n') + 1 = k$. Construct a family $\mathcal{P} = \{P_1, ..., P_{k+1}\}$ as follows:

$$P_i = \{S_{2i-1}, S_{2i} : S'_i \in P'_i\}, i = 1, ..., k; \tag{4.1}$$

$$P_{k+1} = \{S_1, S_3, ..., S_{n-1}\}. \tag{4.2}$$

The family $\mathcal{P}$ is a valid pool design. Indeed, it is clear that $\bigcup_{i=1}^{k+1} P_i = \mathcal{S}$. Since $\mathcal{P}'$ is a feasible pool design for $n' = n/2$, for every $i, j \in \{1, n/2\}$, $i \neq j$ there exists $l \in \{1, ..., k\}$ such that P'_l separates S'_i and S'_j. Thus by definition of the family $\mathcal{P}$, the set P_l separates the sets $\{S_{2i-1}, S_{2i}\}$ and $\{S_{2j-1}, S_{2j}\}$. Finally, the set P_{k+1} separates the samples S_{2i-1}, S_{2i} for every $i = 1, n/2$.■

4.3 Pool design optimization formulation

However, sequencing of heterogeneous RNA viral samples imposes the following additional restrictions on the pool composition: (i) the maximal number of samples that can be pooled without losing detection of many minority viral variants; and (ii) undesirability of mixing samples with drastically different viral titers or samples, which may be epidemiologically related. These restrictions make the pool design problem computationally much harder. Here, we formalize these restrictions and formulate the optimal pool design problem as an optimization problem on graphs.

Let $\mathcal{S} = \{S_1, ..., S_n\}$ be a set of samples and $X \subseteq \mathcal{S}$. The set X *separates* samples S_i and S_j, if X contains exactly one of the samples, i.e. $|X \cap \{S_i, S_j\}| = 1$.

Restrictions on the pool composition can be represented by a *sample compatibility* graph $G = G(S)$ with $V(G) = S$ and $S_iS_j \in E(G)$ if and only if the samples S_i and S_j could be mixed in the same pool. So, every feasible pool is a clique of the graph G. Let T be an upper bound for the pool size. The problem of the optimal pool design for sequencing of viral samples can be formulated as follows:

Viral Sample Pool Design (VSPD) Problem. Given a sample compatibility graph $G = (V, E)$ and a number $T > 0$, find the set of cliques $\mathcal{P} = \{P_1, \ldots, P_m\}$ of G such that m is minimized and

(1) $\cup_{i=1}^{m} P_i = V$;

(2) for every $u, v \in V(G)$ there is a clique $P_i \in \mathcal{P}$ separating u and v;

(3) $|P_i| \leq T$ for every $i = 1, ..., m$;

Unlike the case when any subset of samples can be a pool, the general VSPD problem is more challenging.

Theorem [143]. *Viral Sample Pool Design (VSPD) Problem is NP-hard, even for* $T = 3$.

Proof. We will reduce to VSPD with $T = 3$ the following special case of the yes/no 3-dimensional matching problem.

Problem A. *Given non-intersecting sets* X, Y, Z *, such that* $|X| = |Y| = |Z| = q$; $M \subseteq X \times Y \times Z$, *such that the following condition holds:*

(*) if $(a, b, w), (a, x, c), (y, b, c) \in M$, *then* $(a, b, c) \in M$.

Does M *contain a subset* $M' \subseteq M$ *such that* $|M'| = q$ *and every two elements of* M' *do not have common coordinates?*

The subset M' is called *3-dimensional matching*. It is known that the problem A is NP-complete [144]. Let X, Y, Z, M, $|X| = |Y| = |Z| = q$, be the input of the problem A. Construct a graph G as follows:

$$V(G) = X \cup Y \cup Z \cup A, \tag{4.3}$$

where $A = \{a_v : v \in X \cup Y \cup Z\}$;

$$E(G) = \bigcup_{(a,b,c)\in M} \{ab, bc, ac\} \cup \{va_v : v \in X \cup Y \cup Z\}. \tag{4.4}$$

We will show that the set M contains 3-dimensional matching if and only if the graph G contains a clique test collection $\mathcal{P} = \{P_1, ..., P_m\}$ of size $m = 4q$.

Let $\mathcal{P} = \{P_1, ..., P_m\}$ be a clique test collection of G, $m = 4q$. Let $R = X \cup Y \cup Z$. Let $\mathcal{P}' \subseteq \mathcal{P}$ be a set of cliques covering the vertices from the set A. For every $v \in R$ set $\mathcal{P}'$ contains either clique $\{a_v\}$ or clique $\{v, a_v\}$ or both of them. Let $R = R_1 \cup R_2$, where $R_1 = \{v \in R : \{a_v\}, \{v, a_v\} \in \mathcal{P}\}$, $R_2 = R \setminus R_1$.

Consider an arbitrary vertex $v \in R_2$. Set $\mathcal{P}'$ contains either clique $\{a_v\}$ or clique $\{v, a_v\}$. If $\{a_v\} \in \mathcal{P}'$, then set $\mathcal{P}'' = \mathcal{P} \setminus \mathcal{P}'$ contains at least one clique covering the vertex v. If $\{v, a_v\} \in \mathcal{P}'$, then $\mathcal{P}''$ contains at least one clique, which separates v and a_v. Thus, every $v \in W_2$ is covered by a clique from the set $\mathcal{P}''$.

Let $r_1 = |R_1|$. We have $|R_2| = 3q - r_1$, $|\mathcal{P}'| = 3q + r_1$, $|\mathcal{P}''| = 4q - |\mathcal{P}'| = q - r_1$. So, $3q - r_1$ vertices from the set R_2 are covered by $q - r_1$ cliques from set $\mathcal{P}''$. Since sizes of cliques from $\mathcal{P}''$ are at most 3 (by construction of the graph G), it is possible only if $r_1 = 0$, all cliques from $\mathcal{P}''$ contain exactly 3 vertices and do not pairwise intersect. The

condition (*) guarantees, that every triangle of the graph G belongs to set M, and so $\mathcal{P}''$ is 3-dimensional matching.

Conversely, if $M' \subseteq M$ is a 3-dimensional matching, then $\mathcal{P} = M' \cup \{\{v, a_v\} : v \in R\}$ is a clique test collection of graph G. Indeed, $\mathcal{P}$ covers all vertices of G; for every $v \in R$ a clique $\{v, a_v\}$ separates sets $\{v, a_v\}$ and $V(G) \setminus \{v, a_v\}$ and vertices v and a_v are separated by the clique from M' which contains v. ■

In practice, the condition 1) is not essential. Indeed, since every pair of vertices should be separated by some clique, at most one vertex $v \in V(G)$ is not covered by a clique from the set $\mathcal{P}$. Thus any family of cliques satisfying 2) and 3) can be transformed into a family satisfying 1) by adding just one additional clique $\{v\}$. Therefore, we will consider the problem without the condition 1).

4.3.1 Greedy heuristic for VSPD problem

We propose a heuristic algorithm for the VSPD problem. For the algorithmic purposes, in addition to the graph G, consider the graph H with $V(H) = V(G) = V$ and $ij \in E(H)$ if and only if the pair of vertices (i, j) is not separated yet. Initially, H is a complete graph.

Let $A \subseteq V$ be a set of vertices. A *cut* in the graph H is the pair $(A, V \setminus A)$, the *size of the cut* $c(A, V \setminus A)$ is the number of edges with one end in A and the other end in $V \setminus A$.

The basic scheme of the heuristics is described in Algorithm GPDA. At each iteration, Algorithm GPDA finds and adds to the solution a locally optimal pool, i.e. the pool which consists of compatible vertices and separates the maximal number of non-separated samples.

The crucial step of Algorithm GPDA finds locally optimal pool (step 4). It solves the following

Optimal Clique Cut Bi-Graph (OCBG) Problem. Given a graph $H = (V, E)$ and a constant T, find a clique in G with the set of vertices A, such that $|A| \leq T$ and the size of the cut $(A, V \setminus A)$ is maximized.

The OCBG problem is a previously unstudied discrete optimization problem. It is easy to see that this problem itself is NP-hard, and it is hard to approximate within a linear factor [4].

Theorem *Optimal Clique Cut Bi-Graph (OCBG) Problem is not approximable within* $O(n^{1-\varepsilon})$ *for any* $\varepsilon > 0$*, unless P=NP.*

Proof. Let an n-vertex graph G' be the input of CLIQUE Problem and $G = G' \cup O_n$. Without loss of generality we assume that G is connected. Consider the instance of LOP problem with G and $H = K_{2n}$ as an input. Then for the value f_{opt} of the optimal solution of LOP we have

$$f_{opt} = \max\{f(\omega) = \omega(2n - \omega) : \omega = |A|, a \text{ is a clique of } G\}.$$

Let us first show that the maximum clique size of G' is ω_{opt} if and only if $f_{opt} = \omega_{opt}(2n - \omega_{opt})$. Indeed, by construction $\omega = |A| \leq n$ for every clique A of G. The function $\omega(2n-\omega)$ increases monotonically on the segment $[1, n]$, and therefore f reaches its maximum on $\omega_{opt} = |A_{opt}|$, where A_{opt} is the maximal clique of graph G (and therefore of G').

Let $(A, V(G)\backslash A)$ be a solution of LOP, where A is a clique, $\omega = |A|$ and $f = \omega(2n-w)$. Suppose that

$$\frac{f_{opt}}{f} \leq \frac{1}{4}|V(G)|^{1-\varepsilon} = \frac{1}{4}(2n)^{1-\varepsilon}$$

for some $\varepsilon > 0$. Then

$$\frac{1}{2}\frac{\omega_{opt}}{\omega} \leq \frac{\omega_{opt}(2n - \omega_{opt}))}{\omega(2n - \omega)} = \frac{f_{opt}}{f} \leq \frac{1}{4}(2n)^{1-\varepsilon},$$

and therefore

$$\frac{\omega_{opt}}{\omega} \leq n^{1-\varepsilon}.$$

So, if LOP is approximable within $\frac{1}{4}|V(G)|^{1-\varepsilon}$ for some $\varepsilon > 0$, then CLIQUE is approximable within $|V(G')|^{1-\varepsilon}$. The latter is impossible, unless P=NP [145].■

In the Section 4.3.2 we will describe an efficient heuristic to solve the OCBG problem.

4.3.2 The tabu search heuristic for the OCBG problem

In this subsection we propose tabu search heuristic to solve the OCBG problem.

Let $M = |E(H)| + 1$. OCBGP can be formulated as the following quadratic programming problem:

$$\text{maximize } f(x) = \frac{1}{2}\sum_{ij\in E(H)}(1 - x_i x_j) - \frac{1}{8}M\sum_{ij\notin E(G)}(x_i + x_j)(x_1 + x_j + 2) \tag{4.5}$$

subject to

$$\frac{1}{2}\sum_{i\in V}(x_i + 1) \le T; \tag{4.6}$$

$$x_i \in \{-1, 1\},\ i \in V. \tag{4.7}$$

There is a 1-to-1 correspondence between solutions x of the problem (4.5)-(4.7) and the pairs of sets (A_x, B_x), $A_x \cup B_x = V$, where $A_x = \{i : x_i = 1\}$ and $B_x = \{i : x_i = -1\}$. Next, we will indicate the solution of (4.5)-(4.7) either by x or by (A_x, B_x).

The term $\frac{1}{2}\sum_{ij\in E(H)}(1 - x_i x_j)$ is equal to the size of the cut (A_x, B_x) in H. The term $\frac{1}{8}\sum_{ij\notin E(G)}(x_i + x_j)(x_1 + x_j + 2)$ is equal to the number of non-adjacent pairs of vertices in the induced subgraph $G[A_x]$; in particular, it is equal to 0 if A_x is a clique. So, $f(x) \ge 0$ if and only if A_x is a clique. Therefore, for any optimal solution of the problem (4.5)-(4.7), the set A_x is a clique. The constraint (4.6) ensures that $|A_x| \le T$.

Initially, we relax the constraint (4.6). Suppose that (A_x, B_x) is a feasible solution of (4.5), (4.7). For a vertex $v \in A_x$ consider the solution $(A_{x'}, B_{x'})$, where $A_{x'} = A_x \setminus \{v\}$, $B_{x'} = B_x \cup \{v\}$. Then for $\Delta_1 = f(A_{x'}, B_{x'}) - f(A_x, B_x)$ we have

$$\Delta_1 = deg^H_{A_x}(v) - deg^H_{B_x}(v) + M deg^{\overline{G}}_{A_x}, \tag{4.8}$$

where $deg^H_U(v)$ denotes the number of vertices from the set $U \subseteq V$ adjacent to a vertex $v \in V$ in a graph H, and $\overline{G}$ is a complement of a graph G. In particular, if v is non-adjacent to some vertex $u \in A_x$, then $\Delta_1 > 0$. Analogously, for $v \in B_x$, the solution $(A_{x'}, B_{x'})$ with $A_{x'} = A_x \cup \{v\}$, $B_{x'} = B_x \setminus \{v\}$ and $\Delta_2 = f(A_{x'}, B_{x'}) - f(A_x, B_x)$ we have

$$\Delta_2 = deg^H_{B_x}(v) - deg^H_{A_x}(v) - M deg^{\overline{G}}_{A_x}. \tag{4.9}$$

Thus, according to the relations (4.8) and (4.9), any initial solution (A, B) can be iteratively improved by moving vertices from one part of the partition to the other until a local optimum is reached, and the obtained solution cannot be further improved. According to (4.8), a local optimum A is a clique.

The major well-known general drawback of such local search strategies is that the value of the objective function in a local optimum may be far from the value of the globally optimal solution. Another problem, which is specific to our case, is that it is possible that the size of the locally optimal cut in H is 0. In that case the solution found at the stage 4) of Algorithm GOPDA will not decrease the set $E(H)$, and, therefore, Algorithm GOPDA will go into an infinite loop. To overcome these problems we use the variation of the tabu search strategy [146]. The basic idea is that if after the moving of a vertex v the algorithm arrives at a local optimum, then the following actions are taken: the value of the local optimum is compared to the current best solution, v is placed back and the moving of v is prohibited for the next k_t iterations of the algorithm. This idea is implemented in Algorithm OCBGP, which is described in more detail below.

Let $tabu^i = \{tabu^i_1, ..., tabu^i_n\}$ be the tabu state at the iteration i, i.e. a sequence of integers, where $tabu^i_j$ is a current number of iterations during which it is not allowed to move a vertex j. Let $optStates^i$ be the set of algorithm states, i.e. the set of pairs $((A, B), t_{A,B})$, where (A, B) is a local optimum found by the algorithm at some iteration $j < i$, and $t_{A,B}$ is

the tabu state at that iteration, i.e. $t_{A,B} = \text{tabu}^j$. Let (A^*, B^*) be the current record cut, i.e., the cut of the biggest size $c(A^*, B^*)$ found by the algorithm before the ith iteration. Let also moveList be the sequence of vertices moved by the algorithm from one part of the cut to another in the order of movement. This sequence is easier to implement as a stack, and it allows the algorithm to return to the previous solutions when the neighborhoods of solutions are completely explored. Let k_t denote the initial number of steps for which a move of a vertex is prohibited.

Below we detail the steps of the algorithm. Let $\text{tabu}^i = \{\text{tabu}_1^i, ..., \text{tabu}_n^i\}$ be the tabu state at the iteration i, i.e. a sequence of integers, where tabu_j^i is a current number of iterations during which it is not allowed to move a vertex j. Let optStates^i be a set of algorithm states, i.e. a set of pairs $((A, B), t_{A,B})$, where (A, B) is a local optimum found by the algorithm at certain iteration $j < i$, and $t_{A,B}$ is the tabu state at that iteration, i.e. $t_{A,B} = \text{tabu}^j$. Let (A^*, B^*) be the current record cut, i.e., the cut of the biggest size $c(A^*, B^*)$ found by the algorithm before the ith iteration. Let also moveList be the sequence of vertices moved by the algorithm from one part of the cut to another in the order of movement. This sequence is easier to implement as a stack, and it allows the algorithm to return to the previous solutions when neighborhoods of solutions are completely explored. Let k_t denote the initial number of steps for which a move of a vertex is prohibited.

At each iteration, Algorithm OCBGP tries to improve the current solution by moving one vertex from one part of the current cut to another part (steps 4-7). After the calculations for the cut improvement, the current tabu state is updated (step 8) and, if the current solution can be improved, the algorithm does it and proceeds to the next iteration (steps 9-12). If the current solution (A^i, B^i) cannot be improved, then it is a local optimum (stages 13-32). Then according to (4.8) A^i is a clique. In that case the algorithm compares the obtained locally optimal solution with the record and updates it, if necessary (steps 14-16). Then the algorithm returns to the previous solution (step 18) and forbids for the next k_t steps of moving the vertex, which leads to the previous local optimum (step 19). If the current algorithm state has not occurred previously, then the algorithm adds it to the

set optStates and proceeds to the next iteration (steps 21-24). Otherwise, the algorithm reduces the current record (A^*, B^*) to the solution where $|A^*| \leq T$, and stops (steps 25-30).

The default value of k_t is 1. It is still possible that for the solution (A^*, B^*) found by Algorithm 2 we have $c(A^*, B^*) = 0$. If it happens, we increase k_t by one and repeat Algorithm 2.

4.4 Deconvolution of viral samples from pools

According to Example 1 and 2 deconvolution requires computing of intersections and differences of pools. In Section 4.4.1 we formally define generalized intersections and differences of pools and show how to use them for pool deconvolution. The challenges of implementation of generalized intersections and differences are addressed in Section 4.4.2

4.4.1 Deconvolution using generalized intersections and differences of pools

Let $\mathcal{P}$ be the set of pools designed using a solution of the VSPD problem found by Algorithm GOPDA and sequenced using NGS. As discussed above, the obtained reads theoretically can be assigned to samples by the sequence of set-theoretic intersections and differences of pools (see Examples 1,2). However, owing to the high heterogeneity of viral populations and sampling bias, individual viral variants and even subpopulations of viral variants sequenced from a certain sample mixed into different pools may be different in each pool(see an example on Figure 4.3). It hampers the usage of straightforward set-theoretic intersections and differences, and, therefore, “generalized" intersections and differences should be used instead.

For a pool P_i, let $\mathcal{S}(P_i)$ be a set of samples mixed in it. In particular, for simplicity of notations, we can assume that each individual sample R_j is a special type of pool with $|S(R_j)| = 1$.

We define the *generalized intersection* of pools P_1 and P_2 as the pool $P_1 \overline{\cap} P_2$ with $\mathcal{S}(P_1 \overline{\cap} P_2) = \mathcal{S}(P_1) \cap \mathcal{S}(P_2)$, consisting of sequences from $P_1 \cup P_2$ that belong to the samples

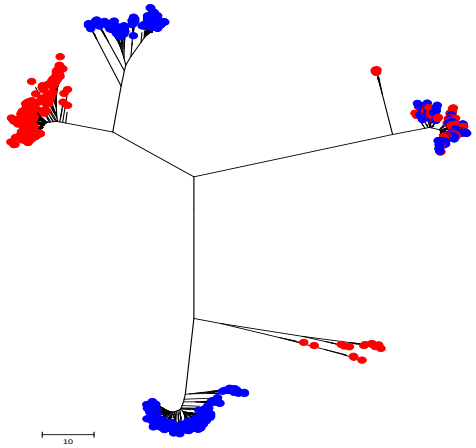


Figure (4.3) Phylogenetic tree representing a union of two pools: P_1 consisting of samples S_1, S_2, S_3 (shown in red) and P_3 consisting of samples S_1, S_4, S_5 (shown in blue) (see Section "Results, Experimental pools"). The intersection of two pools consists of the sample S_1 (upper right cluster in the tree); however, sequences sampled from S_1 in pools P_1 and P_2 are different [4].

from $\mathcal{S}(P_1) \cap \mathcal{S}(P_2)$. The *generalized difference* $P_1 \overline{\setminus} P_2$ then can be defined as follows: $P_1 \overline{\setminus} P_2$ is the pool with $\mathcal{S}(P_1 \overline{\setminus} P_2) = \mathcal{S}(P_1) \setminus \mathcal{S}(P_2)$ that contains sequences of the set $P_1 \setminus (P_1 \overline{\cap} P_2)$.

Individual samples can be inferred from pools by a sequence of generalized intersections and differences using Algorithm IS. By definition, generalized differences may be reduced to generalized intersections. For generalized intersections calculation we propose the scheme described in Algorithm GI, which is based on clustering techniques.

Theoretically, Algorithm GI may be used with the parameter $W = 1$. However, viral populations of highly mutable viruses, such as HCV and HIV, may differ greatly in heterogeneity. In extreme cases, intra- and inter host heterogeneity of certain samples may be comparable. If such samples belong to the same pool, it can lead to the effect when with $W = 1$ highly heterogeneous samples may be partitioned into multiple clusters, while samples with lower heterogeneity will be joined into one cluster. Such clustering will lead to the incorrect detection of generalized intersections and consecutive loss of samples, which were not separated from other samples. To avoid this effect, higher values of W should be used. In our experiments we used the default value $W = 2$. If certain samples are not found by Algorithm IS (i.e. the corresponding data sets are empty), we

increase the value of W by one and repeat Algorithm IS.

4.4.2 Maximum likelihood *k*-clustering

In this section we formulate the viral sample clustering problem and describe our solution, which is based on the probabilistic k-means approach (see [147]).

Sample Clustering Problem. Given a set R of NGS reads drawn from a mix of k' RNA viral samples, partition R into $k = Wk'$ subsets consisting of reads from a single sample.

The presence of numerous sequence variants in each viral sample, extreme heterogeneity of viral populations and a very large number of reads make Sample Clustering Problem challenging. Although a commonly used clustering objective is to minimize intra-cluster distances or distance to cluster centers (e.g., the k-means algorithm), we propose to use a statistically sound objective of maximizing likelihood. Our likelihood model estimates the probability of a certain read being emitted by a cluster consensus (or *centroid*).

Our algorithm receives a multiple sequence alignment of a given set of reads R as an input. We represent R as a matrix with columns corresponding to the consensus positions and rows corresponding to aligned reads. Our model assumes that each read in a cluster is emitted by a particular genotype (centroid). The proposed clustering (a) finds k genotypes $g_1, \ldots, g_k$ that most likely emit the observed set of reads, (b) estimates probability $p_{i,r}$ that read r is emitted by a genotype g_i, and (c) assigns a read r to a cluster which genotype most likely emits r.

Formally, given a set of reads C, a *genotype* $g(C)$ of C is a matrix with each column corresponding to a consensus position and 5 rows each corresponding to one of the alleles $\{a, c, t, g, d\}$. Each entry $f_m(e), e \in \{a, c, t, g, d\}$ is the frequency of allele e in m-th position among all variants in C, $\sum_{e \in \{a,c,t,g,d\}} f_m(e) = 1$. In particular, every read can be considered as a genotype with a single 1 and 4 zeroes in each column. Given a set of reads

R, a k-*genotype* is a set $G^* = \{g_1, \ldots, g_k\}$ of k distinct genotypes that most likely emitted R:

$$G^* = \arg\max_{|G|=k} \Pr(R|G),$$

where $\Pr(R|G)$ is the probability to observe R given a set of genotypes G, which is calculated as a product of probabilities to observe each read from R. The probability to observe read r equals to $\Pr(r) = \sum_{i=1}^{k} f_i \Pr(r|g = g_i)$, where

$$\Pr(r|g = g_i) = \prod_{m=1}^{L} f_{i,m}(r_m), \tag{4.10}$$

and $f_{i,m}(r_m)$ is the frequency of r_m, the m-th character of read r, in the m-th position of genotype g_i. Then the log-likelihood of the set of allele frequencies $\mathcal{F} = \{f_{i,m}(e)|i = 1, \ldots, k; m = 1, \ldots, L; e \in \{a, c, t, g, d\}\}$ equals to

$$\ell(\mathcal{F}) = \sum_{r \in R} o_r \log \Pr(r),$$

where o_r is the observed read frequency.

We iteratively estimate the missing data $p_{i,r}$, i.e., the number of times the read r originated from the genotype g_i, and solve the easier optimization problem of maximizing the log-likelihood of the hidden model

$$\ell_{hid}(\mathcal{F}) = \sum_{r \in R} \sum_{i=1}^{k} p_{i,r} \log(f_i \Pr(r|g = g_i)).$$

Our clustering method is described in Algorithm kGEM. The initial set of genotypes $G^{(0)}$ is selected as follows: starting from the most frequent read, we iteratively select the read maximizing the minimum Hamming distance to the previously selected reads and add to $G^{(0)}$ the corresponding genotype.

4.5 Performance of pooling methods on simulated data

4.5.1 Performance of the viral sample pool design algorithm

The pool generation algorithm was evaluated using 3 sets of simulated data.

1) Complete graphs.

Pools were generated for complete graphs with $n = 4, ..., 1024$ vertices without the pools size threshold. For every test instance, exactly $m = \lceil \log(n) \rceil + 1$ pools were constructed, coinciding with the theoretically justified estimation [77, 143]. Hence, the VSPD algorithm produces optimal solutions for complete graphs.

2) Random graphs, where each vertex v receives a random titer $w_v \in \{1, L\}$, and two vertices u and v are adjacent if and only if $|w_u - w_v| \leq R$. This family of test instances represents *titer compatibility model*, i.e. it simulates the case in which two samples could be mixed into one pool only if their viral titers are not sufficiently different.

25000 test instances were generated with $n = 10, ..., 1000$, parameters $L = 20$, $R = 4$ and with the pools sizes thresholds $T = n$ (i.e., without the threshold), $T = 55$, $T = 35$ and $T = 25$. For each n the mean size of the set of pools constructed by the VSPD algorithm and the mean sequencing reduction coefficient (i.e., the number of pools divided by the number of samples) were calculated. The results are shown on Figure 4.4, (a).

For $n = 1000$ sets of pools generated by the VSPD algorithm, more than 21-fold reduction in the number of sequencing runs is achieved for $T = n$, 15-fold reduction for $T = 55$, 11-fold reduction for $T = 35$, 9-fold reduction for $T = 25$ and 6-fold reduction for $T = 15$. The reduction coefficient in all these cases is a decreasing function of n, which suggests a higher reduction for the larger n.

3) Random graphs, where each edge is chosen with probability $p = 0.25, 0.5, 0.75$ and 1 and sizes of pools are bounded by $T = 35$.

20000 test instances with $n = 10, ..., 1000$ were generated and processed by the VSPD algorithm. As above, for each n the mean sequencing reduction coefficient was calculated. The results are shown in Figure 4.4, (b). In this case, as well as in the previous

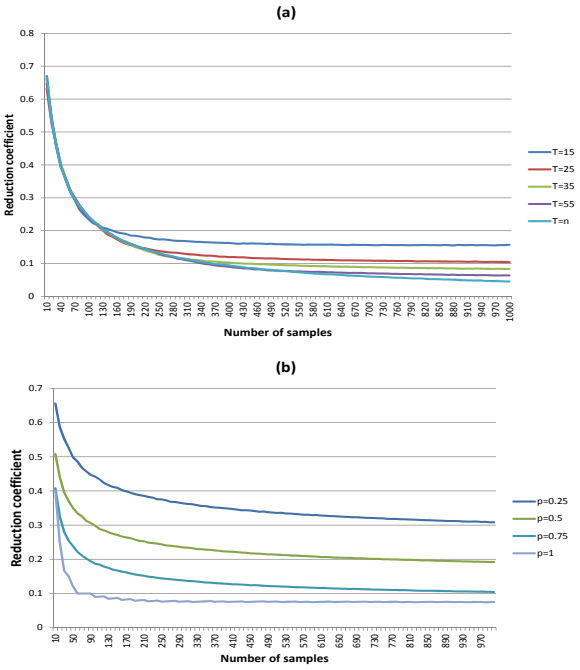


Figure (4.4) Sequencing reduction coefficient for the pools generated by the VSPD algorithm for (a) random titer compatibility model graphs; (b) random graphs [4].

one, pooling provides a great reduction in the number of sequencing runs, although it is generally lower than for the test instances 2) (from more than 13-fold reduction for $p = 1$ (complete graph) to more than 3-fold reduction for $p = 0.25$). The reduction coefficient is also a decreasing function of n.

4.5.2 Performance of the pool deconvolution algorithm

450 test instances with $n = 10, ..., 150$ samples and with the pool sizes thresholds $T = 15, 25, 35$ were generated. Simulated pools were constructed using 155 HCV HVR1 samples previously sequenced in Molecular Epidemiology and Bioinformatics Laboratory, Division of Viral Hepatitis, Centers for Disease Control and Prevention using 454 GS Junior System (454 Life Sciences, Branford, CT) [33, 139, 148, 149]. Reads from each sample were cleaned from sequencing errors using NGS error correction algorithms KEC

and ET [150]. Test instances were generated as follows:

1) n samples were chosen randomly;

2) a random samples compatibility graph on n vertices was generated based on the titer compatibility model and pools were designed using the VSPD algorithm (Algorithm GOPDA);

3) pools were created by taking D randomly selected reads from the samples composing each pool (in order to simulate a sampling bias). The number of reads per pool D was set as $D = 10000$, which approximately corresponds to the sequencing settings, under which the data used for simulation were obtained (454 Junior System with 8-10 MIDs per sequencing run).

For all test instances all samples were inferred, i.e. all n data sets produced by Algorithm IS were non-empty. It is possible that some reads are not classified into samples and therefore are lost by the algorithm. However, the number of such reads was extremely low (Figure 4.5, (a)): in average 99.996% of reads for $T = 15$, 99.993% for $T = 25$ and 99.984% for $T = 35$ were classified into samples.

An overwhelming majority of reads were classified correctly (Figure 4.5, (b)): in average, 99.998% of reads for $T = 15$, 99.982% for $T = 25$ and 99.959% for $T = 35$ were assigned to the right samples. It should be noted that the percentages of classified and correctly classified reads in general do not depend on the number of samples, if this number is large enough.

We call an incorrect assignment of reads to the samples *in silico contamination*. The average percentage of samples without in silico contamination ranges from 100% to 98.13% for $T = 15$, from 100% to 96.13% for $T = 25$ and from 100% to 93.8% for $T = 35$ (Figure 4.6, (a)); the percentage of in silico contaminated samples increases with the total number of samples. In silico contaminants constitute a small minority within contaminated samples: in average 0.163% of all reads for $T = 15$, 0.545% for $T = 25$ and 0.892% for $T = 35$ (Figure 4.6, (b)).

Root Mean Square Error of deconvoluted haplotypes frequencies estimation is in av-

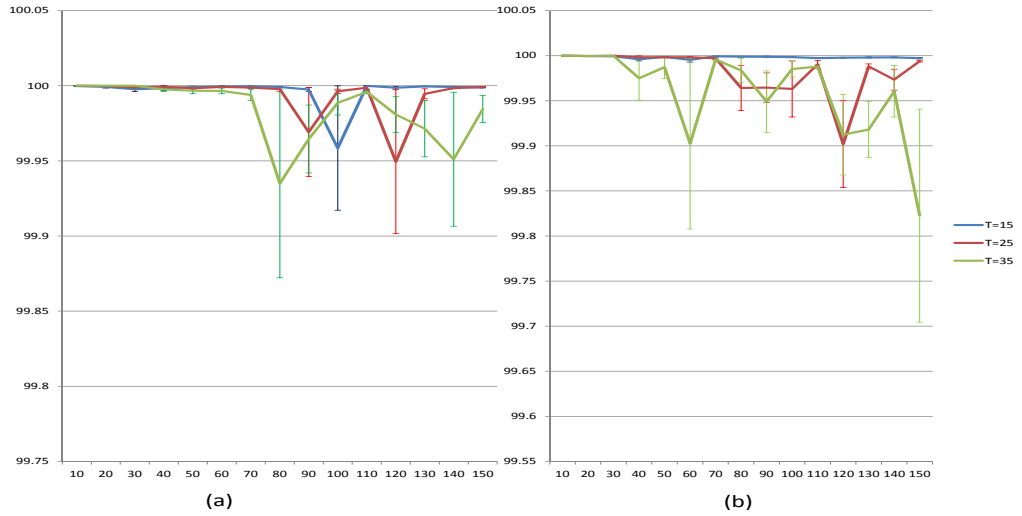


Figure (4.5) (a) Percentage of classified reads (b) Percentage of correctly classified reads. Bars represent a standard error [4].

erage 0.031%-0.107% for $T = 15$, 0.025%-0.139% for $T = 25$ and 0.028%-0.174% for $T = 35$; it is an increasing function of the number of samples (Figure 4.7).

According to all measures considered above the accuracy of the samples deconvolution is affected by the number of allowed samples per pool. The algorithm is more accurate for smaller pools, although the accuracy remain high even for larger pools.

4.6 Experimental validation of pooling strategy

4.6.1 Experimental pools and sequencing

Serum specimens collected from HCV-positive cases [30] were used to sequence HCV HVR1. Seven serum samples $S_1, ..., S_7$ were mixed to form 4 pools $P_1, ..., P_4$ using the VSPD algorithm with the parameter $T = 7$ as follows: P_1 was created by mixing samples S_1, S_2, S_3, P_2 - samples S_4, S_5, S_6, S_7, P_3 - samples S_1, S_4, S_5 and P_4 - samples S_2, S_4, S_6. Then the seven specimens and 4 pools were sequenced using 454 GS Junior System (454 Life Sciences, Branford, CT). Total nucleic acids extraction was performed using MagNA

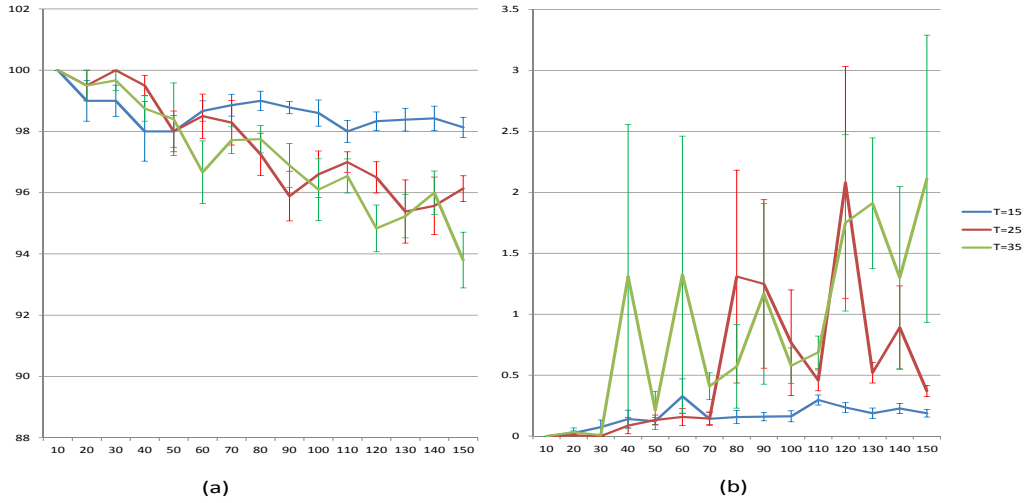


Figure (4.6) (a) Percentage of samples without in silico contamination. (b) Total frequency of in silico contaminants within contaminated samples. Bars represent a standard error [4].

Pure LC Total Nucleic Acid Isolation Kit (Roche Diagnostics, Mannheim, Germany) and reverse-transcribed using the SuperScript Vilo cDNA synthesis kit (Invitrogen, Carlsbad, CA).

The HVR1 amplification was accomplished using two rounds of PCR. For the 1st round of amplification, regular region-specific primers were used. Forward and reverse tag sequences consisting of primer adaptors and multiple identifiers (MID - 454A and 454B) were added to the HVR1-specific nested primers. For the high throughput purpose, pools were processed as a single specimen, tagged with a single MID for deep sequencing. PCR products were pooled and amplified by emulsion PCR using the GS FLX Titanium Series Amplicon kit, and bi-directionally sequenced. The sequenced reads were identified and separated using sample-specific MID tag identifiers. Low quality reads were removed using the GS Run Processor v2.3 (Roche, 2010).

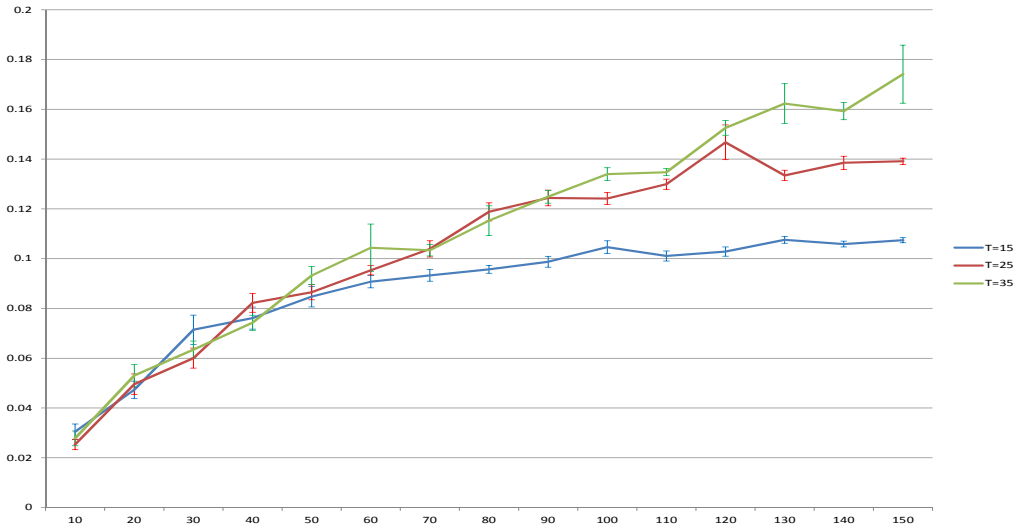


Figure (4.7) Root Mean Square Error of haplotypes frequencies estimation. Bars represent a standard error [4].

4.6.2 Experimental results

The algorithmic approach described in Section 4.4 was used to reconstruct 7 samples from experimental data described in Subsection 4.6.1. Before applying algorithms for samples recovery, the data were preprocessed in order to get rid of sequencing errors and PCR chimeras.

The reads from each pool were separated into clusters using Algorithm kGEM, each cluster was processed using NGS error correction algorithms KEC and ET [150] and the corrected reads were merged back. Then the samples were deconvoluted using Algorithm IS. The obtained samples will be further referred as *pooling samples*

For the verification of pooled samples we compared them with the individually sequenced samples. The sequences were compared using pairwise alignment; insertions and deletions were ignored (since indels are rare in HVR1 and, therefore, are rather sequencing artifacts; moreover, some indels in alignment of sequences from individually sequenced and pooling samples may be introduced due to the inaccurate correction of ho-

mopolymer errors for the samples). For each sample 10 reference sequences were taken from the set of individually sequenced variants, and the correctness of samples reconstruction was assessed using alignment of sequences in the pooled samples with these references. For alignment, Muscle [151] was used.

In average, 259 unique haplotypes per sample from a pool were obtained (from 23 haplotypes in Sample S_2 to 548 haplotypes in Sample S_4), which exceeds the number of HCV haplotypes obtained in other studies [152–154] after the standard individual sequencing using 454 Junior System and subsequent error correction. 99.9634% (5463 of 5465) of all analyzed sequence reads were correctly classified to the samples. Two reads assigned to sample S_7 showed a higher similarity to the reference sequence from Sample S_6. However, the subsequent analysis showed that these reads are only marginally similar to sequences from both individually sequenced samples as well to each other (minimum distance from these reads to the closest haplotype from S_6 and S_7 is 25 and 26, respectively, and the distance between them is 20, while the mean distance among individually sequenced haplotypes of samples S_6 or S_7 is 3.64 bp or 6.12 bp with standard deviations 1.21 bp or 5.25 bp, respectively). Therefore, these 2 reads are likely to be sequencing artifacts, which were not removed by the error correction algorithm.

In general, the percentage of haplotypes from individually sequenced samples found in pooled samples was not high (Figure 4.8, (a)), with an average of 14.66%. However, when the frequencies of these haplotypes were considered, the level of agreement between samples was much higher, with an average total haplotypes frequency of 56.94% (Figure 4.8, (b)). In particular, all individually sequenced haplotypes with frequencies greater than 10% and 72.73% of haplotypes with frequencies greater than 5% were found in pooled samples.

The differences between haplotype frequency distributions for individually sequenced and pooled samples were measured using Jensen-Shannon Divergence (JSD) [155] and correlation coefficient (Table 4.1). JSD varies from 0.15% for the sample S_1 to 0.65% for the sample S_7. There is a statistically significant positive correlation between

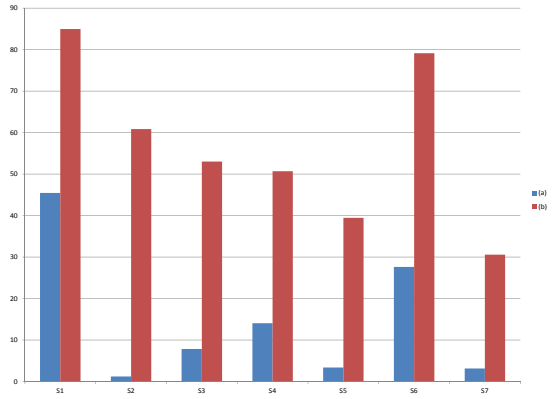


Figure (4.8) (a) Percentage of haplotypes from individually sequenced samples found in pooling experiment. (b) Total frequency of haplotypes from individually sequenced samples found in pooling experiment [4].

Table (4.1) Comparison of frequency distributions for individually sequenced and pooled samples

	JSD	Correlation (p-value)
S_1	0.15	0.95 ($1.7\dot{1}0^{-77}$)
S_2	0.57	0.30 (0.0023)
S_3	0.32	0.89 ($2.7\dot{1}0^{-173}$)
S_4	0.37	0.66 ($4.14\dot{1}0^{-99}$)
S_5	0.50	0.25 ($8.6\dot{1}0^{-7}$)
S_6	0.17	0.99 (0)
S_7	0.65	-0.07 (0.16)

frequency distributions for samples S_1-S_6. The only exception is the sample S_7, in which a large cluster of viral variants was not detected in the individually sequenced specimen but was found in the pooling experiment (see Figure 4.9).

Phylogenetic trees of viral populations from samples S_1-S_7 obtained by individual and pool sequencing are shown in Figure 4.9. Although haplotypes obtained from 2 different sequencing experiments are not completely matching, they cover the same areas of the sequence space. Some tree branches are formed by variants sequenced in one experiment but not another. For instance, sequencing of individual samples S_1 and S_2 produced sequences forming branches that cannot be found when sequences from pooling experi-

ments were considered. The opposite was observed for samples S_6 and S_7.

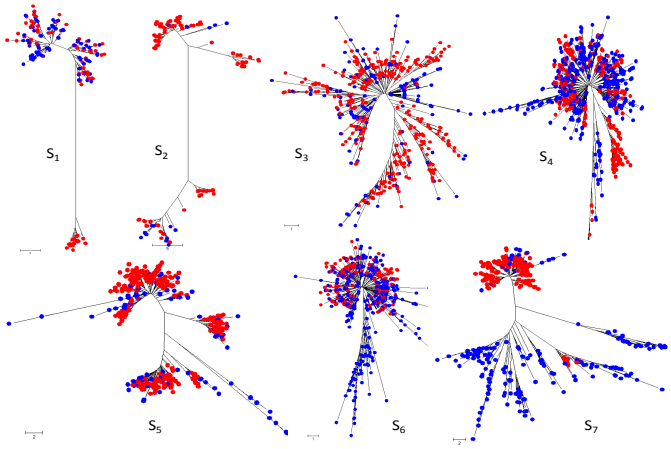


Figure (4.9) Phylogenetic trees of viral populations from samples S_1-S_7. Haplotypes obtained by individual sequencing of samples are shown in red, and haplotypes obtained from sequencing of pools are shown in blue [4].

4.7 Conclusions

In this study, we present a novel framework for massive NGS of highly mutable RNA viruses, such as HCV and HIV. To the best of our knowledge, this is the first application of the pooling strategy to highly heterogeneous viral samples. The developed framework takes into account specific aspects of viral sequencing, such as the extensive heterogeneity of viral samples, the large number of distinct viral variants sequenced from each sample and the effects of PCR and sampling biases. The proposed strategy is highly effective in reducing the number of sequencing runs, while still providing sufficient amount of information in support of molecular surveillance and numerous other applications of viral sequences in clinical and epidemiological settings. The novel clustering algorithm developed here significantly facilitates assignment of intra-host viral variants from massive sequence datasets obtained by pooling specimens to individual patients. The strategy of overlapping pools drastically reduces the cost of sequencing per specimen, especially when large numbers of specimens require to be tested. This computational framework is

applicable to viral agents infecting humans and animals and, with further development of the experimental protocols, it should serve as a cost-effective foundation for accurate molecular surveillance of infectious diseases.

Ultra-deep sequencing of viral samples produces a wide range of intra-host viral variants and allows for detecting minority variants, some of which have been shown to have important clinical implications such as drug resistance [33, 141, 142]. Pooling of numerous specimens reduces the depth of sequencing for each specimen. However, this reduction is not as detrimental for identifying minor viral variants since each specimen is usually used in more than one pool in the strategy developed here. As specimen is tested more than once, the number of sequenced variants is increased, so representative sampling of viral subpopulations infecting each patient can be improved. The experiments conducted here showed that comparable number of haplotypes were recovered from individual specimens and from pools (Fig. 4.9), at least at the pooling scale used in this study. Both individual sequencing and pooling produce sequences covering approximately the same areas of the sequence space, thus providing a consistent structure of a viral population.

Repeat sampling from the same complex viral population results frequently in poorly matched sets of viral sequences, thus presenting a significant challenge to assignment of all sequences obtained by pool sequencing to each patient. Such stochastic sampling has a potential to diminish the effectiveness of pool-sequencing and usefulness of the obtained sequences by impeding the correct allocation of sequences to samples, leaving some samples without sequences assigned or allocating only a fraction of the obtained sequences to samples. The clustering-based approach to finding generalized intersections of pools developed in this study significantly improves identification of sequences that belong to a patient and, thus, not only substantially overcomes the aforementioned potential pitfalls, but converts stochastic sampling into an advantage.

The cost of sequencing and accuracy of pool deconvolution are two major measures of quality of our computational framework. However, these two measures are in conflict

with each other. While increase in pool size improves cost-effectiveness of sequencing by reducing the number of sequencing runs, it reduces accuracy of deconvolution. Considering that accuracy of deconvolution significantly depends on the genetic complexity of intra-host viral populations, an optimal pool size should be carefully selected for each virus and each genomic region.

In conclusion, success of the pool-based mass-sequencing of viral populations depends to a significant degree on the efficacy of sequence assignments and the risk of under-representation of viral variants from some patients, owing to PCR and sample biases. The pool-design and clustering algorithms presented here substantially minimize the detrimental effect of these biases on quality of the mass-sequencing. However, the further reduction of the biases using, for example, generalizations of error-correcting codes and optimization of experimental conditions, should further improve the strategy, facilitating its application to molecular surveillance and study of infectious diseases.

4.8 Software package

Our framework is available online and may be freely used for all non-commercial purposes. http://alan.cs.gsu.edu/NGS/?q=content/pooling

PART 5

ALGORITHMS FOR PREDICTION OF VIRAL TRANSMISSIONS

5.1 Introduction

Inferring transmission clusters and transmission directions from viral sequencing data is crucial for viral outbreaks investigation. Outbreaks of RNA viruses such as as Human Immunodeficiency Virus (HIV) and Hepatitis C virus (HCV) are especially dangerous and pose a significant problem for public health. It is well known that genomes of RNA viruses mutate at extremely high rates [156]. As a result RNA viruses exist in infected hosts as populations of closely related variants called *quasispecies* [157,158]. However only recently with the progress of NGS sequencing technologies it became possible to identify and sample quasispecies at great depth [159–164]. Still, consensus-based methods remain the most common for HIV and HCV outbreaks investigations [87]. Such methods (referred further *consensus-based cutoff* (CBC)) link two hosts by transmission if the distances between representative sequences of their intra-host populations (usually consensus sequences) do not exceed a predefined cutoff. Although CBC methods are useful and simple to implement, they have several limitations:

- Minority viral variants are frequently responsible for transmission of HCV infections [95,96] but could not be easily detected by CBC methods. Although the nature of lower frequency variant transmittance is not completely understandable, the evidence suggests that more frequent variants are less likely to be transmitted because they already dominate the sequence space in the host and highly adapted to that environment which makes them less viable in the naive host environment [97].

- Cutoff values utilized by CBC methods are derived experimentally. Those cutoffs are virus specific, and sometimes even region specific of the virus, therefore different

cutoff values should be considered for the same task. It is also should be noted the experimental data is often incomplete or compromised. The pre-set cutoffs could be too conservative/strict and thus missing potential cases in outbreak surveillance programs.

- CBC methods cannot infer transmission directions which is crucial for detection of outbreak sources and transmission histories. To our knowledge the directions of transmissions were never inferred automatically, rather relying on "expert eye" analysis [92–94] or some additional information which is assumed to be known.

We address the above limitations by proposing two novel algorithms ReD and VOICE.

- Relatedness Depth (ReD) algorithm identifies viral transmission clusters, transmissions and their directions using clustering-based analysis of whole intra-host viral populations. Algorithm ReD is non-parametric, i.e. it does not rely on a specific cutoff value to infer transmissions.

- Viral Outbreak InferenCE (VOICE) infers possible transmissions and their directions between two given viral populations. VOICE is a simulation-based method which imitates viral evolution.

ReD and VOICE algorithms can be applied to infer possible transmissions, to identify their directions, and to predict sources of outbreaks. We evaluate our algorithms on the experimental data obtained from HCV outbreaks. Comparative results suggest that our methods are more effective than CBC in detecting transmission clusters and outbreak sources.

5.2 Methods

5.2.1 Relatedness depth (ReD) algorithm

The key concept of this method is k-*clustered intersection* of viral populations (we used similar idea previously for combinatorial pooling [165]). For two sets of viral sequences P_1 and P_2, the k-clustered intersection $P_1 \overline{\cap} P_2$ is calculated as follows:

1) partition the union $P_1 \cup P_2$ into k clusters $C_1, ..., C_k$;

2) $P_1 \overline{\cap} P_2 = \bigcup_{i \in B} C_i$, where $B = \{i \in \{1, ..., k\} : C_i \cap P_1 \neq \emptyset, C_i \cap P_2 \neq \emptyset\}$, i.e. $P_1 \overline{\cap} P_2$ is the union of clusters, which contain sequences from both P_1 and P_2 (see Fig. 5.1).

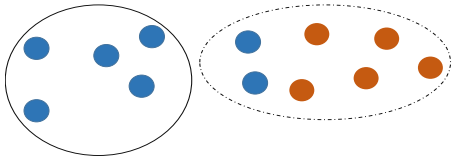


Figure (5.1) Population intersection of two viral populations (blue and red).
Union of populations is partitioned into $k = 2$ clusters (dashed and solid).
Dashed cluster is the k-clustered intersection.
Direction of transmission is from blue to red population.

The parameter k is a *scale* of clustering. In particular, populations P_1 and P_2 are *separable*, if $P_1 \overline{\cap} P_2 = \emptyset$, while the fact that $P_1 \overline{\cap} P_2 \neq \emptyset$ indicates that they may be genetically related. In the most extreme case $P_1 \overline{\cap} P_2 = P_1 \cup P_2$, i.e. populations are *completely inseparable* under the scale k.

The degree of confidence that the samples are genetically close is represented by the *relatedness depth* $d(P_1, P_2)$, which is calculated by Algorithm 1.

Simply speaking, Algorithm 1 tries to recursively separate populations P_1 and P_2. At each iteration, k-clustered intersection is calculated. If two populations are separable then the algorithm stops. Otherwise, it continues the separation of sequences from P_1 and P_2 within their k-clustered intersection. The separation depth is a depth of this recursion.

Algorithm 1 Relatedness depth calculation

Input Two sets of viral sequences P_1, P_2.
Output Relatedness depth $d = d(P_1, P_2)$

```
d ← 0
k ← 2
I ← P1 ⋒ P2
while I ≠ ∅ and k ≤ |P1| + |P2| do
    d ← d + 1
    if I ≠ P1 ∪ P2 then
        P1 ← P1|I, P2 ← P2|I
        k ← 2
    else
        k ← k + 2
    end if
    I ← P1 ⋒ P2
end while
```

It is possible that at some iterations of Algorithm 1 two populations are completely inseparable under a current clustering scale. In this case the scale k is increased and k-clustered intersection is recalculated. The initial value of k used by Algorithm 1 is $k = 2$.

k-clustered intersections depend on a clustering method. In our implementation, a hierarchical clustering using neighbor-joining tree based on Jukes-Cantor distance was used (as implemented in Matlab (MathWorks, Natick, MA)).

Clustered intersections also allow for estimating the direction of transmissions. It is reasonable to assume that if two hosts share a population, then a host with more heterogeneous population is more likely to be the transmission source [166]. Formally, if $I = P_1 \overline{\cap} P_2$, $P_1 \subseteq I$ and $P_2 \setminus I \neq \emptyset$, then we assume that probable transmission direction is from P_2 to P_1 (see Fig. 5.1). The direction is defined according to the first occurrence of such situation during execution of Algorithm 1. Note that in some cases direction may not be identified.

Identification of transmission clusters and sources of outbreaks. Given the collection of viral populations $\mathcal{P} = \{P_1, ..., P_n\}$, ReD produces the weighted directed genetic relatedness graph $G = (V, A, w)$ with $V = \mathcal{P}$. An arc P_iP_j is in A whenever populations P_i and P_j

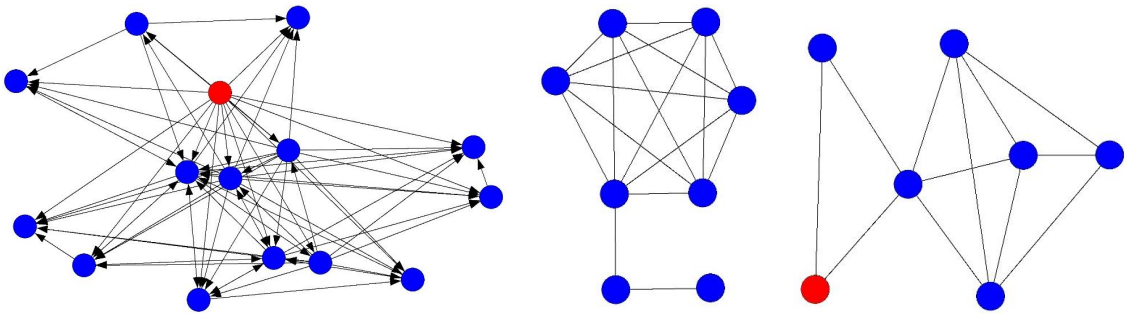


Figure (5.2) Transmission clusters for AI outbreak estimated by ReD and consensus-based algorithm. The known outbreak source is shown in red.

are genetically related, i.e., have sufficiently high relatedness depth; the direction of an arc corresponds to the estimated direction of transmission. In this work, the simplest possible relatedness depth cutoff $T_1 = 1$ was used for ReD algorithm.

Transmission clusters are calculated as weakly connected components of the genetic relatedness graph G. Only components containing at least one edge e of weight $w(e) \geq 2$ were considered as reliable. For each connected component, a source s of the corresponding outbreak identified as a vertex with highest eigenvector centrality.

5.2.2 Viral outbreak inference (VOICE) simulation method

Given two intra-host viral populations P_1 and P_2, VOICE simulates viral evolution in order to estimate times t_{12} and t_{21} needed to acquire an observed genetic heterogeneity under the assumption, that first and second host were sources of infection. We then decide if the intra-host viral populations are related based on the value of $\min\{t_{12}, t_{21}\}$. If they are related, the direction of transmission is assumed to follow the direction which requires less time to evolve.

Two main steps of VOICE are (1) construction of a viral network over observed viral populations in two hosts and (2) running simulation of viral evolution in the viral network starting from variants in the first and the second populations.

Viral Network Construction. The network consists of nodes representing viral variants and edges connecting related variants. In addition to vertices corresponding to observed viral variants we add *median* vertices, which correspond to consensus sequences for all triplets of observed variants.

The edges in the viral network are built by the following procedure. First we start with complete graph G, with weights of its edges being Hamming distances between viral sequences corresponding to its end-nodes. Next, minimum spanning tree T of G is calculated, and edges with weights exceeding maximal weight of an edge of T are removed from G. Finally, edges uv with weights $k > 1$ are subdivided into k edges of weights 1 by adding $k - 1$ additional vertices which represent "invisible" viral variants on the path

of virus evolution from u to v. It is important to note that we fix the mutations corresponding to those vertices "on the go". This allows us to account for random mutation happening at any position, and there is no need to store all possible variants.

Simulation of Viral Evolution. Suppose that the scenario when the first host is the source of infection is considered. We define a *border* set B_1 as the set of vertices of P_1 minimizing pairwise distance between vertices from P_1 and P_2:

$$B_1 = \{u \in P_1 : d(u, v) = \min_{x \in P_1, y \in P_2} d(x, y) \text{ for some } v \in P_2\} \tag{5.1}$$

B_1 represent viral variants likely to be closest to variants that were transmitted.

Then the viral evolution is simulated as following. The simulation starts from all border nodes B_1 and run until all the nodes of the population P_2 are reached. At the beginning of simulation border nodes get count equal to 1, and the rest of the nodes get count 0. At each tact of the simulation node counts are updated according to one of the following scenarios happening with some probability. First, the node could replicate itself, in which case its count label is incremented. Second, the node can mutate into one of its neighboring nodes, in which case the count of that node is incremented. Third, the node might die due to immune response of the host organism, in which case its count is decremented. The probabilities p_1,p_2 and p_3 of these scenarios are calculated as follows:

$$p_1 = (1 - \delta)(1 - 3\epsilon)^L, p_2 = p_1 \frac{\epsilon}{1 - 3\epsilon}, p_3 = \delta, \tag{5.2}$$

where ϵ is an error rate (default value $\epsilon = 0.03$), δ is the death probability (default value $\delta = \epsilon$) and L is the genome length.

We run s simulations (default value $s = 5$) assuming each border node as transmission source. We then take average time t_{12} of s simulations, and finally, minimal value of all averages. Same procedure repeated for the other possible direction of transmission with its own border set, where we calculate t_{21}. The value $\min\{t_{12}, t_{21}\}$ will determine which direction of transmission is more likely.

Data normalization. The size of observed intra-host viral populations may significantly vary. Therefore, VOICE will be biased in estimation of t_{12} and t_{21} since larger population will require more time to cover. In order to avoid such bias VOICE normalizes the data by adjusting the intra-host population size. Namely, across all viral populations from the same outbreak we find one with minimal number q of quasispecies. For all other populations then the set of quasispecies is clustered into q clusters. This is done by finding a consensus among sequences from same cluster and CyPy Average linkage hierarchical clustering.

5.3 Experimental results

Data Sets

For algorithms testing and comparison, two collections of HCV samples were analyzed.

1) Epidemiologically related samples. This collection contains 142 HCV samples from 33 epidemiologically curated outbreaks reported to Centers for Disease Control and Prevention in 2008-2013. Outbreak collections contain from 2 to 19 samples collected from cases infected with HCV subtypes 1a, 1b and 2a. All outbreaks were epidemiologically confirmed (see http://www.cdc.gov/ hepatitis/Outbreaks/Healthcare-HepOutbreakTable.htm). Sources of HCV infection are known for 10 outbreaks as a result of epidemiological investigations.

2) Unrelated samples. This collection contains HCV samples from infected individuals without any known epidemiological relationship, all obtained from national collections and other research projects [167].

For all samples, HCV hypervariable region 1 (HVR1) was used for assessment of intra-host viral populations. Nucleic acids extraction and PCR conditions were previously described [168]. HVR1 was sequenced using End-Point Limiting-Dilution Real-Time PCR (EPLD) protocol [168, 169]. Sequences from each sample were aligned using MAFFT [170] and the primers were removed, yielding a final region of 264bp.

Accuracy

Performances of ReD, VOICE and consensus-based transmission prediction algorithms were evaluated using clustering quality measures proposed in [171]. Let $\mathcal{S}$ be the set of samples, $\mathcal{S}^{(2)}$ be the set of all pairs of samples, $\mathcal{T} = \{T_1, ..., T_n\}$ be the partition of $\mathcal{S}$ into correct transmission clusters, and $\mathcal{U} = \{U_1, ..., U_m\}$ be the partition of $\mathcal{S}$ into transmission clusters estimated by an algorithm. For a partition $\mathcal{T}$ let $P_{\mathcal{T}} = \{\{x, y\} \in \mathcal{S}^{(2)} : x, y \in T_i \text{ for some } T_i \in \mathcal{T}\}$. The set $P_{\mathcal{U}}$ is defined analogously.

We evaluate the quality of relatedness prediction using the following two rates:

- the *true positive rate* (TPR) is a percentage of truly related pairs of samples predicted as related by an algorithm, i.e.

$$TPR = \frac{|P_{\mathcal{T}} \cap P_{\mathcal{U}}|}{|P_{\mathcal{T}}|}. \tag{5.3}$$

- the *false positive rate* (FPR) was calculated as percentage of truly unrelated pairs of samples predicted as related by an algorithm, i.e.

$$FPR = 1 - \frac{|\overline{P_{\mathcal{T}}} \cap \overline{P_{\mathcal{U}}}|}{|\overline{P_{\mathcal{T}}}|}, \tag{5.4}$$

  where $\overline{P_{\mathcal{T}}} = \mathcal{S}^{(2)} \setminus P_{\mathcal{T}}$, $\overline{P_{\mathcal{U}}} = \mathcal{S}^{(2)} \setminus P_{\mathcal{U}}$.

All values of a distance cutoff D for the CBC algorithm with 0.5% increment were considered, and two values $D = 4.5\%$ and $D = 6.5\%$ were chosen for the report for the following reason: $D = 4.5\%$ is the largest value, at which the consensus-based algorithm has a zero FPR on unrelated samples, and $D = 6.5\%$ is the smallest value, at which TPR of the consensus-based algorithm on related samples is comparable to ReD. ReD algorithm was implemented in Matlab (MathWorks, Natick, MA) and all related and unrelated samples were processed together.

The combined results of algorithms for related and unrelated samples are reported in Table 5.1. ReD achieves high quality TPR and FPR. In particular, it is able to correctly

Table (5.1) Combined results for related samples (33 clusters) and unrelated samples (193 samples)

Methods	Related samples				Unrelated samples		
	# predicted clusters	TPR	FPR	Source identification accuracy	# predicted clusters	TPR	FPR
ReD	37	98.96%	0%	90%	192	100%	0.01%
CBC[4.5%]	43	81.84%	0%	0%	193	100%	0%
CBC[6.5%]	38	96.66%	0%	10%	171	100%	1.37%

identify genetic relatedness for almost all pairs of samples, resulting in 98.96% sensitivity. At the same time, only 2 false positive connections were reported for ReD. ReD has significantly higher TPR than the consensus-based algorithm with 4.5% distance cutoff (CBC[4.5%]). The CBC[6.5%] has a higher TPR, but also a significantly higher FPR – it falsely identifies a large transmission cluster containing 23 samples.

At this moment VOICE algorithm can detect relatedness with the TPR rate equal to 91.5%. Currently the work is being done to improve the algorithm to be able to identify clusters of transmissions and sources of outbreaks. However it is the only algorithm which is able to estimate the time of transmissions. ReD algorithm was most accurate in identification of outbreak sources. It was able to correctly identify sources for 9 out of 10 outbreaks, while the consensus-based algorithms correctly identified sources in none or only 1 outbreak. All algorithms failed to detect the source of outbreak AQ. However, it should be noted that this outbreak was caused by blood transfusion, while the other outbreaks were associated with unsafe injection practices or contaminated equipment, which are completely different transmission mechanisms.

5.4 Conclusions

Currently, molecular viral analysis is one of the major tools used for investigations of outbreaks and inference of transmission networks. Although modern sequencing technologies significantly facilitated molecular analyses, providing unprecedented access to intra-host viral populations, they generated novel bioinformatics challenges for molecu-

lar surveillance. Replacement of simplistic consensus-based approaches and expert phylogenetic analyses with novel automatic algorithms using sequencing data for outbreak investigations is a major advancement in molecular surveillance of viral infections. Such algorithms must be highly accurate in prediction of transmission to be suitable in public health inquiries and forensic investigations.

Here, we presented two algorithms for the prediction of viral transmissions based on analysis of the intra-host viral populations, which allow not only to identify HCV populations genetically but also to estimate a possible direction of transmissions. Superior performance of the new algorithms over the state-of-the-art CBC algorithm in the prediction of transmissions using experimental data from actual HCV outbreaks indicates importance of full-fledged quasispecies analyses for viral molecular surveillance and outbreaks investigation.

The advantage of the new algorithms is especially evident in identification of sources of outbreaks. Transmission clusters identified using the CBC algorithm are often indirected cliques in the genetic-relatedness networks (everybody are close to everybody) and do not allow for distinguishing one of the vertices as a source. Moreover, even when a transmission cluster is not a clique, the source of an outbreak may not be its most central vertex, because of new infections being frequently established from minority intra-host viral subpopulations in the source. [95, 96]. An example of outbreak AI is particularly illustrative (Fig. 5.2). In this outbreak, the real source has a low degree and centrality in comparison to other viral samples in a transmission cluster identified using consensuses; at the same time it is central in cluster identified by ReD. In addition, the only viral sample in the outbreak BB that was not linked to the transmission cluster using the CBC[4.5%] is its actual source.

The simulation-based approach VOICE presented here may be further improved by incorporating more complex viral evolution models taking into account cell proliferation rate and immune responses against viral variants. The clustering-based ReD approach may be further improved using a more scalable clustering similar to the algorithm pro-

posed in [165]. All the algorithms are planned to be integrated into the pipeline of cloud-based web portal "Global Hepatitis Outbreak and Surveillance Technology" (GHOST) of Centers for Disease Control and Prevention, Atlanta, GA.

PART 6

DISCUSSION AND FUTURE WORK

It is important to further improve computational methods to accurately estimate the viral population structure. It will facilitate the preventative care and help understand virus evolution. As the NGS technologies become more and more fast and cost-efficient, computational methods should be adjusted to deal with big amount of data.

REFERENCES

[1] A. Caciula, O. Glebova, A. Artyomenko, S. Mangul, J. Lindsay, I. I. Măndoiu, and A. Zelikovsky, "Simulated regression algorithm for transcriptome quantification from rna-seq data," *BMC*, vol. under review.

[2] S. Mangul, A. Caciula, O. Glebova, I. Mandoiu, and A. Zelikovsky, "Improved transcriptome quantification and reconstruction from rna-seq reads using partial annotations," *In silico biology*, vol. 11, no. 5, pp. 251–261, 2011.

[3] W. J. Kent, C. W. Sugnet, T. S. Furey, K. M. Roskin, T. H. Pringle, A. M. Zahler, and DavidHaussler, "The human genome browser at ucsc," *Genome Research*, vol. 12, no. 6, pp. 996–1006, 2002. [Online]. Available: http://genome.cshlp.org/content/12/6/996.abstract

[4] P. Skums, A. Artyomenko, O. Glebova, S. Ramachandran, I. Mandoiu, D. S. Campo, Z. Dimitrova, A. Zelikovsky, and Y. Khudyakov, "Computational framework for next-generation sequencing of heterogeneous viral populations using combinatorial pooling," *Bioinformatics*, vol. 31(5), pp. 682–690, 2015.

[5] M. L. Metzker, "Sequencing technologies - the next generation." *Nature reviews. Genetics*, vol. 11, no. 1, pp. 31–46, 2010.

[6] T. C. Glenn, "Field guide to next-generation DNA sequencers," *Molecular Ecology Resources*, 2011.

[7] J. W. Drake and J. J. Holland, "Mutation rates among RNA viruses," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 96, no. 24, pp. 13 910–13 913, 1999. [Online]. Available: http://www.pnas.org/content/96/24/13910.abstract

[8] E. Domingo and J. Holland, "RNA virus mutations and fitness for survival," *Annu Rev Microbiol*, vol. 51, pp. 151ï£¡–178, 1997.

[9] E. Domingo, M.-S. E., F. Sobrino, J. de la Torre, A. Portela, J. Ortin, C. Lopez-Galindez, P. Perez-Brena, N. Villanueva, and R. Najera, "The quasispecies (extremely heterogeneous) nature of viral rna genome populations: biological relevance – review," *Gene, 40*, pp. 1–8, 1985.

[10] M. E. M, J. McCaskill, and P. Schuster, "The molecular quasi-species," *Adv Chem Phys*, vol. 75, pp. 149ï£¡–263, 1989.

[11] M. Martell, J. Esteban, J. Quer, J. Genesca, A. Weiner, R. Esteban, J. Guardia, and J. Gomez, "Hepatitis c virus (hcv) circulates as a population of different but closely related genomes: quasispecies nature of hcv genome distribution," *Journal of Virology, 66*, pp. 3225–3229, 1992.

[12] D. Steinhauer and J. Holland, "Rapid evolution of rna viruses," *Annual Review of Microbiology, 41*, pp. 409–433, 1987.

[13] N. Beerenwinkel, T. Sing, T. Lengauer, J. Rahnenfuehrer, and K. R. et al., "Computational methods for the design of effective therapies against drug resistant HIV strains," *Bioinformatics*, vol. 21, pp. 3943ï£¡–3950, 2005.

[14] N. G. Douek DC, Kwong PD, "The rational design of an AIDS vaccine." *Cell*, vol. 124, pp. 677–681, 2006.

[15] B. Gaschen, J. Taylor, K. Yusim, B. Foley, and F. G. et al., "Diversity considerations in HIV-1 vaccine selection," *Science*, vol. 296, pp. 2354ï£¡–2360, 2002.

[16] J. Holland, J. de la Torre, and D. Steinhauer, "Rna virus populations as quasispecies," *Current Topics in Microbiology and Immunology, 176*, pp. 1–20, 1992.

[17] S.-Y. Rhee, T. Liu, S. Holmes, and R. Shafer, "HIV-1 subtype B protease and reverse transcriptase amino acid covariation," *PLoS Comput Biol*, vol. 3, p. e87, 2007.

[18] N. Eriksson, L. Pachter, Y. Mitsuya, S. Rhee, and C. W. et al., "Viral population estimation using pyrosequencing," *PLoS Comput Biol*, vol. 4, p. e1000074, 2008.

[19] J. Archer, M. Braverman, B. Taillon, B. Desany, I. James, P. Harrigan, M. Lewis, and D. Robertson, "Detection of low-frequency pretherapy chemokine (CXC motif) receptor 4 (CXCR4)-using HIV-1 with ultra-deep pyrosequencing," *AIDS*, vol. 23, no. 10, pp. 1209–1218, 2009.

[20] C. Hoffmann, N. Minkah, J. Leipzig, G. Wang, M. Arens, P. Tebas, and F. Bushman, "DNA bar coding and pyrosequencing to identify rare HIV drug resistance mutations," *Nucleic Acids Research*, vol. 35, no. 13, 2007, cited By (since 1996) 52. [Online]. Available: http://www.scopus.com/inward/record.url?eid=2-s2.0-34547829263&partnerID=40&md5=8c9e9dcc7a2acd6b5b3c6d09eaee7b9a

[21] J. Simons, M. Egholm, J. Lanza, B. Desany, and G. T. et al., "Ultradeep sequencing of HIV from drug resistant patients," *Antivir Ther*, vol. 10, p. S157, 2005.

[22] A. Tsibris, C. Russ, W. Lee, R. Paredes, and R. A. et al., "Detection and quantification of minority HIV-1 env V3 loop sequences by ultra-deep sequencing: Preliminary results," *Antivir Ther*, vol. 11, p. S74, 2006.

[23] W. C, M. Y, G. B, R. M, and S. RW, "Characterization of mutation spectra with ultra-deep pyrosequencing: application to HIV-1 drug resistance." *Genome Res*, vol. 17, pp. 1195–1201, 2007.

[24] J. Z. Li and *et al.*, "Comparison of illumina and 454 deep sequencing in participants failing raltegravir-based antiretroviral therapy." *PLoS One*, vol. 9, no. 3, p. e90485, 2014. [Online]. Available: http://www.biomedsearch.com/nih/Comparison-illumina-454-deep-sequencing/24603872.html

[25] O. Zagordi, A. Bhattacharya, N. Eriksson, and N. Beerenwinkel, "Shorah: estimating the genetic diversity of a mixed sample from next-generation

sequencing data," *BMC Bioinformatics*, vol. 12, no. 1, p. 119, 2011. [Online]. Available: http://www.biomedcentral.com/1471-2105/12/119

[26] I. Astrovskaya, B. Tork, S. Mangul, K. Westbrooks, I. Mandoiu, P. Balfe, and A. Zelikovsky, "Inferring viral quasispecies spectra from 454 pyrosequencing reads," *BMC Bioinformatics*, vol. 12, no. Suppl 6, p. S1, 2011. [Online]. Available: http://www.biomedcentral.com/1471-2105/12/S6/S1

[27] P. MC and S. M., "Qure: software for viral quasispecies reconstruction from next-generation sequencing data." *Bioinformatics*, vol. 28, no. 1, pp. 132–3, 2012.

[28] N. Mancuso, B. Tork, P. Skums, L. Ganova-Raeva, I. Mandoiu, and A. Zelikovsky, "Reconstructing viral quasispecies from ngs amplicon reads," *In Silico Biology*, vol. 11, no. 5, pp. 237–249, 2012.

[29] P. Skums, N. Mancuso, A. Artyomenko, B. Tork, I. Mandoiu, Y. Khudyakov, and A. Zelikovsky, "Reconstruction of viral population structure from next-generation sequencing data using multicommodity flows," *BMC Bioinformatics*, vol. 14, no. Suppl 9:S2, 2013.

[30] H. M., O. G., S. P.L., L. C.A., K. Y.E., X. G., L. Y., V. R., D. W.E., D. V.J., and C. G.M., "Results from a large-scale epidemiologic look - back investigation of improperly reprocessed endoscopy equipment," *Infect Control Hosp Epidemiol*, vol. 33, no. 7, pp. 649–656, 2012.

[31] V. Gilberto, G. Xia, J. C. Forbi, M. A. Purdy, L. M. G. Rossi, P. R. Spradling, and Y. E. Khudyakov, "Genetic relatedness among hepatitis a virus strains associated with food-borne outbreaks," *PLoS ONE*, vol. 8, no. 11, 2013.

[32] J. Wertheim, A. Leigh Brown, N. Hepler, S. Mehta, D. Richman *et al.*, "The global transmission network of hiv-1," *J Infect Dis*, vol. 209, no. 2, pp. 304–313, 2014.

[33] D. Campo, P. Skums, Z. Dimitrova, G. Vaughan, J. Forbi, C.-G. Teo, Y. Khudyakov, and D. T.-Y. Lau, "Drug-resistance of a viral population and its individual intra-host variants during the first 48 hours of therapy," *Clinical Pharmacology and Therapeutics*, vol. 95, no. 6, pp. 627–635, Jun 2014.

[34] W. Wang, X. Zhang, Y. Xu, G. Weinstock, A. Di Bisceglie *et al.*, "High- resolution quantification of hepatitis c virus genome-wide mutation load and its correlation with the outcome of peginterferon-alpha2a and ribavirin combination therapy," *PLoS ONE*, vol. 9, no. 6, 2014.

[35] I. Dierynck, K. Thy, A. Ghys, J. C. Sullivan, T. L. Kieffer, J. Aerssens, G. Picchio, and S. D. Meyer, "Deep sequencing analysis of the hcv ns3?4a region confirms low prevalence of telaprevir-resistant variants at baseline and end of the realize study," *J Infect Dis*, 2014.

[36] S. Ramachandran, D. Campo, Z. Dimitrova, G. Xia, M. Purdy, and Y. Khudyakov, "Temporal variations in the hepatitis c virus intrahost population during chronic infection," *J Virol*, vol. 85, no. 13, pp. 6369–6380, Jul 2011.

[37] B. A. Palmer, I. Moreau, J. Levis, C. Harty, O. Crosbie, E. Kenny-Walsh, and L. J. Fanning, "Insertion and recombination events at hypervariable region 1 over 9.6 years of hepatitis c virus chronic infection," *J Gen Virol*, vol. 93, pp. 2614–2624, 2012.

[38] A. Culasso, P. Bare, N. Aloisi, M. Monzani, M. Corti, and R. Campos, "Intra- host evolution of multiple genotypes of hepatitis c virus in a chronically infected patient with hiv along a 13-year follow-up period," *Virology*, vol. 449, pp. 317–327, Jan 2014.

[39] C. Trapnell, B. Williams, G. Pertea, A. Mortazavi, G. Kwan, M. van Baren, S. Salzberg, B. Wold, and L. Pachter, "Transcript assembly and quantification by RNA-Seq reveals unannotated transcripts and isoform switching during cell differentiation." *Nature biotechnology*, vol. 28, no. 5, pp. 511–515, 2010. [Online]. Available: http://dx.doi.org/10.1038/nbt.1621

[40] M. Guttman, M. Garber, J. Levin, J. Donaghey, J. Robinson, X. Adiconis, L. Fan, M. Koziol, A. Gnirke, C. Nusbaum, J. Rinn, E. Lander, and A. Regev, "*Ab initio* reconstruction of cell type–specific transcriptomes in mouse reveals the conserved multi-exonic structure of lincRNAs," *Nature Biotechnology*, vol. 28, no. 5, pp. 503–510, 2010. [Online]. Available: http://dx.doi.org/10.1038/nbt.1633

[41] W. Li, J. Feng, and T. Jiang, "IsoLasso: A LASSO Regression Approach to RNA-Seq Based Transcriptome Assembly," *Journal of Computational Biology*, vol. 18, no. 11, pp. 1693–707, 2011. [Online]. Available: http://online.liebertpub.com/doi/full/10.1089/cmb.2011.0171

[42] S. Mangul, A. Caciula, S. Al Seesi, D. Brinza, A. R. Banday, R. Kanadia, I. Mandoiu, and A. Zelikovsky, "An integer programming approach to novel transcript reconstruction from paired-end rna-seq reads," *ACM Conference on Bioinformatics, Computational Biology and Biomedicine*, 2012.

[43] A. Mortazavi, B. Williams, K. McCue, L. Schaeffer, and B. Wold, "Mapping and quantifying mammalian transcriptomes by RNA-Seq." *Nature methods*, 2008. [Online]. Available: http://dx.doi.org/10.1038/nmeth.1226

[44] E. Wang, R. Sandberg, S. Luo, I. Khrebtukova, L. Zhang, C. Mayr, S. Kingsmore, G. Schroth, and C. Burge, "Alternative isoform regulation in human tissue transcriptomes." *Nature*, vol. 456, no. 7221, pp. 470–476, 2008. [Online]. Available: http://dx.doi.org/10.1038/nature07509

[45] M. Nicolae, S. Mangul, I. Mandoiu, and A. Zelikovsky, "Estimation of alternative splicing isoform frequencies from rna-seq data," *Algorithms for Molecular Biology*, vol. 6:9, 2011. [Online]. Available: http://www.almob.org/content/6/1/9

[46] J. F. Degner, J. C. Marioni, A. A. Pai, J. K. Pickrell, E. Nkadori, Y. Gilad, and J. K. Pritchard, "Effect of read-mapping biases on detecting allele-specific expression from rna-sequencing data." *Bioinformatics*, vol. 25, no. 24, pp. 3207–3212,

2009. [Online]. Available: http://dblp.uni-trier.de/db/journals/bioinformatics/bioinformatics25.html#DegnerMPPNGP09

[47] C. Gregg, J. Zhang, J. Butler, D. Haig, and C. Dulac, "Sex-specific parent-of-origin allelic expression in the mouse brain," *Science*, vol. 329, no. 5992, pp. 682–685, 2010.

[48] C. McManus, J. Coolon, M. Duff, J. Eipper-Mains, B. Graveley, and P. Wittkopp, "Regulatory divergence in drosophila revealed by mrna-seq," *Genome research*, vol. 20, no. 6, pp. 816–825, 2010.

[49] J. Duitama, P. Srivastava, and I. Măndoiu, "Towards accurate detection and genotyping of expressed variants from whole transcriptome sequencing data," *BMC genomics*, vol. 13, no. Suppl 2, p. S6, 2012.

[50] Z. Wang, M. Gerstein, and M. Snyder, "RNA-Seq: a revolutionary tool for transcriptomics." *Nat. Rev. Genet.*, vol. 10, no. 1, pp. 57–63, 2009. [Online]. Available: http://dx.doi.org/10.1038/nrg2484

[51] T. R. Mercer, D. J. Gerhardt, M. E. Dinger, J. Crawford, C. Trapnell, J. A. Jeddeloh, J. S. Mattick, and J. L. Rinn, "Targeted RNA sequencing reveals the deep complexity of the human transcriptome." *Nature Biotechnology*, vol. 30, no. 1, pp. 99–104, 2012.

[52] B. Li, V. Ruotti, R. Stewart, J. Thomson, and C. Dewey, "Rna-seq gene expression estimation with read mapping uncertainty," *Bioinformatics*, vol. 26, no. 4, pp. 493–500, 2010.

[53] M. Lefmann, C. Honisch, S. Böcker, N. Storm, F. von Wintzingerode, C. Schlötelburg, A. Moter, D. van den Boom, and U. B. Göbel, "Novel mass spectrometry-based tool for genotypic identification of mycobacteria." *Journal Of Clinical Microbiology*, vol. Jan., pp. 339–346, 2004.

[54] P. Stanssens, M. Zabeau, G. Meersseman, G. Remes, Y. Gansemans, N. Storm, R. Hartmer, C. Honisch, C. P.Rodi, S. Böcker, and D. van den Boom, "High-

throughput maldi-tof discovery of genomic sequence polymorphisms." *Genome Res.*, vol. 14, no. 1, pp. 126–133, 2004.

[55] S. Böcker, "Sequencing from compomers: Using mass spectrometry for dna de novo sequencing of 200+ nt." *Journal Of Computational Biology*, vol. 11, no. 6, pp. 1110–1134, 2004.

[56] ——, "Snp and mutation discovery using base-specific cleavage and maldi-tof mass spectrometry." *Bioinformatics*, vol. 19, no. Suppl 2, pp. i44–i53, 2003.

[57] P. W, K. KO, F. T, S. Y, and K. M., "Genotools snp manager: a new software for automated high-throughput maldi-tof mass spectrometry snp genotyping." *Biotechniques*, vol. 30, pp. 210–215, 2001.

[58] L. Ganova-Raeva, S. Ramachandran, C. Honisch, J. C. Forbi, X. Zhai, and Y. Khudyakov, "Robust hepatitis b virus genotyping by mass spectrometry." *J. Clin. Microbiol.*, vol. 48:4161, no. 11, 2010.

[59] L. Ganova-Raeva, Z. Dimitrova, D. Campo, L. Yulin, S. Ramachandran, G.-L. Xia, C. Honisch, C. Cantor, and Y. Khudyakov, "Detection of hepatitis c virus transmission using dna mass spectrometry." *J Infect Dis.*, vol. Jan 31, 2013.

[60] S. R, H. TA, M. C, L. F, B. LB, E. MW, H. SA, and E. DJ, "Rapid identification of emerging infectious agents using pcr and electrospray ionization mass spectrometry." *Ann. NY Acad. Sci.*, vol. 1102, pp. 109–120, 2007.

[61] von Wintzingerode F, B. S, S. C, C. NH, S. N, J. C, C. CR, G. UB, and van den Boom D, "High-throughput maldi-tof discovery of genomic sequence polymorphisms." *Genome Res.*, vol. 14, no. 1, pp. 126–133, 2004.

[62] K. F, N. E, L. LK, K. K, R. P, and H. F, "Dna sequence analysis by maldi mass spectrometry." *Nucleic acids research*, vol. 26, pp. 2554–2559, 1998.

[63] T. J, S. P, S. M, B. K, and G. IG, "Analysis and accurate quantification of cpg methylation by maldi mass spectrometry." *Nucleic Acids Res*, vol. 31:e50, 2003.

[64] R. JC and V. KJ, "Simultaneous detection of two bacterial pathogens using bacteriophage amplification coupled with matrix-assisted laser desorption/ionization time-of-flight mass spectrometry." *Rapid Commun Mass Spectrom*, vol. 19, pp. 2757–2761, 2005.

[65] S. MI, D. J, and C. J, "Multiplex detection of human herpesviruses from archival specimens by using matrix-assisted laser desorption ionization-time of flight mass spectrometry." *J Clin Microbiol*, vol. 46, pp. 540–545, 2008.

[66] Y. H, Y. K, K. A, T. Y, C. TE, O. AW, L. R, O. PA, L. W, K. HP, K. AO, M. A, K. H, and K. DM, "Sensitive detection of human papillomavirus in cervical, head/neck, and schistosomiasis-associated bladder malignancies." *Proc. Natl. Acad. Sci.*, vol. 10, pp. 1177–1184, 2013.

[67] S. Böcker and H.-M. Kaltenbach, "Mass spectra alignments and their significance." *Journal of Discrete Algorithms*, vol. 5, no. 4, pp. 714–728, 2007.

[68] V. Mäkinen, "Peak alignment using restricted edit distances." *Biomolecular Engineering*, vol. 24, no. 3, pp. 337–342, 2007.

[69] L. S., D. D., A. M., C. F., B. M. *et al.*, "Combinatorial pooling enables selective sequencing of the barley gene space," *PLoS Comput Biol*, vol. 9, no. 4, 2013.

[70] D. Duma, M. Wootters, A. C. Gilbert, H. Q. Ngo, A. Rudra, M. Alpert, T. J. Close, G. Ciardo, and S. Lonardi, "Accurate decoding of pooled sequenced data using compressed sensing," *Lecture Notes in Computer Science*, vol. 8126, pp. 70–84, 2013.

[71] S. Alon, F. Vigneault, S. Eminaga *et al.*, "Barcoding bias in high-throughput multiplex sequencing of mirna," *Genome Research*, vol. 21, no. 9, p. 1506?1511, 2011.

[72] R. Dorfman, "The detection of defective members of large population," *The Annals of Mathematical Statistics*, vol. 14, no. 4, pp. 436–440, Dec 1943.

[73] D.-Z. Du and F. K. Hwang, *Pooling Design and Nonadaptive Group Testing: Important Tools for DNA Sequencing*. World Scientific Publishing Company, 2006, vol. 18, series on Applied Mathematics.

[74] W. Weili, Y. Huang, X. Huang, and Y. Li, "On error-tolerant dna screening," *Discrete Applied Mathematics*, vol. 154, pp. 1753–1758, 2006.

[75] W. Wu, Y. Li, C.-H. Huang, and D.-Z. Du, *Data Mining in Biomedicine*. Springer Optimization and Its Applications, 2007, vol. 7, ch. Molecular Biology and Pooling Design, pp. 133–139.

[76] P. Berman, B. DasGupta, and M.-Y. Kao, "Tight approximability results for test set problems in bioinformatics," in *Algorithm Theory - SWAT 2004*, ser. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 2004, vol. 3111, pp. 39–50.

[77] S. Prabhu and I. Pe'er, "Overlapping pools for high-throughput targeted resequencing," *Genome Res*, vol. 19, no. 7, pp. 1254–1261, 2009.

[78] D. He, N. Zaitlen, B. Pasaniuc, E. Eskin, and E. Halperin, "Genotyping common and rare variation using overlapping pool sequencing," in *Proceedings of the First Annual RECOMB Satellite Workshop on Massively Parallel Sequencing (RECOMB-seq)*, vol. 12, no. Suppl 6, 2011, bMC Bioinformatics.

[79] E. Y., C. K., G. A., R. R., N. O., R. M., and H. G.J., "Dna sudoku - harnessing high-throughput sequencing for multiplexing specimen analysis," *Genome Res*, vol. 19, pp. 1243–1253, 2009.

[80] N. Shental, A. Amir, and O. Zuk, "Identification of rare alleles and their carriers using compressed se(que)nsing," *Nucleic Acids Res*, vol. 38, pp. 1–22, 2010.

[81] D. Golan, Y. Erlich, and S. Rosset, "Weighted pooling-practical and cost-effective techniques for pooled high-throughput sequencing," *Bioinformatics*, vol. 28, no. 12, pp. i197–i206, 2012.

[82] V. Bansal, "A statistical method for the detection of variants from next-generation resequencing of dna pools," *Bioinformatics*, vol. 26, no. 12, pp. i318–i324, 2010.

[83] M. Kuroda, H. Katano, N. Nakajima, M. Tobiume, A. Ainai, T. Sekizuka, H. Hasegawa, M. Tashiro, Y. Sasaki, Y. Arakawa, and othes, "Characterization of quasispecies of pandemic 2009 influenza a virus (a/h1n1/2009) by de novo sequencing using a next-generation dna sequencer," *PLoS One*, vol. 5, no. 4, p. e10256, 2010.

[84] G. J. Hughes, E. Fearnhill, D. Dunn, S. J. Lycett, A. Rambaut, and A. J. L. Brown, "Molecular phylodynamics of the heterosexual hiv epidemic in the united kingdom," *PLoS pathogens*, vol. 5, no. 9, p. e1000590, 2009.

[85] R. D. Kouyos, V. Von Wyl, S. Yerly, J. Böni, P. Taffé, C. Shah, P. Börgisser, T. Klimkait, R. Weber, and B. Hirschel, "Molecular epidemiology reveals long-term changes in hiv type 1 subtype b transmission in switzerland," *Journal of Infectious Diseases*, vol. 201, no. 10, pp. 1488–1497, 2010.

[86] M. K. Grabowski and A. D. Redd, "Molecular tools for studying hiv transmission in sexual networks," *Current Opinion in HIV and AIDS*, vol. 9, no. 2, pp. 126–133, 2014.

[87] J. O. Wertheim, A. J. L. Brown, N. L. Hepler, , and S. L. K. Pond, "The global transmission network of hiv-1," *Journal of Infectious Diseases*, vol. 209, no. 2, pp. 304–313, 2014.

[88] G. Vaughan, G. Xia, J. C. Forbi, M. A. Purdy, L. M. G. Rossi, P. R. Spradling, and Y. E. Khudyakov, "Genetic relatedness among hepatitis a virus strains associated with food-borne outbreaks," *PloS one*, vol. 8, no. 11, p. e74546, 2013.

[89] M. G. Collier, Y. E. Khudyakov, D. Selvage, M. Adams-Cameron, E. Epson, A. Cronquist, R. H. Jervis, K. Lamba, A. C. Kimura, and R. Sowadsky, "Outbreak of hepatitis a in the usa associated with frozen pomegranate arils imported from turkey: an epidemiological case study," *The Lancet Infectious Diseases*, vol. 14, no. 10, pp. 976–981, 2014.

[90] S. Ramachandran, M. A. Purdy, G.-l. Xia, D. S. Campo, Z. E. Dimitrova, E. H. Teshale, C. G. Teo, and Y. E. Khudyakov, "Recent population expansions of hepatitis b virus in the united states," *Journal of virology*, vol. 88, no. 24, pp. 13 971–13 980, 2014.

[91] A. C. Seña, A. Moorman, L. Njord, R. E. Williams, J. Colborn, Y. Khudyakov, J. Drobenuic, G.-L. Xia, H. Wood, and Z. Moore, "Acute hepatitis b outbreaks in 2 skilled nursing facilities and possible sources of transmission north carolina, 2009–2010," *Infection Control*, vol. 34, no. 07, pp. 709–716, 2013.

[92] M. Holodniy, G. Oda, P. L. Schirmer, C. A. Lucero, Y. E. Khudyakov, G. Xia, Y. Lin, R. Valdiserri, W. E. Duncan, and V. J. Davey, "Results from a large-scale epidemiologic look-back investigation of improperly reprocessed endoscopy equipment," *Infection Control*, vol. 33, no. 07, pp. 649–656, 2012.

[93] A. E. Warner, M. K. Schaefer, P. R. Patel, J. Drobeniuc, G. Xia, Y. Lin, Y. Khudyakov, C. W. Vonderwahl, L. Miller, and N. D. Thompson, "Outbreak of hepatitis c virus infection associated with narcotics diversion by an hepatitis c virus–infected surgical technician," *American journal of infection control*, vol. 43, no. 1, pp. 53–58, 2015.

[94] W. C. Hellinger, L. P. Bacalis, R. S. Kay, N. D. Thompson, G.-L. Xia, Y. Lin, Y. E. Khudyakov, and J. F. Perz, "Health care–associated hepatitis c virus infections attributed to narcotic diversion," *Annals of internal medicine*, vol. 156, no. 7, pp. 477–482, 2012.

[95] G. E. Fischer, M. K. Schaefer, B. J. Labus, L. Sands, P. Rowley, I. A. Azzam, P. Armour, Y. E. Khudyakov, Y. Lin, and G. Xia, "Hepatitis c virus infections from unsafe injection practices at an endoscopy clinic in las vegas, nevada, 2007–2008," *Clinical infectious diseases*, vol. 51, no. 3, pp. 267–273, 2010.

[96] A. Apostolou, M. L. Bartholomew, R. Greeley, S. M. Guilfoyle, M. Gordon, C. Genese, J. P. Davis, B. Montana, and G. Borlaug, "Transmission of hepatitis c virus associated with surgical procedures-new jersey 2010 and wisconsin 2011." *MMWR. Morbidity and mortality weekly report*, vol. 64, no. 7, pp. 165–170, 2015.

[97] P. Skums, L. Bunimovich, and Y. Khudyakov, "Antigenic cooperation among intrahost hcv variants organized into a complex network of cross-immunoreactivity," *Proceedings of the National Academy of Sciences*, p. 201422942, 2015.

[98] D. S. Campo, Z. Dimitrova, L. Yamasaki, P. Skums, D. T. Lau, G. Vaughan, J. C. Forbi, C.-G. Teo, and Y. Khudyakov, "Next-generation sequencing reveals large connected networks of intra-host hcv variants," *BMC genomics*, vol. 15, no. Suppl 5, p. S4, 2014.

[99] C. Trapnell, D. G. Hendrickson, M. Sauvageau, L. Goff, J. L. Rinn, and L. Pachter, "Differential analysis of gene regulation at transcript resolution with rna-seq," *Nature biotechnology*, vol. 31, no. 1, pp. 46–53, 2012.

[100] J. C. Alwine, D. J. Kemp, and G. R. Stark, "Method for detection of specific rnas in agarose gels by transfer to diazobenzyloxymethyl-paper and hybridization with dna probes," *Proceedings of the National Academy of Sciences*, vol. 74, no. 12, pp. 5350–5354, 1977.

[101] M. Schena, D. Shalon, R. W. Davis, and P. O. Brown, "Quantitative monitoring of gene expression patterns with a complementary dna microarray," *Science*, vol. 270, no. 5235, pp. 467–470, 1995.

[102] A. M. Wang, M. V. Doyle, and D. F. Mark, "Quantitation of mrna by the polymerase chain reaction," *Proceedings of the National Academy of Sciences*, vol. 86, no. 24, pp. 9717–9721, 1989.

[103] M. A. Moran, "Metatranscriptomics: eavesdropping on complex microbial communities," *Issues*, 2010.

[104] C. Trapnell, L. Pachter, and S. Salzberg, "TopHat: discovering splice junctions with RNA-Seq." *Bioinformatics*, vol. 25, no. 9, pp. 1105–1111, 2009. [Online]. Available: http://dx.doi.org/10.1093/bioinformatics/btp120

[105] K. F. Au, H. Jiang, L. Lin, Y. Xing, and W. H. Wong, "Detection of splice junctions from paired-end rna-seq data by splicemap," *Nucleic Acids Research*, 2010. [Online]. Available: http://nar.oxfordjournals.org/content/early/2010/04/05/nar.gkq211.abstract

[106] M. Garber, M. G. Grabherr, M. Guttman, and C. Trapnell, "Computational methods for transcriptome annotation and quantification using RNA-seq," *Nature Methods*, vol. 8, no. 6, pp. 469–477, May 2011. [Online]. Available: http://dx.doi.org/10.1038/nmeth.1613

[107] M. Grabherr, "Full-length transcriptome assembly from rna-seq data without a reference genome." *Nature biotechnology*, vol. 29, no. 7, pp. 644–652, 2011. [Online]. Available: http://dx.doi.org/10.1038/nbt.1883

[108] G. Robertson, J. Schein, R. Chiu, R. Corbett, M. Field, S. D. Jackman, K. Mungall, S. Lee, H. M. Okada, J. Q. Qian, and et al., "De novo assembly and analysis of rna-seq data." *Nature Methods*, vol. 7, no. 11, pp. 909–912, 2010. [Online]. Available: http://www.ncbi.nlm.nih.gov/pubmed/20935650

[109] P. A. Pevzner, "1-Tuple DNA sequencing: computer analysis." *J Biomol Struct Dyn*, vol. 7, no. 1, pp. 63–73, Aug. 1989.

[110] A. Roberts, H. Pimentel, C. Trapnell, and L. Pachter, "Identification of novel transcripts in annotated genomes using rna-seq," *Bioinformatics*, 2011. [Online]. Available: http://bioinformatics.oxfordjournals.org/content/early/2011/06/21/bioinformatics.btr355.abstract

[111] A. Roberts, C. Trapnell, J. Donaghey, J. Rinn, and L. Pachter, "Improving rna-seq expression estimates by correcting for fragment bias," *Genome Biology*, vol. 12, no. 3, p. R22, 2011.

[112] R. Tibshirani, "Regression shrinkage and selection via the lasso," *Journal of Royal Statistical Society*, vol. 58, pp. 267–288, 1996.

[113] Y. Y. Lin, P. Dao, F. Hach, M. Bakhshi, F. Mo, A. Lapuk, C. Collins, and S. C. Sahinalp, "Cliiq: Accurate comparative detection and quantification of expressed isoforms in a population," *Proc. 12th Workshop on Algorithms in Bioinformatics*, 2012.

[114] S. Mangul, A. Caciula, I. Mandoiu, and A. Zelikovsky, "Rna-seq based discovery and reconstruction of unannotated transcripts in partially annotated genomes," in *Bioinformatics and Biomedicine Workshops (BIBMW), 2011 IEEE International Conference on*, nov. 2011, pp. 118 –123.

[115] UCSC Genome Database, http://genome.ucsc.edu.

[116] CCDS Genome Database, http://www.ncbi.nlm.nih.gov/projects/CCDS/CcdsBrowse.cgi.

[117] D. Bentley, S. Balasubramanian, H. Swerdlow, G. Smith, J. Milton, C. Brown, K. Hall, D. Evers, C. Barnes, H. Bignell *et al.*, "Accurate whole human genome sequencing using reversible terminator chemistry," *Nature*, vol. 456, no. 7218, pp. 53–59, 2008.

[118] J. M. Rothberg, W. Hinz, T. M. Rearick, J. Schultz, W. Mileski, M. Davey, J. H. Leamon, K. Johnson, M. J. Milgrew, M. Edwards, and et al., "An integrated semiconductor device enabling non-optical genome sequencing."

Nature, vol. 475, no. 7356, pp. 348–352, 2011. [Online]. Available: http://www.nature.com/doifinder/10.1038/nature10242

[119] B. Li, V. Ruotti, R. Stewart, J. Thomson, and C. Dewey, "RNA-Seq gene expression estimation with read mapping uncertainty," *Bioinformatics*, vol. 26, no. 4, pp. 493–500, 2010. [Online]. Available: http://dx.doi.org/10.1093/bioinformatics/btp692

[120] B. Langmead, C. Trapnell, M. Pop, and S. Salzberg, "Ultrafast and memory-efficient alignment of short DNA sequences to the human genome," *Genome Biology*, vol. 10, no. 3, p. R25, 2009. [Online]. Available: http://genomebiology.com/2009/10/3/R25

[121] W. Li, J. Feng, and T. Jiang, "IsoLasso: A LASSO Regression Approach to RNA-Seq Based Transcriptome Assembly," *Lecture Notes in Computer Science*, vol. 6577, pp. 168–+, 2011.

[122] S. Pal, R. Gupta, H. Kim, P. Wickramasinghe, V. Baubet, L. C. Showe, N. Dahmane, and R. V. Davuluri, "Alternative transcription exceeds alternative splicing in generating the transcriptome diversity of cerebellar development," *Genome Research*, 2011. [Online]. Available: http://genome.cshlp.org/content/early/2011/06/28/gr.120535.111.abstract

[123] IBM, "Inc: IBM ILOG CPLEX 12.1." http://www.ibm.com/software/integration/optimization/cplex/, 2009.

[124] B. Paşaniuc, N. Zaitlen, and E. Halperin, "Accurate estimation of expression levels of homologous genes in RNA-seq experiments," in *Proc. 14th Annual Intl. Conf. on Research in Computational Molecular Biology (RECOMB)*, ser. Lecture Notes in Computer Science, B. Berger, Ed., vol. 6044. Springer Berlin / Heidelberg, 2010, pp. 397–409.

[125] T. Steijger, J. F. Abril, P. G. EngstrÃűm, F. Kokocinski, T. J. H. The RGASP Consor-

tium, R. GuigÃş, J. Harrow, and P. Bertone, "Assessment of transcript reconstruction methods for RNA-Seq." *Nature Methods*, vol. 10, pp. 1177–1184, 2013.

[126] B. Li and C. Dewey, "Rsem: accurate transcript quantification from rna-seq data with or without a reference genome," *BMC bioinformatics*, vol. 12, no. 1, p. 323, 2011.

[127] W. Li and T. Jiang, "Transcriptome assembly and isoform expression level estimation from biased RNA-Seq reads." *Bioinformatics*, vol. 28, no. 22, pp. 2914–2921, 2012.

[128] A. I. Tomescu, A. Kuosmanen, R. Rizzi, and V. MÃd'kinen, "A novel min-cost flow method for estimating transcript expression with rna-seq," in *Proc. RECOMB-seq 2013*, 2013.

[129] W. Li, J. Feng, and T. Jiang, "IsoLasso: A LASSO Regression Approach to RNA-Seq Based Transcriptome Assembly," *Journal of Computational Biology*, vol. 18, pp. 1693–1707, 2011.

[130] A. I. Tomescu, A. Kuosmanen, R. Rizzi, and V. Mäkinen, "A novel min-cost flow method for estimating transcript expression with rna-seq," *BMC Bioinformatics*, vol. 14, no. S-5, p. S15, 2013.

[131] H. Jiang and W. Wong, "Statistical inferences for isoform expression in RNA-Seq," *Bioinformatics*, vol. 25, no. 8, pp. 1026–1032, 2009. [Online]. Available: http://dx.doi.org/10.1093/bioinformatics/btp113

[132] F. Rapaport, R. Khanin, Y. Liang, A. Krek, P. Zumbo, C. E. Mason, N. D. Socci, and D. Betel, "Comprehensive evaluation of differential expression analysis methods for rna-seq data," *arXiv preprint arXiv:1301.5277*, 2013.

[133] J. Li, H. Jiang, and W. Wong, "Method modeling non-uniformity in short-read rates in rna-seq data," *Genome Biol*, vol. 11, no. 5, p. R25, 2010.

[134] K. D. Hansen, S. E. Brenner, and S. Dudoit, "Biases in illumina transcriptome sequencing caused by random hexamer priming," *Nucleic acids research*, vol. 38, no. 12, pp. e131–e131, 2010.

[135] A. Roberts, C. Trapnell, J. Donaghey, J. L. Rinn, and L. Pachter, "Improving rna-seq expression estimates by correcting for fragment bias," *Genome biology*, vol. 12, no. 3, p. R22, 2011.

[136] F. E. Angly, D. Willner, F. Rohwer, P. Hugenholtz, and G. W. Tyson, "Grinder: a versatile amplicon and shotgun sequence simulator," *Nucleic Acids Research*, vol. 40, no. 12, p. e94, 2012. [Online]. Available: http://nar.oxfordjournals.org/content/40/12/e94.abstract

[137] MAQC Consortium, "The Microarray Quality Control (MAQC) project shows inter- and intraplatform reproducibility of gene expression measurements." *Nature Biotechnology*, vol. 24, no. 9, pp. 1151–1161, Sep. 2006.

[138] K. MM, "Digital multiplexed gene expression analysis using the nanostring ncounter system." *Curr Protoc Mol Biol.*, vol. 94, 2011.

[139] Z. Dimitrova, D. Campo, S. Ramachandran, G. Vaughan, L. Ganova-Raeva, Y. Lin, J. Forbi, G. Xia, P. Skums, B. Pearlman, and Y. Khudyakov, "valuation of viral heterogeneity using next-generation sequencing, end-point limiting-dilution and mass spectrometry." *In Silico Biology*, vol. 11, pp. 183–192, 2011/2012.

[140] P. Pevzner, V. Dancik, and C. Tang, "Mutation-tolerant protein identification by mass-spectrometry." *Journal of Computational Biology*, vol. 7, pp. 777–787, 2000.

[141] P. Skums, D. Campo, Z. Dimitrova, G. Vaughan, D. Lau, and Y. Khudyakov, "Numerical detection, measuring and analysis of differential interferon resistance for individual hcv intra-host variants and its influence on the therapy response," *In Silico Biol*, vol. 11, no. 5, pp. 263–269, Jan 2012.

[142] K. J. Metzner *et al.*, "Minority quasispecies of drug-resistant hiv-1 that lead to early therapy failure in treatment-naive and -adherent patients," *Clin Infect Dis*, vol. 48, no. 2, pp. 239–247, 2009.

[143] P. Skums, O. Glebova, A. Zelikovsky, I. Mandoiu, and Y. Khudyakov, "Optimizing pooling strategies for the massive next-generation sequencing of viral samples," in *3rd Workshop on Computational Advances for Next Generation Sequencing (CANGS)*, 2013.

[144] G. M.R. and J. D.S., *Computers and Intractability. A Guide to the Theory of NP-completeness*. San Francisco, CA, 1979.

[145] D. Zuckerman, "Linear degree extractors and the inapproximability of max clique and chromatic number," *Proc. 38th ACM Symp. Theory of Computing*, pp. 681–690, 2006.

[146] F. Glover and G. Kochenberger, *Handbook of Metaheuristics*. Kluwer Academic Publishers, 2003.

[147] A. Artyomenko, N. Mancuso, P. Skums, I. Mandoiu, and A. Zelikovsky, "kGEM: An em-based algorithm for local reconstruction of viral quasispecies," in *2013 IEEE 3rd International Conference onComputational Advances in Bio and Medical Sciences (ICCABS)*, 2013.

[148] L. J., T. J.E., D. MJ, a. Y. H. Lee W.M, P. B.L., V. G., F. J.C., X. G.L., and K. Y.E., "Coordinated evolution among hepatitis c virus genomic sites is coupled to host factors and resistance to interferon," *In Silico Biol.*, vol. 11, no. 5-6, pp. 213–224, 2011-2012.

[149] D. S. Campo, Z. Dimitrova, L. Yamasaki, P. Skums, D. Lau, G. Vaughan, J. Forbi, C.-G. Teo, and Y. Khudyakov, "Next-generation sequencing reveals large connected networks of intra-host hcv variants," accepted to BMC Genomics.

[150] P. Skums, Z. Dimitrova, D. S. Campo, G. Vaughan, L. Rossi, J. C. Forbi, J. Yokosawa, A. Zelikovsky, and Y. Khudyakov, "Efficient error correction for next-generation sequencing of viral amplicons," *BMC Bioinformatics*, vol. 13, no. Suppl 10:S6, 2012.

[151] R. Edgar, "Muscle: multiple sequence alignment with high accuracy and high throughput," *Nucleic Acids Res*, vol. 32, no. 5, pp. 1792–1797, 2004.

[152] R. A. Bull, F. Luciani, K. McElroy, S. Gaudieri, S. T. Pham, A. Chopra, B. Cameron, L. Maher, G. J. Dore, P. A. White, and A. R. Lloyd, "Sequential bottlenecks drive viral evolution in early acute hepatitis c virus infection," *PLoS Pathogens*, vol. 7, no. 9, Sept 2011.

[153] K. C. Cortes, O. Zagordi, T. Laskus, R. Ploski, I. Bukowska-Osko, A. Pawelczyk, H. Berak, and M. Radkowski, "Ultradeep pyrosequencing of hepatitis c virus hypervariable region 1 in quasispecies analysis," *BioMed Research International*, 2013.

[154] G. J., E. J.I., C. M., D. Garcia-Cehic, P. C., C. R. *et al.*, "Ultra-deep pyrosequencing (udps) data treatment to study amplicon hcv minor variants," *PLoS ONE*, vol. 8, no. 12, 2013.

[155] L. J., "Divergence measures based on the shannon entropy," *IEEE T Inform Theory*, vol. 37, p. 145?151, 1991.

[156] J. W. Drake and J. J. Holland, "Mutation rates among rna viruses," *Proceedings of the National Academy of Sciences*, vol. 96, no. 24, pp. 13 910–13 913, 1999.

[157] E. Domingo and J. Holland, "Rna virus mutations and fitness for survival," *Annual Reviews in Microbiology*, vol. 51, no. 1, pp. 151–178, 1997.

[158] E. Domingo, J. Sheldon, and C. Perales, "Viral quasispecies evolution," *Microbiology and Molecular Biology Reviews*, vol. 76, no. 2, pp. 159–216, 2012.

[159] N. Eriksson, L. Pachter, Y. Mitsuya, S.-Y. Rhee, C. Wang, B. Gharizadeh, M. Ronaghi, R. W. Shafer, and N. Beerenwinkel, "Viral population estimation using pyrosequencing," *PLoS computational biology*, vol. 4, no. 5, p. e1000074, 2008.

[160] J. Archer, M. S. Braverman, B. E. Taillon, B. Desany, I. James, P. R. Harrigan, M. Lewis, and D. L. Robertson, "Detection of low-frequency pretherapy chemokine (cxc motif) receptor 4-using hiv-1 with ultra-deep pyrosequencing," *AIDS (London, England)*, vol. 23, no. 10, p. 1209, 2009.

[161] C. Hoffmann, N. Minkah, J. Leipzig, G. Wang, M. Q. Arens, P. Tebas, and F. D. Bushman, "Dna bar coding and pyrosequencing to identify rare hiv drug resistance mutations," *Nucleic acids research*, vol. 35, no. 13, p. e91, 2007.

[162] W. Wang, X. Zhang, Y. Xu, G. M. Weinstock, A. M. Di Bisceglie, and X. Fan, "High-resolution quantification of hepatitis c virus genome-wide mutation load and its correlation with the outcome of peginterferon-alpha2a and ribavirin combination therapy," *PloS one*, vol. 9, no. 6, p. e100131, 2014.

[163] P. Skums, D. S. Campo, Z. Dimitrova, G. Vaughan, D. T. Lau, and Y. Khudyakov, "Numerical detection, measuring and analysis of differential interferon resistance for individual hcv intra-host variants and its influence on the therapy response," *In silico biology*, vol. 11, no. 5, pp. 263–269, 2011.

[164] D. S. Campo, P. Skums, Z. Dimitrova, G. Vaughan, J. C. Forbi, C.-G. Teo, Y. Khudyakov, and D. T. Lau, "Drug resistance of a viral population and its individual intrahost variants during the first 48 hours of therapy," *Clinical Pharmacology & Therapeutics*, vol. 95, no. 6, pp. 627–635, 2014.

[165] P. Skums, A. Artyomenko, O. Glebova, S. Ramachandran, I. Mandoiu, D. S. Campo, Z. Dimitrova, A. Zelikovsky, and Y. Khudyakov, "Computational framework for next-generation sequencing of heterogeneous viral populations using combinatorial pooling," *Bioinformatics*, vol. 31, no. 5, pp. 682–690, 2015.

[Online]. Available: http://bioinformatics.oxfordjournals.org/content/31/5/682.abstract

[166] I. V. Astrakhantseva, D. S. Campo, A. Araujo, C.-G. Teo, Y. Khudyakov, and S. Kamili, "Differences in variability of hypervariable region 1 of hepatitis c virus (hcv) between acute and chronic stages of hcv infection," *In silico biology*, vol. 11, no. 5, pp. 163–173, 2011.

[167] I. Williams, "Epidemiology of hepatitis c in the united states," *The American journal of medicine*, vol. 107, no. 6, pp. 2–9, 1999.

[168] S. Ramachandran, G.-l. Xia, L. M. Ganova-Raeva, O. V. Nainan, and Y. Khudyakov, "End-point limiting-dilution real-time pcr assay for evaluation of hepatitis c virus quasispecies in serum: performance under optimal and suboptimal conditions," *Journal of virological methods*, vol. 151, no. 2, pp. 217–224, 2008.

[169] S. Ramachandran, X. Zhai, H. Thai, D. S. Campo, G. Xia, L. M. Ganova-Raeva, J. Drobeniuc, and Y. E. Khudyakov, "Evaluation of intra-host variants of the entire hepatitis b virus genome," *PloS one*, vol. 6, no. 9, p. e25232, 2011.

[170] K. Katoh and D. M. Standley, "Mafft multiple sequence alignment software version 7: improvements in performance and usability," *Molecular biology and evolution*, vol. 30, no. 4, pp. 772–780, 2013.

[171] D. L. Wallace, "Comment," *Journal of the American Statistical Association*, vol. 78, no. 383, pp. 569–576, 1983.