EMERGING TOPICS IN GENOME SEQUENCING AND ANALYSIS

A Dissertation

by

CHUN-CHI CHEN

Submitted to the Office of Graduate and Professional Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

Chair of Committee,	Byung-Jun Yoon
Co-Chair of Committee,	Xiaoning Qian
Committee Members,	Edward Dougherty
	P. R. Kumar
	Won-Bo Shim
Head of Department,	Miroslav M. Begovic

December 2017

Major Subject: Electrical Engineering

Copyright 2017 Chun-Chi Chen

ABSTRACT

This dissertation studies the emerging topics in genome sequencing and analysis with DNA and RNA. The optimal hybrid sequencing and assembly for accurate genome reconstruction and efficient detection approaches for novel ncRNAs in genomes are discussed.

The next-generation sequencing is a significant topic that provides whole genetic information for the further biological research. Recent advances in high-throughput genome sequencing technologies have enabled the systematic study of various genomes by making whole genome sequencing affordable. To date, many hybrid genome assembly algorithms have been developed that can take reads from multiple read sources to reconstruct the original genome. An important aspect of hybrid sequencing and assembly is that the feasibility conditions for genome reconstruction can be satisfied by different combinations of the available read sources, opening up the possibility of optimally combining the sources to minimize the sequencing cost while ensuring accurate genome reconstruction. In this study, we derive the conditions for whole genome reconstruction from multiple read sources at a given confidence level and also introduce the optimal strategy for combining reads from different sources to minimize the overall sequencing cost. We show that the optimal read set, which simultaneously satisfies the feasibility conditions for genome reconstruction and minimizes the sequencing cost, can be effectively predicted through constrained discrete optimization.

The availability of genome-wide sequences for a variety of species provides a large database for the further RNA analysis with computational methods. Recent studies have shown that noncoding RNAs (ncRNAs) are known to play crucial roles in various

biological processes, and some ncRNAs are related to the genome stability and a variety of inherited diseases. The discovery of novel ncRNAs is hence an important topic, and there is a pressing need for accurate computational detection approaches that can be used to efficiently detect novel ncRNAs in genomes. One important issue is RNA structure alignment for comparative genome analysis, as RNA secondary structures are better conserved than the RNA sequences. Simultaneous RNA alignment and folding algorithms aim to accurately align RNAs by predicting the consensus structure and alignment at the same time, but the computational complexity of the optimal dynamic programming algorithm for simultaneous alignment and folding is extremely high. In this work, we proposed an innovative method, *TOPAS*, for RNA structural alignment that can efficiently align RNAs through topological networks. Although many ncRNAs are known to have a well conserved secondary structure, which provides useful clues for computational prediction, the prediction of ncRNAs is still challenging, since it has been shown that a structure-based approach alone may not be sufficient for detecting ncRNAs in a single sequence. In this study, we first develop a new approach by utilizing the n-gram model to classify the sequences and extract effective features to capture sequence homology. Based on this approach, we propose an advanced method, *piRNAdetect*, for reliable computational prediction of piRNAs in genome sequences. Utilizing the n-gram model can enhance the detection of ncRNAs that have sparse folding structures with many unpaired bases. By incorporating the n-gram model with the generalized ensemble defect, which assesses structure conservation and conformation to the consensus structure, we further propose *RNAdetect*, a novel computational method for accurate detection of ncRNAs through comparative genome analysis. Extensive performance evaluation based on the Rfam database and bacterial genomes demonstrates that our approaches can accurately and reliably detect novel ncRNAs, outperforming the current advanced methods.

ACKNOWLEDGEMENTS

I would like to express my thanks to all people who contributed to this study. Without their assistance, it would not have been possible for me to finish my Ph.D. degree.

First and foremost, I would like to express my sincere gratitude to my advisor Dr. Byung-Jun Yoon and co-advisor Dr. Xiaoning Qian for their guidance and support throughout the research. My sincere appreciation is extended to my committee members: Dr. Edward Dougherty, Dr. P. R. Kumar and Dr. Won-Bo Shim for their advice to improve the research.

Last but not least, I would like to thank my family and friends for their encouragement and support during the research.

CONTRIBUTORS AND FUNDING SOURCES

Contributors

This work was supported by a dissertation committee consisting of Professor Byung-Jun Yoon, Professor Xiaoning Qian, Professor Edward Dougherty, and Professor P. R. Kumar of the Department of Electrical and Computer Engineering and Professor Won-Bo Shim of the Department of Plant Pathology and Microbiology.

The methods and data were developed and analyzed by Professor Byung-Jun Yoon, Professor Xiaoning Qian, and the student. All of this work was improved by the dissertation committee members, and all other work conducted for the dissertation was completed by the student independently.

Funding Sources

This work was supported by the NSF Awards CCF-1149544 and CCF- 1447235 from the National Science Foundation, the Agriculture and Food Research Initiative Competitive Grants Program (Grant number 2013- 68004-20359 and 06-505570-01006) from the USDA National Institute of Food and Agriculture, and the TEES-AgriLife Center for Bioinformatics and Genomic Systems Engineering (CBGSE).

NOMENCLATURE

The mathematics notations in the first section.

G	Length of the target genome sequence
L_i	Length of the read from the i-th source
N_i	Number of reads from the i-th source
$\mathbf{L}$	Set of overall read lengths $\{L_i\}$
$\mathbf{N}$	Set of overall read numbers $\{N_i\}$
$\overline{L}$	Average read length over available sequencing sources
N	Number of total reads
K	Length of valid overlap in bps (base pairs)
C	Coverage depth $N\overline{L}/G$
θ	Normalized valid overlap $K/\overline{L}$

The mathematics notations in the second section.

G_n	Graph of the n-th topological network
V_n	Set of nodes in the n-th graph
E_n	Set of weighted edges in the n-th graph
N	Length of RNA sequence
R	Topological similarity
R_S	Structural similarity
R_C	Connected similarity
R_E	Sequence similarity

The mathematics notations in the third section.

L	Length of the sequence
R	Homologous likelihood
S	Maximum homologous likelihood for a sequence
Z	Z-score for the similarity measure
n	Size of n-gram model

The mathematics notations in the fourth section.

Ω	Structure ensemble of the RNA sequence
S	Structure matrix of the RNA structure
N	Length of the RNA sequence
E_i	MFE of the i-th sequence
E_{single}	Average MFE for the sequences in the alignment
E_{cons}	MFE for the sequence alignment
P_A	Probability of the alignment between two RNA sequence
P_S	Probability of the RNA structure
P_{cons}	Consensus structure score
R	Homologous likelihood
Z	Z-score for the features
d	Distance between two RNA secondary structures
n	Ensemble defect of the RNA structure

TABLE OF CONTENTS

Page

LIST OF FIGURES

FIGURE Page

LIST OF TABLES

TABLE Page

1. INTRODUCTION

1.1 Background

In 1944, Oswald Avery and his colleagues published experimental evidence revealing that deoxyribonucleic acid (DNA) is the carrier of genetic information [2]. The study shed light on our understanding of genetics and inspired the discovery of genetic code. Later in 1958, the central dogma of molecular biology was first stated by Francis Crick [3]. The central dogma describes genetic information is transcribed from DNA to RNA, and then coding RNA is translated to proteins. In this scenario, RNA has been regarded as an intermediary in the gene expression until the discovery of functional noncoding RNAs in the 1980s [4]. Noncoding RNAs were found to be much more abundant than coding RNAs and play crucial roles in diverse cellular processes such as transcriptional and post-transcriptional regulation, chromosome replication, RNA processing and modification, and protein degradation and translocation [5–7]. It is now known that housekeeping RNAs, such as transfer RNA (tRNA) and ribosomal RNA (rRNA), perform structural organization and catalytic roles in the translation process. Moreover, several small noncoding RNAs, such as micro RNA (miRNA), Piwi-interacting RNA (piRNA), and short interfering RNA (siRNA), are found to be associated with regulation and suppression in diverse biological processes [8–10]. These ncRNAs play important roles in gene silencing and protecting the genome from invasive transposons [11, 12]. Recent studies have shown that some ncRNAs are linked to the genome stability and a variety of inherited diseases and cancers [13–20]. These findings suggest the clinical importance of ncRNAs, and hence there is a pressing need for effective computational methods that can be used for computational identification of ncRNAs.

The sequencing technology had advanced rapidly since Fred Sanger developed chain-termination sequencing method and sequenced the first DNA genome with 5,375 nucleotides in 1977 [21, 22]. With the advent of the high-throughput technique, it becomes possible to sequence large-scale genomes and transcriptome with reasonable cost. Meanwhile, these genome-wide sequences provide a mine of genetic information, that facilitates the further analysis and ncRNA discovery with computational methods. In this dissertation, we discuss the emerging topics in genome sequencing and analysis with DNA and RNA. First, the optimal hybrid sequencing and assembly for accurate genome reconstruction are discussed in the beginning. Second, the structural RNA alignment is discussed since RNA secondary structures are better conserved and identified. Furthermore, the efficient new detection approaches for novel ncRNAs in genomes are studied.

1.2 Dissertation Outline

This dissertation is organized as follows. In Section 2, the optimal hybrid sequencing and assembly are studied. We derive the conditions for whole genome reconstruction from multiple read sources at a given confidence level and also introduce the optimal strategy for combining reads from different sources to minimize the overall sequencing cost. In Section 3, we address the problem of global structural alignment of pairwise RNA sequences, and propose an innovate method for RNA structural alignment through topological networks. In Section 4, we study computational detection for piRNAs using n-gram models and support vector machine. We develop a new approach by utilizing the n-gram model to classify the sequences and extract effective features to capture sequence homology for the efficient detection. In Section 5, we further study the computational detection for novel noncoding RNAs, and propose a novel computational method for accurate detection of ncRNAs through efficient

comparative genome analysis. Finally, conclusions and future works are summarized in Section 6.

2. OPTIMAL HYBRID SEQUENCING AND ASSEMBLY *

2.1 Introduction

Modern high-throughput shotgun sequencing devices sequence genomes using proprietary techniques to generate a large number of relatively short sequence fragments. Depending on the technology used, the sequence fragments, typically called reads, have different lengths. The desired read length affects the choice of sequencing technology and the overall cost of the sequencing experiments. In genome assembly studies, assembly algorithms go through multiple steps to reconstruct the original genome from the numerous tiny reads, where conditions on minimum read length and coverage need to be met to distinguish repeats and faithfully reconstruct the original genome. At present, there are various high-throughput sequencing platforms [23, 24], where the major commercially available technologies for next-generation sequencing (NGS) include Illumina HiSeq, Roche 454, and Life Technologies SOLiD. Additionally, third generation technologies such as PacBio have emerged, which are based on single-molecule sequencing and generate long reads. Depending on the technology used, different sequencing platforms generate reads of different length and quality at different costs. In general, the cost of generating long reads is substantially higher than that of obtaining short reads, while longer reads make the assembly more accurate, particularly when repeated regions and gaps are present in the genome. It is possible to reduce the average sequencing cost by combining reads with different length and cost from multiple sources obtained through different sequencing technologies. This is referred to as the hybrid assembly, and hybrid assemblers have been developed

*Reprinted with permission from Optimal hybrid sequencing and assembly: Feasibility conditions for accurate genome reconstruction and cost minimization strategy by Chun-Chi Chen, Noushin Ghaffari, Xiaoning Qiana, Byung-Jun Yoon, 2017. Computational Biology and Chemistry, Volume 69, August 2017, Pages 153-163, Copyright 2017 by Elsevier Ltd.

to assemble genome sequences based on reads from multiple sources [25–29], which include widely-used algorithms such as CABOG [25] and ALLPATHS-LG [26].

Although there exist various hybrid assemblers that can assist with genome assembly from multiple read sources, there is still a pressing need for rigorous investigation of the *feasibility* of complete genome reconstruction and the overall *sequencing cost* for such hybrid approaches. In recent years, there have been research efforts to examine the minimum requirements for complete genome reconstruction [30] and to derive a lower bound for the read length and the coverage [31] for the case of genome assembly based on a single read source. The increasing popularity of hybrid assembly, as well as the potential quality improvement and cost reduction that can be attained through the combination of multiple read sources have motivated us to study critical aspects of hybrid assembly in this work. First, we investigate the feasibility conditions to ensure complete genome reconstruction based on multiple read sources. Second, we propose the optimal strategy for combining different read sources to minimize the overall sequencing cost while ensuring the feasibility of complete genome reconstruction. Finally, we present simulation results that verify the feasibility conditions presented in this work and clearly demonstrate that the proposed optimal hybrid sequencing strategy can lead to complete genome reconstruction at the minimum sequencing cost.

2.2 Feasibility Conditions for Accurate Genome Reconstruction and Cost Minimization Strategy

The main research question that we address in this study is how one can identify the *optimal hybrid sequencing strategy* that combines reads from multiple sources obtained through different high-throughput sequencing technologies such that it (i) guarantees the feasibility of accurate whole genome assembly at a given confidence

level (or "target success rate"); and (ii) minimizes the total sequencing cost.

In Section 2.2.1, we first discuss the feasibility of whole genome reconstruction based on multiple read sources, and derive the conditions that can ensure a reliable assembly of error-free reads. Following previous studies on the feasibility of complete genome reconstruction based on a single read source [30–32], our work, which extends the feasibility analysis to multiple read sources, also focuses on the error-free case to investigate the theoretical bounds for complete genome reconstruction. In practice, we note that reads contain errors and there are paired-reads that can be regarded as long reads with erasures. In this work, we simplify the read model and focus on deriving feasible bounds and optimal sequencing strategies for complete genome reconstruction with error-free reads. The derived results can be extended to paired-reads and reads with errors in a relatively straightforward manner by incorporating read error corrections [33–36].

As observed in Motahari et al. [30] and Bresler et al. [31], the assembly feasibility depends on both the read lengths and the genome coverage. We can accurately reconstruct the target genome sequence only by taking reads with proper lengths, and at the same time, only when the sufficient number of such reads are available to reasonably cover the entire genome. Although sequencing technologies that yield longer reads may satisfy the feasibility conditions for a wider variety of genomes, they also tend to incur higher sequencing cost. Consequently, from the perspective of resource (or budget) allocation, it would not be prudent to solely rely on read sources that yield long reads to satisfy the feasibility conditions as such an approach will incur very high sequencing cost to meet the coverage conditions.

In the proposed optimal hybrid sequencing approach, which we present in Section 2.3, we optimally combine multiple read sources to meet the assembly feasibility conditions – comprised of the "coverage condition" and the "bridging conditions" –

where the longer (and more expensive) reads are used to ensure the feasibility of accurate genome assembly while the shorter (and more affordable) reads are used to satisfy the coverage condition. The proposed hybrid sequencing and assembly approach is optimal in the sense that it identifies the best strategy for combining multiple read sources to minimize the total sequencing cost with probabilistic guarantees of the accurate whole genome assembly. This is achieved by formulating a constrained optimization problem based on the derived assembly feasibility conditions to determine the optimal combination that results in the minimum cost.

Finally, in Section 2.3.1, we present an enhanced version of the multi-bridging algorithm that was originally proposed in Bresler et al. [31], which is a genome assembly algorithm based on de Bruijn K-mer graph. We show that the modified algorithm can faithfully reconstruct the whole genome at the desired target success rate when the predicted feasibility conditions are met.

2.2.1 Feasibility of Accurate Whole Genome Reconstruction

We first define the mathematical notations to be used in our feasibility analysis for whole genome hybrid assembly. G denotes the length of the target genome to be reconstructed. L_i denotes the read length from the i-th source and $\mathbf{L} = \{L_i\}$ is defined as the set of all read lengths. Similarly, N_i denotes the number of reads from the i-th source and $\mathbf{N} = \{N_i\}$ is the set of all read numbers for all sources. The total number of reads is denoted by N and we denote the average read length for all available reads as $\overline{L}$. $C = N\overline{L}/G$ denotes the coverage depth. Finally, K is defined as the length of valid overlap in base pairs (bps) and $\theta = K/\overline{L}$ denotes the normalized valid overlap. The value K denotes the minimum overlap that is needed to recognize that consecutive overlapping reads that can be assembled into a contig certainly belong to segments in the target genome sequence. A contig is defined as

a set of overlapping reads that represent an extended segment in the genome. It is typical for most genome assemblers to assemble reads into multiple contigs. However, whole genome reconstruction (and the feasibility thereof) being the main focus of this work, we aim to assemble the given reads into a single contig.

When considering multiple read sources from different sequencing technologies, we will have multiple types of reads at our disposal, each of which has different read length and per-base sequencing cost. An important question we face is how to combine the available read sources and how many reads to draw from each source to ensure accurate whole genome reconstruction at a desired confidence level (i.e., target success rate). In what follows, we aim to address this question. All proofs and mathematical derivations of the propositions presented in this section can be found in the **Appendix**.

2.2.1.1 Feasibility

It is challenging to faithfully reconstruct the original genome from millions of short reads, partly due to the huge amount and the short length of the reads to be assembled, but also due to the large size and the inherent complexity of many genomes. For example, the size of a simple bacterial genome can be several millions of base pairs while the size of eukaryotic genomes can range from 2 million to over 100 billion in base pairs [37]. We assess the feasibility of genome assembly from a probabilistic perspective by adopting the concept of ϵ-feasibility introduced in Lander and Waterman [32]. As in Bresler et al. [31], we define "successful" genome reconstruction according to the notion in the "Human genome sequence quality standards" [38] published by the National Human Genome Research Institute (NHGRI), where "finishing" the sequencing of a given chromosome requires that there should be a contiguous sequence that covers at least 95% of the entire chromosome. Based on this definition, given a set

of reads from multiple sources with **L** (i.e., read length) and **N** (i.e., number of reads), if there exists an assembler that can successfully reconstruct the original genome sequence with a success rate of $1 - \epsilon$, we say the assembly is ϵ-*feasible* with reads $(\mathbf{N}, \mathbf{L})$. The value ϵ can be viewed as the *target failure rate* for genome reconstruction. In the following, we discuss conditions that need to be met for ϵ-feasible assembly.

2.2.1.2 Coverage Condition

Read coverage is defined as the average number of reads that cover a base pair in the target genome sequence. While high read coverage can lead to better assembly, it also results in higher sequencing cost. Obviously, it is impossible to completely reconstruct the whole genome unless every base pair in the genome sequence is covered by one or more reads. Lander and Waterman's coverage condition provides a coverage bound with the required number of reads to make the assembly feasible based on reads with fixed length [32]. The coverage condition can be further extended by considering a set of random reads that originate from a long genome sequence, where their starting locations are assumed to follow a Poisson arrival process [30, 39]. The following proposition summarizes some key properties regarding the read coverage based on multiple read sources.

Proposition 1

1. The probability of having reads without valid overlap can be bounded by:

$$P_{\text{overlap}}(\mathbf{N}, \mathbf{L}) \leq Ne^{-C(1-\theta)}.$$

2. The expected number of contigs is $Ne^{-C(1-\theta)}$.

3. The expected number of reads in a contig is $e^{C(1-\theta)}$.

4. The expected length of a contig (in base-pairs) is given by:

$$\sum_{m=1}^{G} \frac{N}{G} P_L (1 - \frac{N}{G} P_L)^{m-1} (m - 1 + \overline{L}) - L_c(\mathbf{L}) \backsimeq \frac{G}{N} e^{C(1-\theta)} + \overline{L},$$

where $P_L = e^{-C(1-\theta)}$ and $L_c(\mathbf{L})$ is a correction term for the terminal effect and the approximation is based on the long sequence assumption.

In **Proposition 1**, the probability of having non-overlapping reads is dependent on the total number of reads and the average length of reads $(N, \overline{L})$, or equivalently, on the coverage and the average length $(C, \overline{L})$. Given a set of reads with $(\mathbf{N}, \mathbf{L})$, for the assembly to be ϵ-feasible, the configuration $(C, \overline{L})$ needs to lower the probability P_{overlap} of having non-overlapping reads. We define C_{LW} as the minimum coverage that is needed to satisfy the coverage condition so that $P_{\text{overlap}} \leq \epsilon$.

2.2.1.3 Bridging Conditions

The feasibility of assembly also depends on the repeat patterns that are present in the target genome sequence. Repeats may lead to ambiguity unless they can be resolved based on the obtained reads. Figure 2.1 illustrates the examples of repeat patterns that need to be resolved.

A simple example is shown in Fig. 2.1(a), where two identical genome segments of length ℓ_{repeat} are present in the genome sequence. In order to accurately localize such repeats in the target genome, we have to check both sides of each repeating segment to ensure that the two segments are bounded by different neighboring bases. For this, we need a "bridging read" whose length is at least $\ell_{\text{repeat}} + 2$ (see Fig. 2.1(a)). If such a read exists for a repeating segment, it is said to be "bridged" as defined in Bresler et al. [31]. Otherwise, the repeat remains "unbridged". Figure 2.1(b)

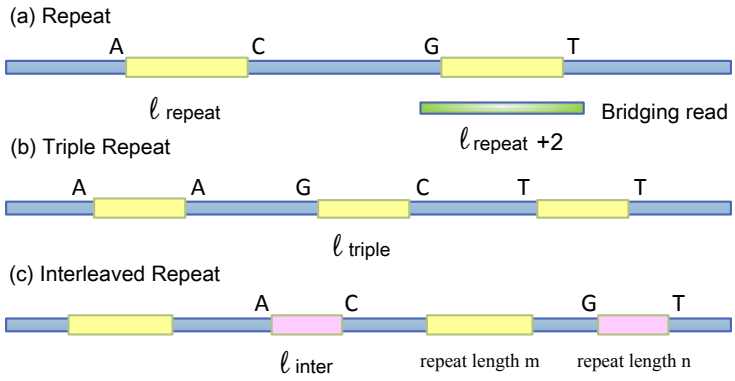


Figure 2.1: Repeats in genome sequences. (a) A repeat pattern that consists of two identical genome segments. A bridging read covering the second segment and the neighboring bases at both ends is depicted. (b) A triple repeat pattern. (c) An interleaved repeat pattern.

illustrates another example, called a triple repeat. A triple repeat consists of three identical genome segments, and we say that the triple repeat is unbridged if none of the segments is bridged. There can be also interleaved repeats, where two different pairs of repeats are located in an interleaved manner as shown in Fig. 2.1(c). As before, an interleaved repeat is said to be unbridged if none of the repeating segments is bridged. Both triple repeats and interleaved repeats can cause ambiguity unless there exist bridging reads that allow us to distinguish the repeating segments and properly locate their respective positions in the genome.

As illustrated in Fig. 2.2, in the absence of bridging reads, we cannot unambiguously resolve the locations of the repeating segments. Such segments may be switched during the assembly process, and as a result, the ϵ-feasible assembly may not be guaranteed. For an unambiguous assembly in the presence of triple repeats and interleaved repeats, we need bridging reads whose length is longer than the **critical length** $\ell_{\text{crit}} = 1 + \max\{\ell_{\text{inter}}, \ell_{\text{triple}}, \ell_{\text{self}}\}$ to ensure ϵ-feasible assembly. ℓ_{triple} denotes the longest length of any triple repeat, and ℓ_{inter} is the maximum length

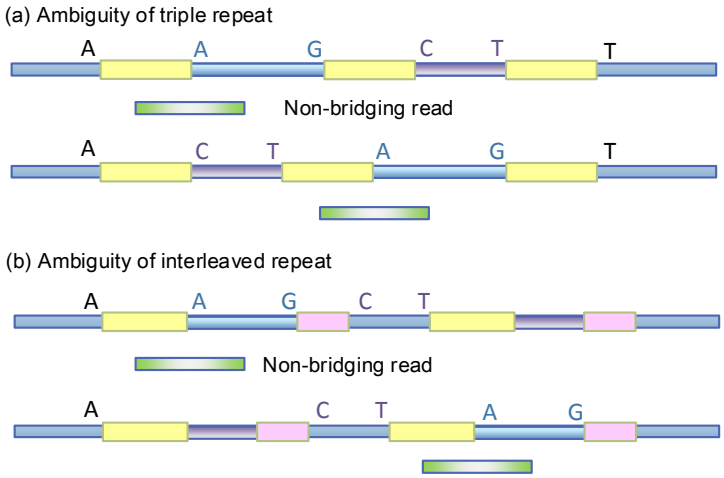


Figure 2.2: Ambiguity from repeats. (a) Ambiguity from triple repeat without bridging read. (b) Ambiguity from interleaved repeat without bridging read.

of a shorter repeat that is involved in an interleaved repeat, when all such shorter repeats for all interleaved repeats in the target genome are considered. ℓ_{self} is the length of the longest self-repeat, which will be discussed later. This is typically referred to as the Ukkonen's condition [31, 40]. The probability of having unbridged repeats is another critical factor in considering the feasibility of complete genome assembly. The probability bounds associated with unbridged reads are summarized in **Proposition 2**.

Proposition 2

1. An interleaved repeat is unbridged if neither repeat pair is bridged. Therefore, the probability bound for all interleaved repeats to remain unbridged is given by:

$$P^{(2)}_{\text{bridged}} = \sum_{m,n} b_{m,n} e^{-2\sum_i \frac{N_i}{G}[(L_i-m-1)^+ + (L_i-n-1)^+]},$$

where $b_{m,n}$ is the number of interleaved repeats in the genome sequence with repeat

lengths $m \geq n$, and $(L-n-1)^+ = \max(L-n-1, 0)$.

2. The probability bound for triple repeats that can lead to assembly error is given by $P^{(3)}_{\text{bridged}} = \sum_m d_m e^{-3\sum_i \frac{N_i}{G}(L_i-m-1)^+} + P^{(3)}_{\text{comb}}$, which is the sum of the probabilities that all triple repeats remain unbridged and a correction term to account for cases where some bridged triple repeats may still lead to ambiguity. d_m is the number of triple repeats with length m, and $P^{(3)}_{\text{comb}}$ is the correction term for the case when some triple repeats are also involved in other repeats in the target genome to be reconstructed.

3. Finally, the overall bound for the probability that the repeat patterns in the genome may lead to ambiguity in the assembly process is given by $P_{\text{bridged}}(\mathbf{N}, \mathbf{L}) = P^{(2)}_{\text{bridged}} + P^{(3)}_{\text{bridged}}$.

For ϵ-feasible genome assembly, there should be a read source whose reads are longer than the critical length ℓ_{crit}, and at the same time, $P_{\text{bridged}}(\mathbf{N}, \mathbf{L})$ has to be less than ϵ. In other words, for a read set $(\mathbf{N}, \mathbf{L})$ to have an ϵ-feasible assembler, there should be long reads with length L_i such that $L_i > L_{\text{crit}}$. At the same time, there should be a sufficient number of long reads to ensure that $P_{\text{bridged}}(\mathbf{N}, \mathbf{L}) \leq \epsilon$.

2.2.1.4 Self-repeat Bridging Conditions

A self-repeat consists of multiple consecutive repetitions of the same sequence pattern, as seen in tandem repeats and poly-A segments. Let us consider a self-repeat segment that consists of identical nucleotide bases. For instance, we may have a segment of n repeated adenines (A), which we denote as A^n for simplicity. If similar self-repeat patterns appear more than once in the target genome, they can lead to ambiguities in the assembly. Figure 2.3 illustrates such an example, where we have two self-repeats A^n and A^m whose locations may be switched during the assembly process. To ensure ϵ-feasible assembly, we need to resolve these ambiguities, for which

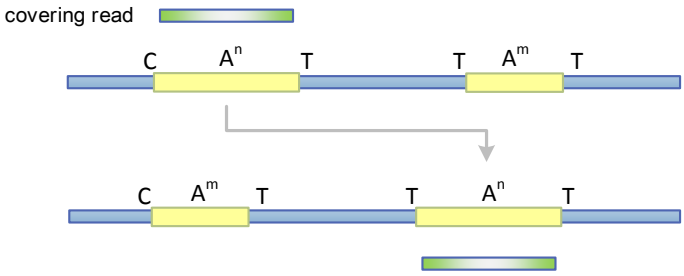


Figure 2.3: Ambiguity due to multiple self-repeats. Unless the covering read can cover the entire self-repeat and its both ends, there will be ambiguity in the assembly.

we need bridging reads that can cover the self-repeats and the neighboring pairs at both ends. As a result, the self-repeat condition requires bridging reads that are longer than $\ell_{\text{self}} + 1$, where ℓ_{self} is defined as the length of the longest self-repeat in the target genome. An upper bound P_{self} for the probability of having ambiguous self-repeats is given in **Proposition 3**.

Proposition 3 Given the number of self-repeats c_m with length m, the bound for the probability of having unbridged self-repeats is:

$$P_{\text{self}}(\mathbf{N}, \mathbf{L}) = \sum_{m} c_m e^{\sum_i \frac{N_i}{G}(L_i - m - 1)^+}.$$

2.2.1.5 Final Conditions for ϵ-feasible Genome Assembly

So far, we have considered several conditions that need to be met to guarantee ϵ-feasible assembly and have derived the probability bounds for non-overlapping reads, unbridged interleaved repeats, unbridged triple repeats, and unbridged self-repeats. Based on our analysis, we arrive at the following overall probability bound:

$$P_{\text{feasible}}(\mathbf{N}, \mathbf{L}) = P_{\text{overlap}}(\mathbf{N}, \mathbf{L}) + P_{\text{bridged}}(\mathbf{N}, \mathbf{L}) + P_{\text{self}}(\mathbf{N}, \mathbf{L})$$

Therefore, given a read set $(\mathbf{N}, \mathbf{L})$, the assembly of the target genome is ϵ-feasible if $P_{\text{feasible}}(\mathbf{N}, \mathbf{L}) \leq \epsilon$.

2.3 Minimizing the Cost of Sequencing

When we have access to multiple read sources that use different sequencing technologies, there exists considerable flexibility regarding how to combine the available read sources – *i.e.*, how many reads to draw from each source. Reads from different sources may have different lengths as well as different per-base sequencing cost. As a result, this flexibility can be exploited to minimize the overall sequencing cost while ensuring ϵ-feasible assembly. Intuitively, this can be achieved by utilizing longer reads (which are typically more expensive) to meet the bridging conditions and the shorter reads (which are relatively inexpensive) to meet the coverage condition. The problem of identifying the optimal strategy for combining multiple read sources can be naturally formulated as a constrained optimization problem, where the goal is to find the read set with $(\mathbf{N}, \mathbf{L})$ that satisfies the assembly feasibility conditions and minimizes the total sequencing cost. Let w_i be the per-base sequencing cost to obtain reads from the i-th source, where L_i and N_i are the read length and the number of reads, respectively. Then the optimization problem can be formulated as follows:

$$\begin{aligned} \underset{\mathbf{N}=\{N_i\}}{\text{minimize}} \quad & \left(\sum_i N_i L_i w_i \right) \\ \text{s.t.} \quad & P_{\text{feasible}}(\mathbf{N}, \mathbf{L}) \leq \epsilon; \;\; N_i \in \mathbb{N}, \forall i. \end{aligned}$$

It is important to note that the above optimization problem is a convex discrete optimization problem with constraints as stated in **Proposition 4**.

Proposition 4 The problem of finding the optimal read numbers $\mathbf{N}$ that minimize the sequencing cost is a discrete convex optimization problem if $L_i \geq \frac{2\rho}{C}\overline{L} \simeq \frac{2}{C}\overline{L}$ for

all $L_i \in \mathbf{L}$, where $\rho = \frac{N}{(N-1)(1-\theta)}$.

Furthermore, minimizing the assembly feasibility bound P_{feasible} based on a fixed sequencing budget is also a discrete convex optimization problem. In case there are two read sources with read lengths L_1 and $L_2 (> L_1)$, the condition for the sequencing cost minimization problem to be convex is $L_2 \leq \frac{C}{2} L_1$. This condition is easily met in practical cases, since the coverage is typically high in order to achieve accurate genome assembly. As there exist efficient techniques for solving convex optimization problems, the cost minimization problem at hand can be solved by mixed-integer convex programming without difficulty [41, 42]. Moreover, fairly accurate approximate solutions can be found in a very efficient manner, by relaxation of the discrete variables [43].

2.3.1 Assembly Algorithm

Most genome assembly algorithms take the so-called overlap-layout-consensus approach to reconstruct the genome from a large number of short reads. When using a greedy strategy, reads with overlap are gradually joined together to form a longer contig. This assembly process can be easily trapped in local optima due to the ambiguities that may arise from repeat patterns that are present in the genome sequence [30, 44]. To resolve such ambiguities, we need longer reads that can bridge the repeats, thereby allowing us to accurately stitch the contigs together. By representing the reads as nodes and by connecting the nodes that correspond to overlapping reads, we can construct an assembly graph that reflects the relationship between the numerous reads [45]. Based on the constructed graph, we can identify the consensus genome sequence by finding a Hamiltonian path in the graph that visits every node exactly once, although finding such a path is computationally expensive [46].

Another approach that is especially popular for genome assembly is the K-mer

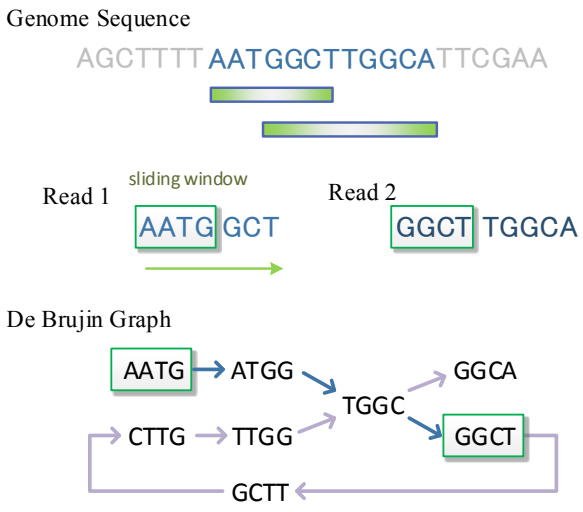


Figure 2.4: Construction of a de Bruijn graph. Each read is scanned by a sliding window with length $K = 4$ to form a graph of K-mers. The K-mer "TGGC" is an X-node with two incoming edges as well as two outgoing edges.

graph based approach, in which reads are assembled by first constructing a de Bruijn graph and then looking for an Euler path [46, 47] that visits every edge exactly once to reconstruct the target genome [31, 48]. A de Bruijn graph is a directed graph of nodes that represent K-mers, where the edges connect the K-mers that overlap in some of the reads. To construct a de Bruijn graph, each read is scanned sequentially by sliding a window of length K, and the scanned subsequences are extracted as K-mer nodes.

As illustrated in Fig. 2.4, K-mers that are adjacent in a given read are connected to each other in the de Bruijn graph in the same order. K-mers from different reads may also be connected if the reads overlap by at least K bases. Due to the presence of repeats in the target genome sequence, K-mers may have more than one incoming or outgoing edges in the de Bruijn graph. A K-mer with multiple incoming and outgoing edges is referred to as an X-node [31]. If there are only simple X-nodes in a given de

Bruijn graph, which correspond to non-interleaved pairwise repeats, it is relatively easy to find the target genome sequence through the Euler tour algorithm [47, 49]. However, if interleaved or triple repeats are present in the target genome, the graph will contain tangled X-nodes leading to multiple candidate Euler paths. Unless such repeats are properly resolved, ϵ-feasible assembly cannot be guaranteed.

Repeats, and therefore the X-nodes that correspond to repeats, can be distinguished by bridging reads, which motivates us to remap X-nodes to the corresponding reads and resolve the multi-path problem through the use of bridging reads. The "multi-bridging" algorithm proposed in Bresler et al. [31] takes a K-mer based approach and incorporates a scheme to bridge X-nodes to ensure ϵ-feasible reconstruction of the target genome, in case the read set satisfies the following conditions: (i) triple repeats are all bridged; (ii) at least one repeat is bridged in an interleaved repeat; (iii) the genome sequence is covered by reads with valid overlap (minimum of K). The first condition ensures that X-nodes whose in-degree and out-degree are higher than two are *all* bridged. This increases the probability bound for unbridged triple repeats to $\sum_m 3d_m e^{-\sum_i \frac{N_i}{G}(L_i-m-1)^+}$, where d_m is the number of triple repeats and the factor 3 is due to the requirement that all three repeating segments in each triple repeat should be bridged. Despite this increase, the multi-bridging algorithm has

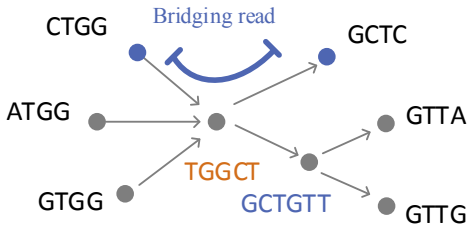


Figure 2.5: Example of an unresolved X-node for a triple repeat. The X-node TGGCT is not resolved since not all repeats are bridged. The K-mers CTGG and GCTC are marked with the bridging read for the unresolved X-node.

been shown to nearly achieve the lower bound on the minimum coverage required for genome reconstruction when ℓ_{inter} is significantly larger than ℓ_{triple}. However, the performance gap is known to increase when the ℓ_{triple} is comparable to or exceeds ℓ_{inter}.

In fact, we can modify the original multi-bridging algorithm proposed in Bresler et al. [31] to further reduce the gap between the *lower bound on minimum coverage depth* required for genome reconstruction and the *actual lowest coverage depth* at which ϵ-feasible assembly can be practically achieved by the algorithm. The performance gap of the enhanced multi-bridging algorithm to the feasibility bound depends on the target genome sequence and the available read sets as illustrated in the Appendix. The pseudocode of the enhanced multi-bridging algorithm is shown in Algorithm 1.

Unresolved X-nodes may cause difficulties when identifying the Euler path, since the path may not be unique in such a case. However, by marking the repeating segment (or, equivalently, the corresponding incoming-outgoing edge pair in an X-node) that can be bridged as described in Step 3, we may still be able to find the correct Euler path from the graph traversal, despite the presence of some unresolved X-nodes (see Fig. 2.5 for illustration). The proposed enhanced multi-bridging algorithm can accurately assemble the target genome sequence if the remaining unmarked repeats are all simple repeats, in which case the Euler path can be uniquely identified. For example, an X-node that corresponds to a triple repeat may be reduced to a simple X-node when one of the repeating segments is bridged, and unless the remaining unbridged repeating segments are involved in another interleaved repeat, the remaining ambiguities can be resolved while traversing the unique Euler path. As a result, the proposed algorithm can resolve additional ambiguities that could not be handled by the original multi-bridging algorithm [31], thereby further reducing any gap between the theoretical lower bound on minimum coverage depth [32] and

Algorithm 1: Enhanced Multi-Bridging Algorithm
Output: sequence $\hat{S}$.
Input: reads R, parameter K.

K-mer graph construction:
1.Build K-mer nodes for all reads R.
  foreach *read belongs to R* **do**
  | Extract and record K-mer from the read
  | Connect K-mer nodes that are adjacent in one read
  end
Repeat *step2 and step3*:
2.Condense the constructed graph.
  foreach *pair of K-mer nodes belongs to graph* **do**
  | Combine K-mer nodes that correspond to a unique path
  end
X-nodes resolution:
3.Bridge X-nodes.
  if *repeat for X-nodes are all bridged* **then**
  | Resolve X-nodes by separating repeats w.r.t. the bridging reads.
  else
  | Mark bridged repeats at X-nodes.
  end
until *all bridging reads are applied*;
4.Find the Euler path in the final graph
5.Reconstruct the genome sequence $\hat{S}$ accordingly.

Figure 2.6: Enhanced Multi-Bridging algorithm.

the lowest coverage depth at which ϵ-feasible assembly is actually possible.

Finally, following the complexity analysis of the original multi-bridging algorithm in Bresler et al. [31], the computational complexity of the enhanced assembly algorithm presented in this section can be analyzed in two phases: (i) K-mer graph construction and (ii) X-node resolution. The run-time for constructing the K-mer graph is bounded by $O(\sum_i (L_i - K)N_i K)$ with the assumption that the complexity for accessing K-mers is $O(K)$. For the resolution of X-nodes, the complexity is upper bounded by $O\left(\sum_i L_i \sum_{m=K}^{L_i} a_m\right)$, where a_m is the number of repeats with length m. Our

enhanced multi-bridging algorithm marks the bridged repeats for unresolved X-nodes in a look-up table, and therefore the computational complexity increase is upper bounded by $O(\sum_{m=K}^{L_{max}} a_m)$, where L_{max} is the longest read length. Consequently, the worst-case computational complexity of our enhanced multi-bridging algorithm is still $O\left(\sum_i L_i \sum_{m=K}^{L_i} a_m\right)$ for the X-node resolution.

2.4 Results and Discussion

To validate the derived feasibility conditions for hybrid genome assembly and to assess the performance of the enhanced multi-bridging algorithm proposed in this section, we conducted extensive numerical experiments using a number of bacterial and archaeal genomes. We considered two read sources with read lengths $\mathbf{L} = \{L_1, L_2\}$ and we sampled error-free reads from the target genome for the two sources with read counts $\mathbf{N} = \{N_1, N_2\}$. We tested our enhanced multi-bridging algorithm for ϵ-feasible assembly at $\epsilon = 5\%$, where K was set to 40 to maintain appropriate complexity for constructing the K-mer de Bruijn graph. Increasing K can reduce the complexity of the X-node bridging step, but on the other hand, it will increase the complexity of building the K-mer graph and also increase the probability of having reads without valid overlap. The read set was sampled from the target genome for different values of average read length, based on $(\mathbf{N}, \mathbf{L})$ predicted to result in ϵ-feasible assembly. The trials were repeated 100 times in each case and the number of successful genome reconstructions was recorded. For a given average read length, the minimum coverage that makes ϵ-feasible assembly possible based on the proposed algorithm was compared to the theoretical lower bound on minimum coverage depth. The vertical axes in Figs. 2.7–2.9 (a,b) correspond to the normalized coverage C/C_{LW}. The green line in each figure shows the assembly feasibility bound for different average read lengths, and the upper-right region is the feasible region for ϵ-feasible genome

assembly. The results based on the original Multi-Bridging algorithm [31] are shown in purple lines for comparison.

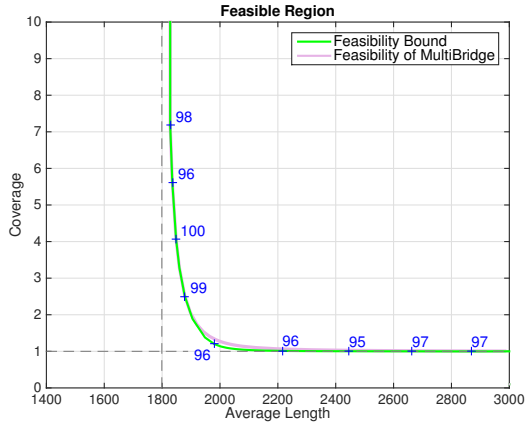
In our studies, we used two bacterial genome sequences from *Staphylococcus aureus* (NCBI ACCN: PRJNA19489) and *Rhodobacter sphaeroides* (NCBI ACCN: PRJNA57653) that are included in the Genome Assembly Gold-standard Evaluations (GAGE) [50]. Additionally, we used the archaeal genome of a *Sulfolobus Islandicus* strain (NCBI ACCN: PRJNA162067), whose triple repeat length ℓ_{triple} is roughly the same as ℓ_{inter}. For each of these genomes, we considered two different read sets. In the first read set, both read lengths L_1 and L_2 were set to exceed the critical length ℓ_{crit} to satisfy the Ukkonen's condition, where the second read length L_2 was set to be similar to ℓ_{crit} to test the critical length. In the second read set, the read lengths (L_1, L_2) were set to (4,300, 150) in order to simulate the case of combining the reads obtained from PacBio P4-C2 [51] and Illumina HiSeq [52]. Although the average read length can take any value between L_1 and L_2, to ensure complete genome reconstruction, the coverage condition needs to be satisfied by properly combining the two types of reads. More specifically, the actual read coverage C should be no smaller than C_{LW}.

It is important to note that a much larger number of short reads are needed (compared to long reads) to suppress P_{bridged}, which is why the normalized coverage shown in Figures 2.7–2.9 (a,b) always surges when most of the reads used for the assembly are short reads (i.e., when the average length is short). As the average read length increases, the term that corresponds to the coverage condition dominates the probability bound $P_{\text{feasible}}(\mathbf{N}, \mathbf{L})$ and the normalized coverage converges to one. In case we are using reads whose lengths are $L_1 = 4{,}300$ and $L_2 = 150$, the average length can be significantly lower than the critical length, as shown in Figs. 2.7–2.9 (b), which is because the bridging condition can be satisfied with a moderate number of long reads ($L_1 = 4{,}300$) while the coverage conditions can be satisfied by the short

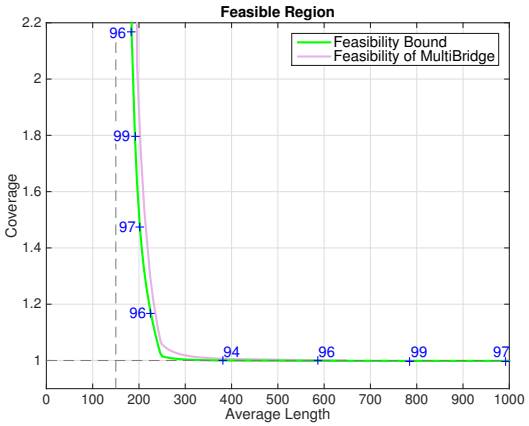
reads. As a result, the coverage does not increase too much to satisfy the bridging condition by using the long reads. However, there is some overhead for incorporating the short reads to improve the coverage, since many of the short reads will be covered by the long reads.

Suppose the ratio between the per-base sequencing cost for the long reads ($L_1 =$ 4,300) and that for the short reads ($L_2 = 150$) is approximately 6:1 [for reference, see 52]. Figures 2.7–2.9 (c) show the minimum sequencing cost for different $\mathbf{N} = (N_1, N_2)$ as well as the optimal value of $\mathbf{N}$ that corresponds to the optimal read set that minimizes the sequencing cost while meeting the assembly feasibility conditions.

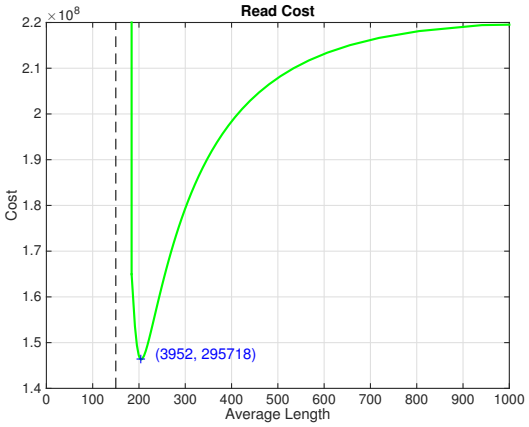
One interesting difference across the experiments based on different genomes is that, in *S. aureus* and *R. sphaeroides* genomes, ℓ_{inter} was significantly larger than ℓ_{triple}, hence the terms corresponding to unbridged interleaved repeats dominated the feasibility probability $P_{\text{feasible}}(\mathbf{N}, \mathbf{L})$. On the other hand, in the *S. Islandicus* genome, ℓ_{inter} and ℓ_{triple} were comparable, hence the term for unbridged interleaved repeats and that for unbridged triple repeats were both significant in $P_{\text{feasible}}(\mathbf{N}, \mathbf{L})$.



(a) ϵ-feasibility with read set ($L_1 = 3{,}000$, $L_2 = 1{,}820$)

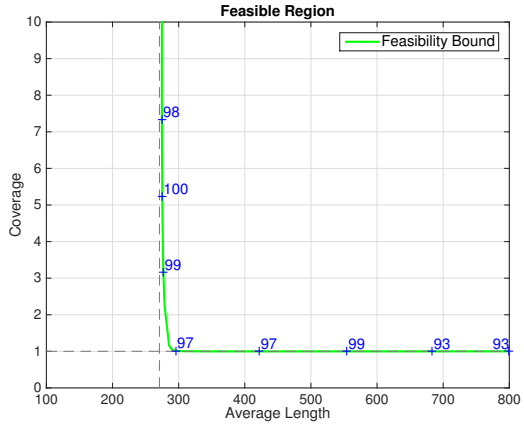


(b) ϵ-feasibility with read set ($L_1 = 4{,}300$, $L_2 = 150$)

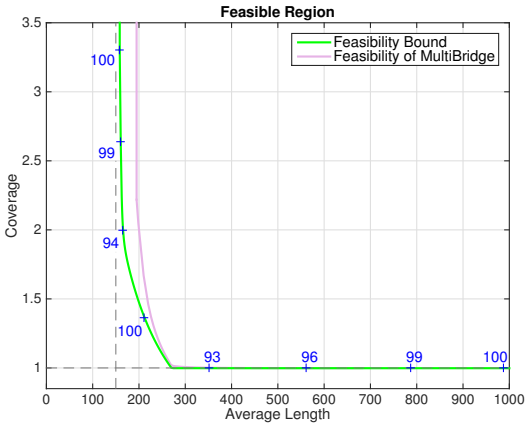


(c) sequencing cost with read set ($L_1 = 4{,}300$, $L_2 = 150$)

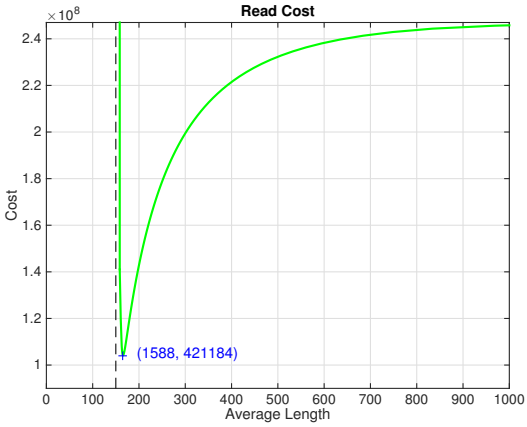
Figure 2.7: *Staphylococcus aureus*: $G = 2{,}872{,}915$, $\ell_{\text{inter}} = 1{,}799$, $\ell_{\text{triple}} = 1{,}397$, $\ell_{\text{self}} = 330$.



(a) ϵ-feasibility with read set ($L_1 = 800$, $L_2 = 274$)

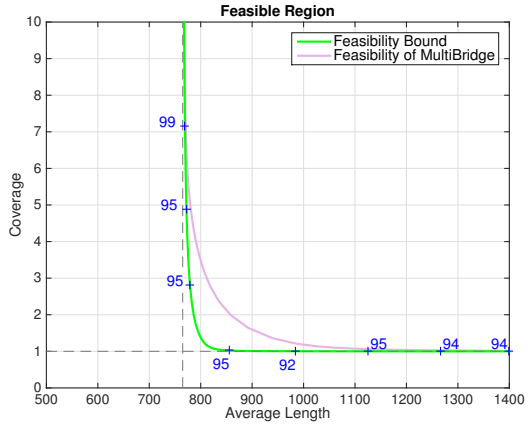


(b) ϵ-feasibility with read set ($L_1 = 4{,}300$, $L_2 = 150$)

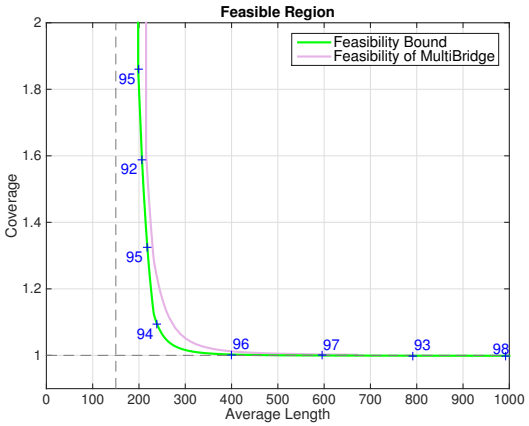


(c) sequencing cost with read set ($L_1 = 4{,}300$, $L_2 = 150$)

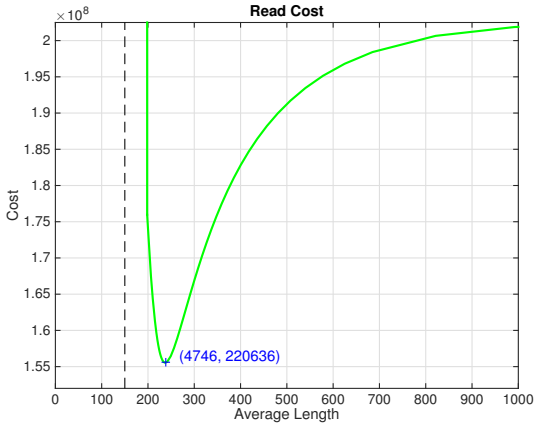
Figure 2.8: *Rhodobacter sphaeroides*: $G = 3{,}188{,}524$, $\ell_{\text{inter}} = 271$, $\ell_{\text{triple}} = 114$, $\ell_{\text{self}} = 126$.



(a) ϵ-feasibility with read set ($L_1 = 1{,}400$, $L_2 = 765$)



(b) ϵ-feasibility with read set ($L_1 = 4{,}300$, $L_2 = 150$)



(c) sequencing cost with read set ($L_1 = 4{,}300$, $L_2 = 150$)

Figure 2.9: *Sulfolobus islandicus*: $G = 2{,}655{,}201$, $\ell_{\text{inter}} = 761$, $\ell_{\text{triple}} = 734$, $\ell_{\text{self}} = 15$.

2.5 Conclusions

In this section, we discussed two important issues that relate to whole genome assembly based on multiple read sources, namely, the feasibility of assembly and the minimization of sequencing cost. Our work extends a previous study by Bresler et al. [31], in which they investigated the assembly feasibility conditions based on a single read source. To take advantage of multiple read sources through the use of different sequencing technologies that are currently available, we examined the conditions that can ensure complete genome reconstruction at a desired success rate based on multiple read sources. An important aspect of hybrid sequencing and assembly is that, when multiple read sources are available, the feasibility conditions for genome reconstruction can be satisfied by different combinations of the available sources. This opens up the possibility of optimally combining the reads to minimize the overall sequencing cost while ensuring complete genome reconstruction. We showed that one can predict the optimal read set that satisfies the feasibility conditions and minimizes the sequencing cost by formulating and solving a constrained discrete optimization problem that is practically convex. Furthermore, we also introduced an enhanced assembly algorithm that improves the performance of the original multi-bridging algorithm in Bresler et al. [31]. Through extensive simulations based on several genomes and different read sets, we verified the feasibility conditions derived in this section, showed the potential of the proposed optimal hybrid sequencing and assembly scheme, and demonstrated the performance of the enhanced multi-bridging algorithm. In this work, we focused on the case of error-free reads in order to investigate the feasibility of complete genome reconstruction based on hybrids reads that do not contain any sequencing error. In addition to the assembly feasibility conditions regarding the minimum required length of the reads, our study provides the theoretical bound on the minimum coverage

required for complete genome reconstruction at the desired success rate. In the presence of sequencing errors, the minimum coverage depth required for complete genome reconstruction is bound to increase, in order to be able to effectively correct the errors for accurate assembly. Assembly feasibility conditions for hybrid reads with potential sequencing errors require further analysis in the future. However, the overall concept and strategy for optimal hybrid sequencing and assembly discussed in this section will carry over to the case when sequencing errors are present.

3. PAIRWISE GLOBAL STRUCTURAL ALIGNMENT OF RNA SEQUENCES THROUGH TOPOLOGICAL NETWORKS

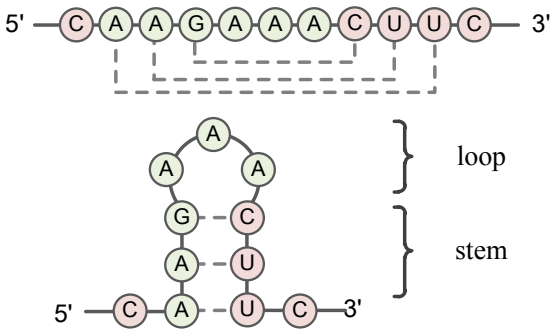
3.1 Introduction

RNA sequence alignment is one of the important bioinformatics tasks for comparative genomic analysis to help speed up functional study and annotation of novel genes as more and more RNAs have been identified through next-generation sequencing (NGS). RNA sequence alignment based on the sequence similarity is one of the common approaches to identify homolog RNA families for comparative analysis. Homologous RNA sequences with a high similarity can be easily recognized through the dynamic programming algorithms that find the optimal alignment to minimize the edit distances between sequences [53, 54]. While for the sequences with a low level of sequence similarity, the performance of sequence alignment based on the edit distance is generally inappropriate due to the increasing discrepancies between the sequences from accumulated nucleotide mutations [55]. The homologous sequences that descend from the same ancestor can share the similar structure and genomic functions, but the sequences might differ significantly due to accumulated mutations from genome evolution. As revealed in the comparative structural analyses, the RNA structures between homologous sequences are more conserved than the sequences themselves [56–60], and therefore RNA sequence alignment should consider their underlying RNA folding structures as well.

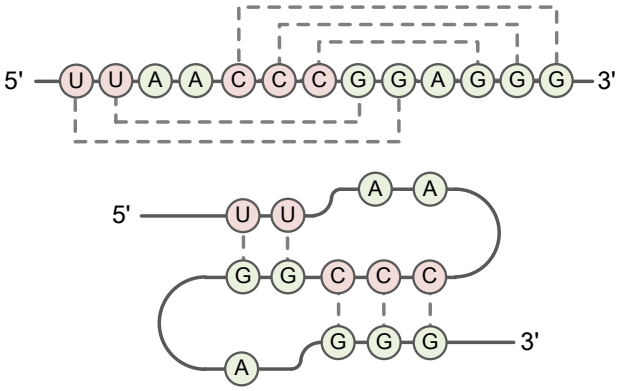
RNA is a single stranded sequence composed of polymers of the nucleotides with four types of nitrogenous bases (A, C, G, and U), and has the comprehensive structure motifs due to the local interaction of the hydrogen bonds between the organic compound purines (A and G) and pyrimidines (C and U). In general, the

native three-dimensional structure analysis for RNA is challenging because of the convoluted tertiary interactions between multiple molecules. Fortunately, due to the quasi-hierarchical folding structure, RNA secondary structure is more stable and predictable, and hence is accessible to mathematical analysis [61–63]. RNA secondary structure is a topology of binary contacts formed by base pairing between Watson-Crick pairs (AU and CG), and wobble pair (GU), and can be further decomposed into stem and loop structures, where the stems are the consecutive stacked base pairs and the loops are unpaired segments bounded by the base pairs as illustrated in Figure 3.1. For the RNA secondary structure, most base pairs stack in a nested style, in which for any two base pairs (i_1, i_2) and (j_1, j_2) either $i_1 < i_2 < j_1 < j_2$ or $i_1 < j_1 < j_2 < i_2$. In addition, they are non-nested crossing base pairs called **pseudoknots**. The RNA structures with pseudoknots make it difficult for structural alignment through the standard dynamic programming approaches.

Without given RNA structures, Sankoff first proposed a dynamic programming algorithm to simultaneously solve RNA sequence alignment and common folding problem (structural alignment) [64]. Several different implementations of **Sankoff algorithm** have been developed for RNA structural alignment. Among these implementations, *Dynalign* and *Foldalign* use the nearest-neighbor thermodynamic model to evaluate the free energies of the secondary structure and finds the structure with the lowest free energy common to the aligned sequences through dynamic programming [65–68]. Similarly, *PARTS* introduces the pseudo free energy model based on the base pairing and alignment probabilities to find the structural alignment with the maximum of the joint probability [67]. However, the complexity of Sankoff algorithm for the structural alignment of two RNA sequences of the length N is $O(N^6)$ in time and $O(N^4)$ in space. The extreme time complexity of Sankoff algorithm is impractical for large-scale genome analysis, and hence a number of simplified



(a) RNA structure with stem-loop



(b) RNA structure with pseudoknots

Figure 3.1: RNA structures. (a) Example of RNA structures with stem-loop. The stems are the consecutive stacked base pairs while the loop is unpaired segments bounded by the base pairs. (b) Example of RNA structures with pseudoknots. The non-nested crossing base pairs are pseudoknots.

variations of Sankoff-like algorithms were developed to efficiently solve the RNA structural alignment problem [55, 69, 70]. By using the base pairing probability as a lightweight energy model, *PMcomp* modifies the dynamic programming with restrictions for the matching base pairs to reduce the computational complexity to $O(N^3)$ in time [71]. Following the lightweight energy model of *PMcomp*, Will's *LocARNA* simplifies the dynamic programming approach with the sparse property

of the base pairing, and further speed up by the ensemble-based sparsification in *SPARSE* to achieve the quadratic time complexity [69, 70]. These programs of Sankoff-like algorithms implement with a more or less complete energy model and find the optimal structural alignment through dynamic programming with various simplifications.

In contrast to the Sankoff-like algorithms, we propose a novel approach for RNA structural alignment by introducing a *topological network* to integrate RNA sequence and structure information. The topological network is a convenient representation for describing elementary features of the underlying structure and has been used to quantify certain topological features of molecular relationships [72, 73], such as gene coexpression networks and neural networks. In particular, one of its specific applications is for global alignment of protein-protein interaction (PPI) networks in comparative analysis. By capturing physical interactions among proteins in the graph model for PPI networks [74], PPI network alignment aims to match proteins across networks in terms of the protein *sequence similarity* and *topological similarity* so that it can transfer functional information of proteins based on aligned or conserved regions across different networks. Here the sequence similarity refers to the degree of homologous resemblance between the protein sequences while the topological similarity is the similarity of interaction profiles between proteins [75]. One remarkably efficient method for global alignment of PPI networks is *IsoRank* algorithm [76] based on the spectral graph approach [77] to find the global alignment of multiple PPI networks.

In this study, we adopt the concept of the topological network alignment to derive an RNA structural alignment by converting RNA sequences to topological networks according to probabilities of folding structure prediction. Our proposed method for structural alignment with topological networks (*TOPAS*) can efficiently capture both the sequence similarity and topological similarity with the computational complexity

$O(N^2)$ in time. Besides, this approach is not restricted to the consecutive nested structures, so that it can support the RNA structures with pseudoknots. Finally, we compare our *TOPAS* algorithm with the Sankoff and Saknoff-like algorithms with the lightweight energy model. We will show the performance comparison results based on the benchmark structural RNA families and demonstrate the efficiency and accuracy for structural alignment through topological networks.

3.2 Materials and Methods

RNA structural alignment aims to align common folding (stem-loop or pseudoknot) structures between given RNA sequences. To achieve this, in addition to the sequence similarity, we innovate to adopt a graphical representation for the sequences composed of nucleotide bases to capture the topological similarity across sequences based on their predicted potential folding structures. Such an integration of sequential and topological information has been proven to be effective in comparative network analysis [78]. One of such effective approaches to estimate the topological similarity across networks is Google's *PageRank* algorithm [79] where its main idea is that a pair of objects are likely to be matched if the contiguous neighbors are also matched. By the similar approach of diffusing the neighborhood similarities, *IsoRank* [76] shows the effectiveness and potential of *PageRank* algorithm in PPI network alignment. For RNA structural alignment, we first construct the topological networks for the RNA sequences, and then estimate the similarity between the constructed topological networks based on the same principle to integrate sequential and structural information by diffusing the neighborhood similarities.

3.2.1 Topological Network Construction from RNA Sequences

In order to construct the topological network for the structural alignment of RNA sequences, we need to identify or infer the pair of interacting nucleotides in the RNA sequences using the probabilities of folding structure prediction. To this aim, we take RNA sequences as sequential backbones and model each nucleotide as a node. Those nodes that can form Watson-Crick pairs or wobble pair are further connected and weighted by the corresponding base pairing probabilities. The topological network for an RNA sequence is similar to the PPI networks with probabilistic base pairing interactions replacing the PPI links. Since the base pairing probability only depends on the structure of each individual RNA sequence, it can be precomputed by using the thermodynamic equilibrium model with the experimentally determined parameters [80–82]. Furthermore, those less-reliable edges with the base pairing probabilities lower than the threshold (P_{Th}) are removed to reduce the computational complexity and enhance the accuracy of the modeling [81].

In addition to the topological structure, the information of sequence resemblance is also incorporated to measure the similarity between topological networks. Though normalized bit-score can be used as an estimation of sequence similarity between nucleotides, a hidden Markov model (HMM) is adopted for more appropriate probabilistic estimation for pairwise sequence similarity between nucleotides in the topological network alignment. Given a pair of RNA sequences, the posterior probability of matched nodes can be efficiently estimated for sequence similarity through the *forward-backward* algorithm in the hidden Markov model [83, 84].

3.2.2 RNA Structural Alignment Through Topological Network

Since topological network alignment aims to align the nodes across different networks in terms of the topological similarity and sequence similarity, to compute the overall similarity R for the pair of nodes, we integrate the following three types of similarities:

1. Structural similarity R_S for RNA secondary structure.
2. Connected similarity R_C for continuous connectedness.
3. Sequence similarity R_E for sequence resemblance.

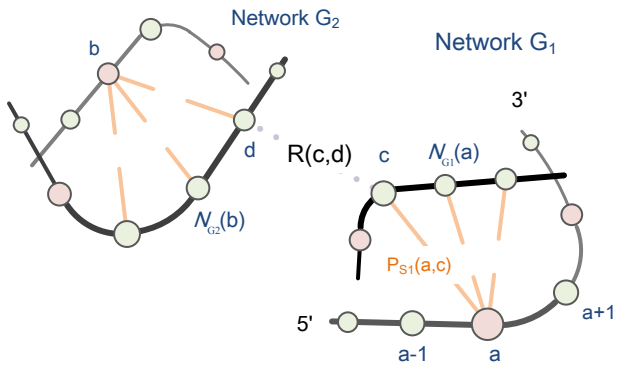


Figure 3.2: The similarity in topological networks. $R(c, d)$ denotes the pairwise similarity between nodes at position c in network G_1 and position d in network G_2. $P_{S1}(a, c)$ is the base pairing probability for nodes at position (a, c) in network G_1. $N_{G_1}(a)$ denotes the set of neighbors of the node at position a if there exists the base pairing interaction in network G_1.

Let $G_n = (V_n, E_n)$ be the n–th topological network, and the nucleotide base in the n-th sequence can be modeled as a node in V_n, and if the nodes have a positive base pairing probability greater than the threshold (P_{Th}), it can be modeled as a weighted edge in E_n. Suppose that two topological networks G_1 and G_2 are compared

to find the network alignment according to the overall similarity R. Let $R(a,b)$ be the overall similarity for the node pair (a,b), where $a \in V_1$ and $b \in V_2$ are node indices in the node sets, and $N_{G_n}(x)$ be the set of connected neighbors of the node x in the topological network G_n. Note that the structural similarity and connected similarity compose the topological similarity for network alignment. As the similar methodology to estimate the similarity in *IsoRank*, two nodes from different networks could be matched if their neighbors are also well matched. In order to reflect such a similarity diffusing principle, we compute the structural similarity $R_S(a,b)$ and connected similarity $R_C(a,b)$ by

$$R_S(a,b) = \sum_{\substack{c \in N_{G_1}(a) \\ d \in N_{G_2}(b)}} \frac{P_{S_1}(a,c) P_{S_2}(b,d)}{D(c)D(d)} R(c,d); \tag{3.1}$$

$$R_C(a,b) = \frac{1}{2}(R(a-1,b-1) + R(a+1,b+1)), \tag{3.2}$$

where $P_{S_1}(a,c)$ is the base pairing probability for nodes at the positions (a,c) in the network G_1 and $P_{S_2}(b,d)$ is the base pairing probability for nodes at the positions (b,d) in the network G_2 as illustrated in Figure 3.2; and $D(c) = \sum_{u \in N_{G_1}(c)} P_{S_1}(u,c)$, $D(d) = \sum_{v \in N_{G_2}(d)} P_{S_2}(v,d)$ are the weighted degrees of nodes c and d, respectively. The structural similarity R_S is associated with its neighbors' similarities according to the probabilistic base pairing interactions to make sure that the alignment matches the nodes that are likely to form base pairs according to the secondary structures. Next, since the consecutive base pairs are likely to stack together to form the stem and loop structure in an RNA secondary structure, the connected similarity R_C is associated with the contiguous similarity to describe the continuous connection in alignment as the message-passing approach [85]. Hence, both the equations attempt to integrate the neighborhood similarity in the network alignment. Finally, the topological similarity

including the structural similarity R_S and the connected similarity R_C are integrated with the sequence similarity R_E to iteratively estimate the similarity R as

$$R = (\alpha \cdot R_S + \beta \cdot R_C + (1 - \alpha - \beta) \cdot R_E), \tag{3.3}$$

where α and β are weighting parameters for the structural similarity R_S and the connected similarity R_C with the constraints $0 \leq \alpha, \beta, \alpha + \beta \leq 1$.
The equation (3.3) can be rewritten in a matrix form as $R = \mathbf{A}R$, where the matrix $\mathbf{A}$ represents the linear combination of the similarities (R_S, R_C, R_E) according to equations (3.1-3.3) that describe the relationships in the neighborhood for similarity, and thus the similarity R can be estimated efficiently by the power method as follows:

$$R^{(k+1)} \leftarrow \mathbf{A}R^{(k)}/|\mathbf{A}R^{(k)}|, \tag{3.4}$$

where $R^{(k+1)}$ is the estimation of similarity in the $(k+1)$–th iteration, and the initial similarity $R^{(0)}$ is set to a random unit vector in L_1–norm. The convergence rate of the power method is dominated by the second largest eigenvalue of the matrix A, but the power iteration can be limited to N_{It} or stop if the residual is lower than a predefined tolerance factor. Based on the estimated similarity, the topological network alignment can be constructed by maximizing the overall similarity through dynamic programming, such as *Needleman-Wunsch* algorithm [53] or *FOGSAA* [86], and then mapped to the final RNA structural alignment.

The computational complexity is dominated by the estimation of overall similarity R, and the sparsity of matrix $\mathbf{A}$ makes the computation efficient in $O(kd_1d_2N^2)$, where k is the number of iterations in power method and (d_1, d_2) are the average base pairing interaction edges in the networks G_1 and network G_2, respectively

($kd_1d_2 \ll N^2$). Additionally, the space complexity is $O(N^2)$ that is much lower than $O(N^4)$ required by the Sankoff algorithm.

The pseudo-code of the proposed RNA structural alignment through topological network is summarized as Algorithm *TOPAS* 3.3 on the following page.

Algorithm 2: RNA Structural Alignment (*TOPAS*)

Output: Structural alignment $(\hat{S}_1, \hat{S}_2)$.

Input: RNA sequences (S_1, S_2), probabilistic model (P_{S_1}, P_{S_2})

Parameters $(\alpha, \beta, N_{It}, P_{Th})$.

1. Construct topological networks

for *n= 1 to 2* **do**

  Construct $G_n = (V_n, E_n)$ from the sequence data (S_n, P_{S_n}, P_{Th})

end

2. Run power method to estimate similarity R

Initialize the similarity vector $R^{(0)}$ with a nonzero random unit vector.

for *k= 1 to* N_{It} **do**

  Initialize R_S, R_C to 0

  for *a= 1 to length(V_1)* **do**

    for *b= 1 to length(V_2)* **do**

      Update structure similarity

      foreach $(c, d) \in (N_{G_1}(a), N_{G_2}(b))$ **do**

        $R_S(a,b) += R^{(k-1)}(c,d)[P_{S_1}(a,c)P_{S_2}(b,d)/D(c)D(d)]$

      end

      Update connected similarity

      if *Exist* $R(a-1, b-1)$ **then**

        $R_C(a,b) += \frac{1}{2}R^{(k-1)}(a-1,b-1)$

      end

      if *Exist* $R(a+1, b+1)$ **then**

        $R_C(a,b) += \frac{1}{2}R^{(k-1)}(a+1,b+1)$

      end

      Update overall similarity

      $R_A^{(k)}(a,b) = \alpha R_S(a,b) + \beta R_C(a,b) + (1-\alpha-\beta)R_E(a,b)$

    end

  end

  Normalize overall similarity

  $R^{(k)} = R_A^{(k)}/|R_A^{(k)}|$

  Stop criterion

  if $|R^{(k)} - R^{(k-1)}| < Tolerance$ **then**

    break

  end

end

3. Run Needleman-Wunch (R) to maximize overall similarity

4. Find RNA structural alignment $(\hat{S}_1, \hat{S}_2)$

Figure 3.3: RNA structural alignment through topological network ($TOPAS$).

3.3 Results and Discussion

RNAstructure (version 5.8) is a software package for RNA secondary structure analysis that includes single structure prediction based on the nearest-neighbor thermodynamic model and sequence alignment derived from an HMM [67, 87]. As similarly done in *PARTS*, which uses precomputed base pairing and alignment probabilities to evaluate the pseudo free energies, the probabilistic model in *RNAstructure* can also be applied to the RNA structural alignment through topological networks. Based on the probabilistic model, the topological network is built and then the parameters and performance assessment are discussed as follows.

3.3.1 Parameters for Topological Similarity

The performances of structural alignment are assessed in terms of sensitivity (SEN)= $\frac{TP}{TP+FN}$ and positive predictive value (PPV)= $\frac{TP}{TP+FP}$, where TP, FP, and FN are the number of true positives, false positives, and false negatives, respectively. In the equation (3.3), where it estimates the overall similarity, the parameter α controls the contribution of the topological similarity R_T and the parameter β controls the contribution of the connected similarity R_C. In general, both the parameters (α, β) in the topological similarity are important in the structural alignment since RNA secondary structures mainly consist of the continuous stem and loop structures, and these parameters can be trained through grid search with training data. In addition, the sequence similarity R_E should be included to avoid symmetric structural ambiguity (i.e. $\alpha + \beta < 1$), and the level of sequence similarity should keep low to avoid dominating the alignment results when analyzing sequences with low sequence identity (SI). To check the performance dependency on the different parameter settings (α and β), tRNA sequence pairs in *Rfam* database [88]: (i) X14835.1/6927-7002 and M32222.1/12777-1363, are selected for the high sequence identity scenario

(SI= 0.77) and (ii) X14835.1/6927-7002 and M86496.1/1024-1089 are selected for the low sequence identity scenario (SI= 0.24). For the sequences with high SI, the performance of structural alignment is not so sensitive to the parameters (α, β) as shown in Figure 3.4. In this case, sequence similarity provides enough information to identify the conserved sequences in the alignment, but including the topological similarity could further improve the performance. However, for the sequences with low similarity, the higher level of the topological similarity gains better SEN and PPV in alignment. In this case, the structural alignment relies on the topological similarity with the well-predicted structure model.

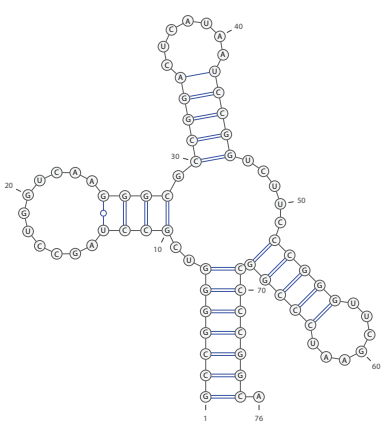
3.3.2 Performance Comparison

In order to evaluate the performance of our proposed structural alignment method through topological networks (*TOPAS*), pairwise sequences without unknown bases from *BRAliBase* 2.1 dataset K2 [89] are used as the benchmark for performance evaluation and comparison. Including 36 RNA structural families, this benchmark has total 8,587 pairs of RNA sequences with the average length 109 nt and average sequence identity 0.67. We compared the performance of the proposed method against the structural alignment methods based on the Sankoff and Sankoff-like algorithms as listed in Table 3.1.

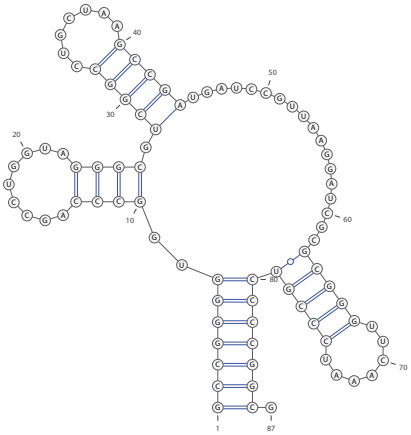
In the following analysis, the RNA structural alignment through topological networks is abbreviated as *TOPAS* with the corresponding parameters $(\alpha, \beta, N_{It}, P_{Th})$= (0.40, 0.56, 30, 0.01), and the computational time was measured when running the experiments on an iMAC (3.5GHz/ 32 GB RAM/ OS X 10.9.5). The computation time of structural alignment with *TOPAS* depends on the sequence lengths and the number of probabilistic interaction edges inferred by the probabilistic model, while the alignment performance depends on the accuracy of the probabilistic model. As we

can see in Table 3.2, *TOPAS* outperforms the programs based on Sankoff or Sankoff-like algorithms, and the computation of *TOPAS* is significantly more efficient than the other algorithms.

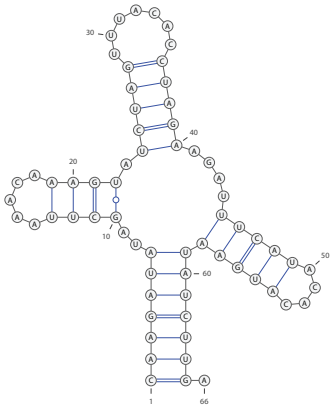
To thoroughly evaluate our proposed *TOPAS* algorithm, the sequences with the sequence identity ranging from the value $n - 0.05$ to $n + 0.05$ are grouped into the corresponding SI class n to help evaluate the alignments with different levels of



(a) tRNA X14835.1

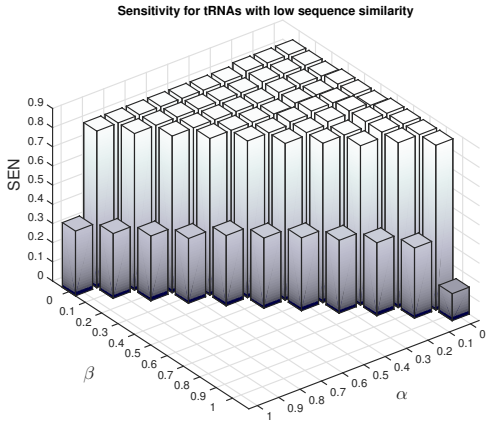


(b) tRNA M32222.1

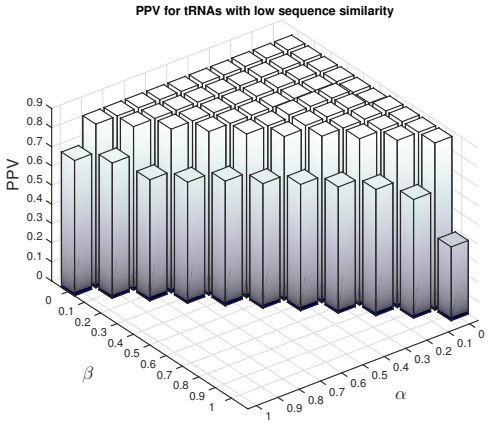


(c) tRNA M86496.1

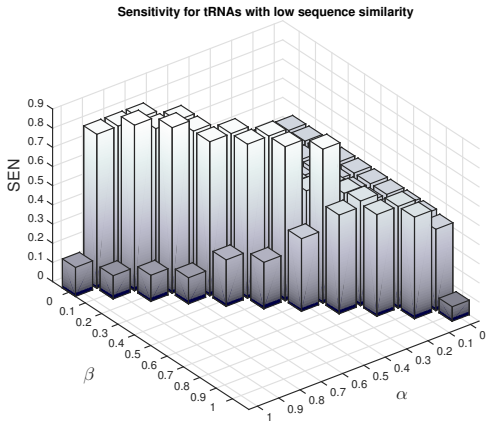
Figure 3.4: Secondary structure of tRNA sequences: (a) tRNA X14835.1/6927-7002, (b) tRNA M32222.1/1277-1363, and (c) tRNA M86496.1/1024-1089. The RNA secondary structures were drawn with *VARNA* [1].



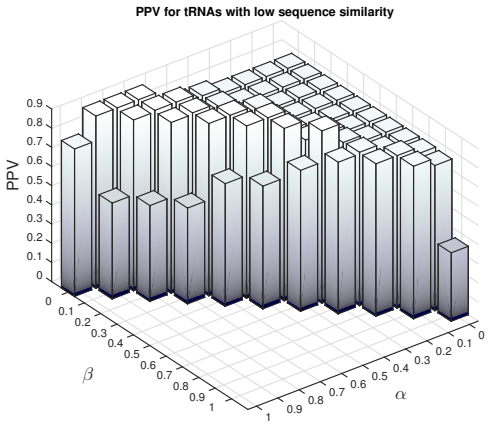
(d) SEN for high SI



(e) PPV for high SI



(f) SEN for low SI



(g) PPV for low SI

Figure 3.4: (Continued) Secondary structure of tRNA sequences: Sensitivity (SEN) and positive predictive values (PPV) for different sequence similarities: (d) SEN for tRNA with the high SI, (e) PPV for tRNA with the high SI, (f) SEN for tRNA with the low SI, and (g) PPV for tRNA with the low SI.

sequence similarity. Figure 3.5 shows the performances with respect to the classified sequence identity. As illustrated in the figure, although *Dynalign* has shown promising results in structural prediction, it does not give the best structural alignment because the sequence similarity is not included when aligning RNAs and only the helix regions

Table 3.1: Structural alignment programs used in performance comparison.

Program	Version/Package	Command† (Configure file)	Reference
PARTS	RNAstructure 5.8	parts default.conf	[67]
Dynalign2	RNAstructure 5.8	dynalign_ii default.conf	[68]
Foldalign	2.1.0	foldalign -global seq_files	[66]
LocARNA	LocARNA 1.8.7	locarna seq_files	[69]
SPARSE	LocARNA 1.8.7	spare seq_files	[70]

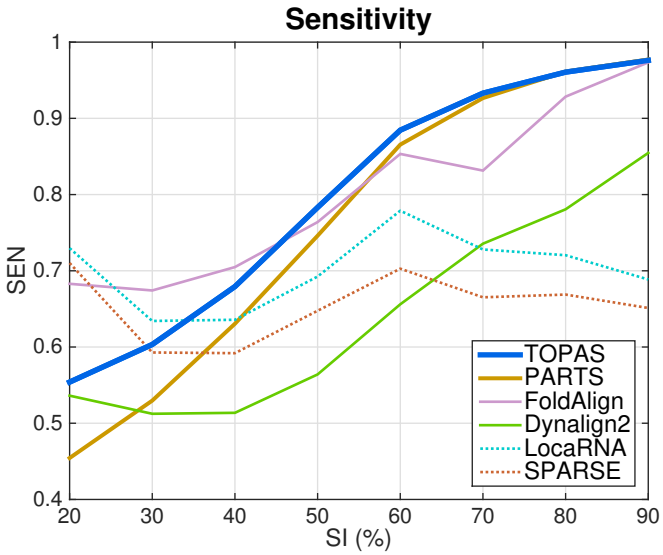
†Note that "Command (Configure file)" column describes the command to run the program and we used the default configurations provided by the corresponding programs.

Table 3.2: Performances for BRAliBase 2.1 K2 dataset.

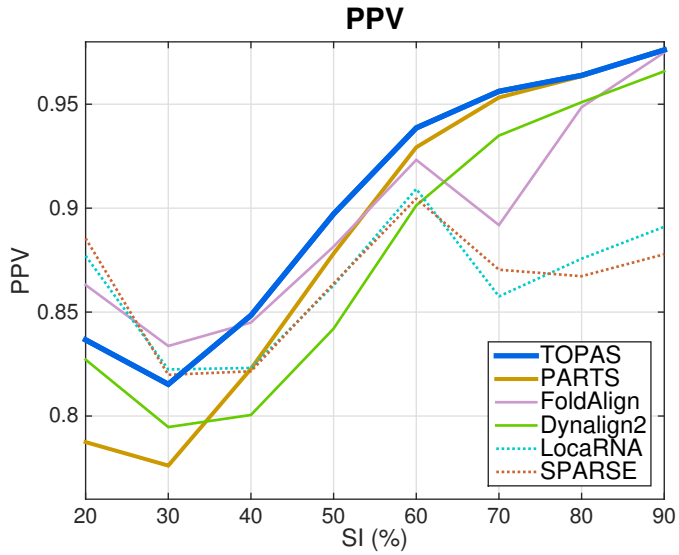
	SEN	PPV	Log_{10}(Time)
TOPAS	**0.878**	**0.938**	3.349
PARTS	0.860	0.931	5.625
Foldalign	0.860	0.923	5.657
Dynalign2	0.706	0.914	5.803
LocaRNA	0.704	0.873	3.697
SPARSE	0.654	0.869	**3.281**

are aligned [68]. The performance of *PARTS* drops significantly when the sequence identity decreases because of the inaccuracy of its probabilistic model for those small samples. It is clear that the structural alignment of *TOPAS* algorithm is accurate especially for the sequences with high sequence identity in the benchmark. For the sequences with low sequence identity, the SEN and PPV of *TOPAS* alignments are not as good as *FoldAlign*. That is because the structural alignment depends on the estimation of topological similarity, but the probabilistic model is not accurate enough for these sequences in the benchmark so that the accuracy of the estimation in topological similarity is degraded.

In order to evaluate the performance of structural alignment for RNA sequences with pseudoknot structures, the sequence pairs of Downstream-peptide and wcaG RNA are taken from *Rfam* database for performance evaluation. The performances are summarized in Table 3.3, where 2,000 random pairs are selected from each RNA structural family for the test. We can again observe that the performance improvement of our *TOPAS* algorithm is remarkable in both alignment accuracy and computation time. In addition, the performance of *TOPAS* alignment can be further improved if the structures with pseudoknots can be better estimated. In Table 3.3, *TOPAS* (PK) denotes that the results obtained by *TOPAS* alignment given with the minimum crossing base pairs of pseudoknots in Downstream-peptide and wcaG RNA. For instance, for RNA structure with pseudoknots in Figure 3.1b, the wobble pairs (GU) are minimum crossing base pairs, and the remaining structure without those crossing pairs is a simple stem and loop structure. There are in average 5 and 6 minimum crossing base pairs in Downstream-peptide and wcaG RNA families respectively, and the topological networks given with crossing base pairs can include the pseudoknot structures if well-predicted. The *TOPAS* (PK) improves the performance and demonstrates it works well for RNA structures with pseudoknots when the topological networks are appropriately constructed.



(a) SEN with respective to SI



(b) PPV with respective to SI

Figure 3.5: Performances for *BRAliBase* 2.1 K2 dataset. (a) SEN with respective to SI. (b) PPV with respective to SI.

Table 3.3: Performances for RNA structures with pseudoknot

	wcaG RNA			Downstream-peptide RNA		
	SEN	PPV	Log_{10}(Time)	SEN	PPV	Log_{10}(Time)
TOPAS	**0.847**	**0.911**	**2.410**	**0.861**	0.899	**1.908**
TOPAS (PK)	0.854	0.912	2.401	0.866	0.901	1.903
PARTS	0.839	0.908	4.401	0.827	0.895	3.879
Foldalign	0.834	0.905	3.381	0.805	0.890	2.725
Dynalign2	0.413	0.806	3.979	0.438	0.797	3.266
LocaRNA	0.827	0.902	2.730	0.834	0.898	2.544
SPARSE	0.738	0.901	2.444	0.85	**0.906**	1.991

3.4 Conclusions

Many approaches developed for RNA structural alignment have comparable performances with different strength and weakness. Sankoff algorithms simultaneously optimize folds and alignments to minimize the free energy but require extremely high complexity both in time and space. In this study, we proposed an efficient approach for the pairwise structural alignment of RNA sequences. We first build the topological networks based on the probabilistic model for potential folding structures of RNAs, and then performs structural alignment based on the estimated similarity that integrates topological similarity and sequence similarity. Through the extensive performance comparison over the RNA structural families and the benchmark *BRAliBase* 2.1 K2 dataset, our proposed *TOPAS* method is efficient and the performance is comparable to the Sankoff algorithm with significantly improved computational efficiency. Moreover, the proposed structural alignment through topological networks is not restricted to nested folding structures and can effectively align RNA sequences with pseudoknots. Thus structural alignment with *TOPAS* provides a significant advantage in accuracy and efficiency without structural restriction for genomic analysis.

4. EFFECTIVE COMPUTATIONAL DETECTION OF PIWI-INTERACTING RNAS USING N-GRAM MODELS AND SUPPORT VECTOR MACHINE *

4.1 Introduction

The Piwi-interacting RNA (piRNA) is a new class of small non-coding RNAs (ncRNAs) whose functions are not fully understood. Recently, the studies have shown that piRNAs are associated with control of transposon silencing, transcriptional regulation, and mRNA deadenylation [10, 90, 91]. The piRNAs interact with Piwi proteins to form RNA-protein complexes involved in silencing of retrotransposons and other genetic elements. Furthermore, piRNAs are found to be emerging players in cancer genomes, and hence to have potential clinical utilities [19, 20]. Thus, there is a prompt demand for identifying the novel piRNAs through effective computational methods due to their clinical prospect. However, piRNA detection is not straightforward since piRNAs lack conserved structure motifs and sequence homology between different species [92, 93].

The piRNAs are the largest class of small ncRNAs with a wide variety of sequences in size about 26-31 nucleotide bases [94, 95]. There are two major classes of approaches developed for piRNA detection. The first class utilizes sequence-based features to identify piRNAs [96, 97]. Betel et al. [96] found piRNAs have the tendency to have the nucleobase Uridine at the 5' cleavage sites and identified piRNAs by checking the Uridine positions and its 10 upstream and downstream bases. However, the prediction based on the Uridine positions is not accurate and the classification accuracy is 61-72% for Mouse piRNAs. The K-mer scheme [97] can have a superior performance

*Reprinted with permission from Effective computational detection of piRNAs using n-gram models and support vector machine by Chun-Chi Chen, Xiaoning Qiana, Byung-Jun Yoon, 2017. In proceeding of BMC bioinformatics.

by checking the frequencies of K-mer strings. All 1,364 K-mers from 1-mer strings to 5-mer strings are included to predict piRNAs. Since most piRNAs are derived from genomic piRNA clusters [98–100], the second class utilizes the information on clustering locus for piRNA detection. Among the approaches based on clustering locus of piRNAs, proTRAC [101] can identify piRNA clusters and piRNAs from a small RNA-seq dataset through a probabilistic analysis of mapped sequence reads. Furthermore, piClust [102] uses a density-based clustering method to identify piRNA clusters without assuming any parametric distribution model. Besides, the sequence-based approach can further incorporate distinctive features to detect piRNAs. For example, *piRPred* [103] integrates both the features of K-mer string and clustering locus based on multiple kernel fusion.

In this section, we propose a novel sequence-based piRNA detection algorithm, called *piRNAdetect*, which can be used to detect novel piRNAs in genome sequences. First, we adopt the n-gram models (NGMs) based on the seed sequences to efficiently classify the recognized piRNAs into the homologous families. By integrating NGMs into the sequence classification, it enables flexible exploration of different sequence motifs and patterns in a dataset. Based on the classified families, we can further build the corresponding NGMs and utilize the support vector machine (SVM) to detect the potential piRNAs. The performance results based on the piRNAs from distinct species in the piRBase [104] database demonstrate the efficiency and the accuracy for piRNA detection using *piRNAdetect*.

4.2 Materials and Methods

The main task of piRNA detection is to identify novel piRNAs in genome sequences. To achieve this, we first adopt the n-gram model (NGM) to classify a given database of recognized piRNAs into families with similar sequence motifs. The NGM is a

class of probabilistic models, widely applied in bioinformatics research, including protein identification [105, 106], RNA structure modeling [107], and genome sequence analysis [108]. Based on homologous sequences, the NGM can estimate the similarity between sequences with the tolerance for the potential variations involved with insertions, deletions, and substitutions in the nucleotide or amino acid sequences [108]. The NGM is an $(n-1)$th-order Markov chain model and each nucleotide or amino acid base in a sequence only depends on what the preceding $(n-1)$ bases are. Therefore, the homologous likelihood for a sub-sequence with length L in the sequence $\underline{b}$ can be efficiently estimated by the following equation (4.1):

$$R(\underline{b}, k) = \log P(b_{k+1,k+n-1}) + \sum_{i=k+n}^{k+L} \log P(b_i | b_{i-n+1,i-1}), \tag{4.1}$$

where k is the offset of the sub-sequence in $\underline{b}$, and b_i represents the i^{th} base of the sequence $\underline{b}$ while $b_{i,j}$ represents the sub-sequence $(b_i, b_{i+1}, \cdots, b_j)$ in $\underline{b}$. Moreover, the likelihood $R(\underline{b}, k+1)$ can be efficiently updated from $R(\underline{b}, k)$ when scanning the sequence $\underline{b}$ to search for the homology.

For the sake of piRNA detection, we can first classify the piRNA sequences into homologous families through NGMs based on the seed sequences in the dataset. Based on the classified families, we can then build the corresponding NGMs for detection and further extract the features through the NGMs for an SVM to detect piRNAs. Based on this idea, we propose a novel piRNA detection method called *piRNAdetect*. The procedure for piRNA detection using *piRNAdetect* is detailed in the following subsections.

4.2.1 Clustering Sequences That Share Common Motifs

For a given dataset of sequences, we can classify the sequences with similar motifs into a homologous family through the NGM based on the seed sequence. Since there exists a subset of piRNAs derived from repeat regions [109, 110], some piRNAs have common motifs with repeat sub-sequences. Hence the sequence with the highest (n-1)-grams frequency is first taken as a seed to collect sequences with the similar sequence motifs. Based on the seed sequence, we can estimate the state probability $P(b_{k+1,k+n-1})$ and the transition probability $P(b_i|b_{i-n+1,i-1})$ of the sequence $\underline{b}$ from the statistics, and a pseudo-count is added in the statistics to model potential mutations. Furthermore, the maximum $R(\underline{b}, k)$ for all the sub-sequences with length L, which is set to the minimum sequence length within the dataset, is taken as the homologous sequence similarity $S(\underline{b})$. To normalize the bias of the sequence content in the sequence classification, the Z-score is adopted as the final similarity measure of the given sequence with respect to the corresponding NGM:

$$Z(\underline{b}) = \frac{S(\underline{b}) - \mu}{\sigma}, \tag{4.2}$$

where $S(\underline{b})$ is the sequence similarity of the sequence $\underline{b}$, and the parameters μ and σ are the average and the standard deviation of the sequence similarity over the statistical ensemble for the dataset. Lastly, those similar sequences with the Z-score $Z(\underline{b}) \geq Z_{th}$ are collected as a homologous family if the collected sequence number $N \geq N_{th}$, where the parameters Z_{th} and N_{th} are predefined threshold values. The classified family is then extracted from the dataset, and the process to classify sequences into the homologous family is repeated until all sequences in the dataset are checked to be the potential seeds.

4.2.2 Predicting piRNAs Using NGM-based Features

For the purpose of piRNA detection, we first update the NGMs based on the classified sequences with the similar process as in the sequence classification. For each classified family, the state probability and the transition probability with pseudo-counts are estimated for the corresponding NGM. Since we utilize the Z-score of the sequence similarity $S(\underline{b})$ to normalize the bias of sequence length and family sequence content, the statistical average and the standard deviation of the sequence similarity are computed based on 18,000 randomly generated sequences obtained from Monte Carlo shuffling simulation [111]. Moreover, the lengths of the test sequences in the statistical evaluation are ranged from 21 to 36 nucleotides with a step size of 5, and the Z-score of the sequence similarity can be further estimated by SVM regression analysis based on the statistical averages and the standard deviations. The LIBSVM package [112] is employed for SVM regression based on the ϵ-support vector regression models using the radial basis function (RBF) kernel. With the Z-scores of the sequence similarities from the NGMs with respect to the classified families, *piRNAdetect* incorporates those features to detect piRNAs based on the SVM classifier.

In order to train the SVM classifier for piRNA detection, the sequences are drawn from the piRBase [104] and Rfam database 12.1 [113, 114] to construct the datasets with positive samples and negative samples for training and assessment. For each sequence in the positive samples, the sub-sequence with the same length is randomly drawn from the Rfam database and is shuffled to be considered as the negative control sample. Based on the dataset, we can train a c-support vector classification (c-SVC) model using the RBF kernel through the LIBSVM package [112] to detect potential piRNAs and compute the confidence probability for piRNA detection in a given

genome sequence.

4.3 Results and Discussion

To test *piRNAdetect*, the piRNAs from the piRBase database with length from 26 to 36 are randomly taken to test the performance using 5-fold cross-validation (CV) approach. In the 5-fold CV, the test samples are randomly partitioned into 5 equal sized folds, and each fold is in turn retained as the test data for the validation while the remaining 4 folds are taken as the training data. The piRNA detection performance is evaluated in terms of the accuracy (ACC)=$^{(\mathrm{TP}+\mathrm{TN})}/_{(\mathrm{TP}+\mathrm{TN}+\mathrm{FP}+\mathrm{FN})}$, the true positive rate (TPR)=$\frac{\mathrm{TP}}{\mathrm{TP+FN}}$, and the false positive rate (FPR)=$\frac{\mathrm{FP}}{\mathrm{TN+FP}}$. TP denotes the number of correctly identified piRNAs, and TN denotes the number of correctly identified negative samples. FP denotes the number of negative samples incorrectly identified as piRNAs, and FN denotes the number of piRNAs that are missed in the detection.

In order to apply the n-gram model to piRNA detection, the size of n needs to be less or equal to the length of the target string. Besides, the larger size of n is suitable for the sequences with longer common motifs while the smaller size of n is proper for the sequences with intensive variations. Since piRNAs are divergent in both their structure and sequence, the tetragram is used to have superior performance in piRNA detection with reasonable computational complexity. In the following discussion, the parameters in the clustering sequences are first tested to better realize the NGM for piRNA detection and then the performance of *piRNAdetect* is compared with the K-mer scheme [97] as well as *piRPred* [103] based on the piRNAs from various species. To simulate *piRPred*, the locus information for the positive sample is referenced from piRBase database while random loci are assigned to the negative samples.

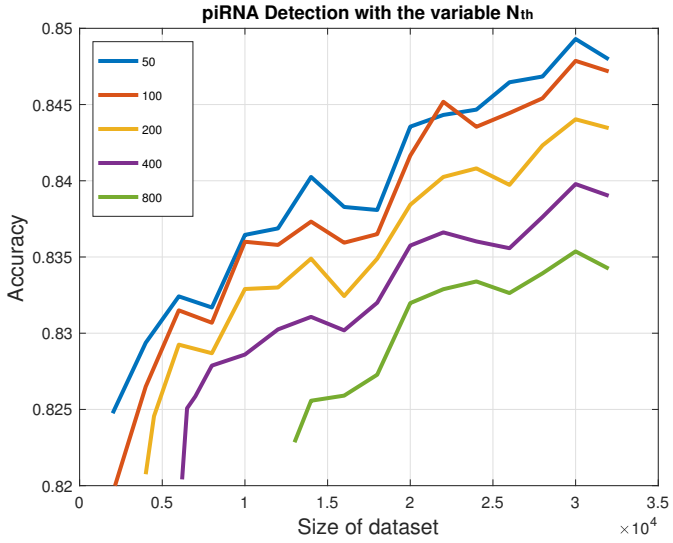
4.3.1 Evaluating the Effectiveness of NGMs for Detecting piRNAs

The piRNAs from *H. sapiens* with a total number of 32,826 sequences in the piRBase database are first tested for the parameters in NGMs. In order to test the effect of the parameters Z_{th} and N_{th} in the NGMs for piRNA detection with the different size of the test datasets, one parameter is taken as a control variable and the other parameter is varied to check the corresponding accuracy of piRNA detection. Besides, the sizes of the test dataset used for 5-fold CV are ranged from 2,000 to 32,000 with a step size 2,000.

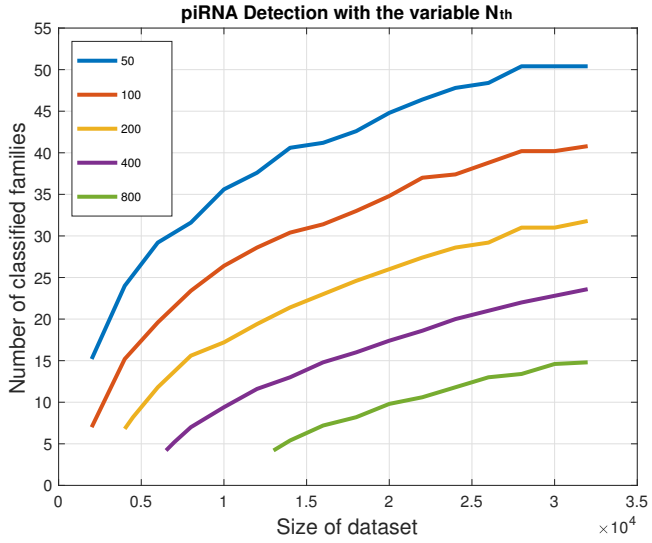
For the case with the fixed parameter $Z_{th} = 1.5$, Figure 4.1 illustrates the accuracy and the average number of classified family with respect to the variable parameter N_{th} and the sizes of the dataset. The sequence classification needs the size of the dataset large enough to build the NGMs, and hence the classification with smaller N_{th} can build the NGMs easier and detect piRNAs in a smaller dataset. Moreover, when the size of the dataset increases, it can build more NGMs with the corresponding classified families and become more accurate in the detection since more motif patterns are recognized. In this case with piRNAs from *H. sapiens*, the piRNA detection with the parameter $N_{th} = 50$ has the highest possible accuracy. However, it also builds the maximum amount of the NGMs with the parameter $N_{th} = 50$ and the computational complexity is proportional to the amount of NGMs in both training and detection.

For the case with fixed parameter $N_{th} = 200$, Figure 4.2 illustrates the accuracy and the average number of the classified family with respect to the variable parameter Z_{th} and the sizes of datasets. The sequence classification with a higher threshold Z_{th} needs a larger dataset to build NGMs. With the size of the dataset large enough, the detection with a higher threshold Z_{th} can build more elaborate NGMs to characterize piRNAs and better improve the detection accuracy. However, the extremely high

threshold Z_{th} can degrade the accuracy, and the piRNA detection with the parameter $Z_{th} = 2.0$ has the highest possible accuracy in this test case.

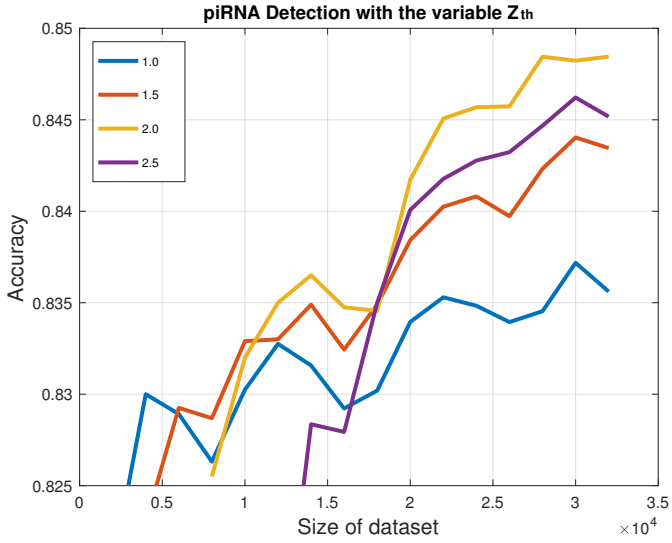


(a) Accuracy of piRNA detection

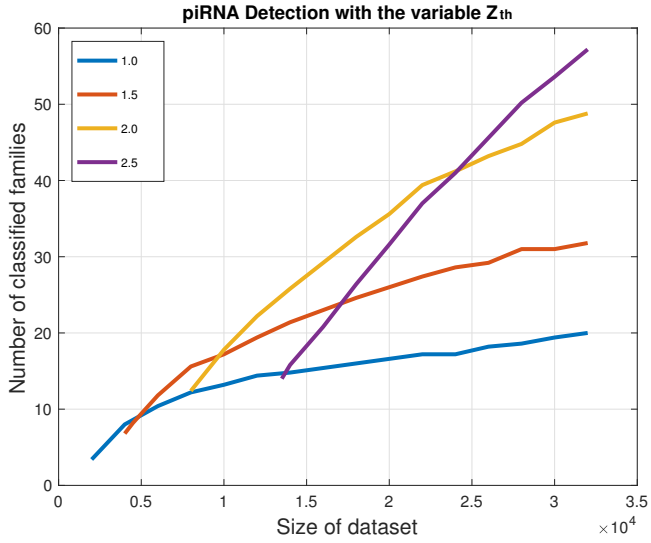


(b) Average number of classified families

Figure 4.1: The piRNA detection accuracy and the average number of classified families for $Z_{th} = 1.5$. (a) The prediction accuracy is shown on the y-axis and the dataset size is shown on the x-axis. Lines in different colors correspond to different values of N_{th}. (b) The average number of classified families for different N_{th} and dataset size.



(a) Accuracy of piRNA detection



(b) Average number of classified families

Figure 4.2: The piRNA detection accuracy and the average number of classified families for $N_{th} = 200$. (a) The prediction accuracy is shown on the y-axis and the dataset size is shown on the x-axis. Lines in different colors correspond to different values of Z_{th}. (b) The average number of classified families for different Z_{th} and dataset size.

4.3.2 Performance Evaluation of piRNAdetect

To assess the piRNA detection performance of the proposed *piRNAdetect* algorithm, we perform 5-fold CV on the piRNAs from the species *H. sapiens*, *R. norvegicus*, and *M. musculus*. Moreover, the numbers of sequences for each species are listed in Table 4.1. We randomly drew 30,000 sequences from each species as the positive samples for the test datasets.

Table 4.1: Dataset size for each species.

Species	Size
H. sapiens	32,826
R. norvegicus	63,182
M. musculus	51,664,769

In the following analysis, *piRNAdetect* utilizes the threshold parameters (N_{th}, Z_{th})= (200, 1.5) to balance the performance and computational complexity. For performance comparison, the K-mer scheme [97] and *piRPred* [103] are also evaluated on the same test datasets. Table 4.2 summarizes the performance of piRNA detection by *piRNAdetect*, *piRPred* with default settings, and K-mer scheme with the cutoff parameter t=1.2 [97]. The accuracy of *piRNAdetect* for piRNA detection outperforms K-mer scheme and *piRPred* in all three distinct species. The *piRPred* algorithm uses loci information for piRNA detection and it may need a large dataset to make accurate predictions, as prediction schemes that utilize clustering locus typically require a large number of sequence reads to identify clusters.

Table 4.2: Prediction accuracy of *piRNAdetect* compared against the K-mer scheme and *piRPred*.

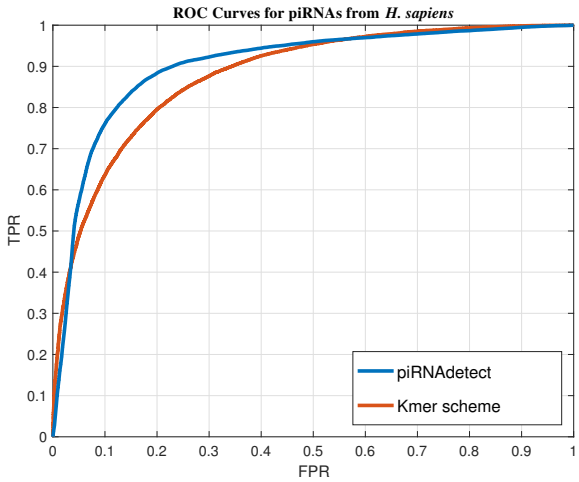
Method	***H. sapiens***			***R. norvegicus***			***M. musculus***		
	TPR	FPR	ACC (%)	TPR	FPR	ACC (%)	TPR	FPR	ACC (%)
piRNAdetect	**0.848**	**0.160**	**84.40**	**0.837**	**0.195**	**82.11**	**0.806**	**0.213**	**79.65**
K-mer scheme	0.821	0.226	79.76	0.781	0.222	77.95	0.698	0.259	71.95
piRPred	0.375	0.098	63.85	0.290	0.201	54.42	0.208	0.020	59.39

Since the cutoff parameter is introduced in the K-mer scheme to adjust the threshold in the decision, the receiver operating characteristic (ROC) curves for three species are also demonstrated in Figure 4.3. Please note that the ROC curve for *piRPred* is not shown in the figure, as *piRPred* does not assign confidence probabilities to the predictions it makes. For comparisons based on ROC curves, the area under curve (AUC) can be used as a useful overall performance measure [115, 116], where a larger AUC indicates superior prediction performance. As summarized in Table 4.3, *piRNAdetect* clearly outperforms the K-mer scheme based on AUC.

In general, the performance of piRNA detection depends on the characteristics of the training dataset and the prediction model that is constructed. For a sequence-based approach, the prediction method can achieve good performance if the sequences are regular and the dataset is large enough to be representative for all sequences. The K-mer scheme checks all possible sub-sequences with length $L \leq 5$ and extracts a total of 1,364 features to detect piRNAs. In comparison, *piRNAdetect* can practically check longer sub-sequences while extracting a smaller number of useful features by utilizing NGMs. However, NGMs rely on the shared sequence motifs in the training dataset, hence their effectiveness will degrade if significant sequence motifs are absent or the dataset is not large enough to extract the representative sequence motifs. In this work, *piRNAdetect* extracts and utilizes less than 50 features based on NGMs for predicting piRNAs in *H. sapiens*, *R. norvegicus*, and *M. musculus*.

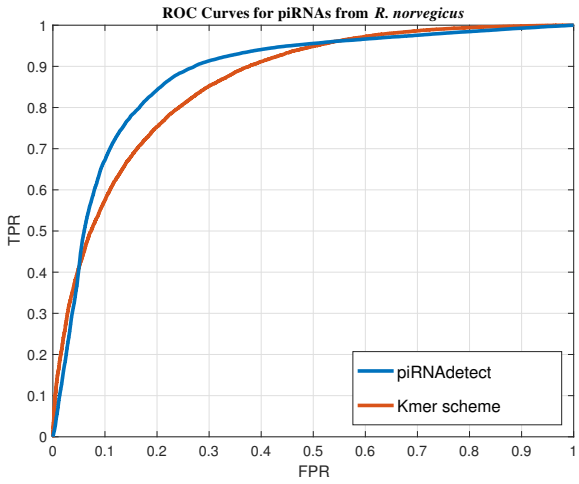
Table 4.3: Prediction performance based on average AUC.

	Average AUC		
species	*H. sapiens*	*R. norvegicus*	*M. musculus*
piRNAdetect	**90.28**	**88.15**	**85.97**
K-mer scheme	87.84	86.06	79.36

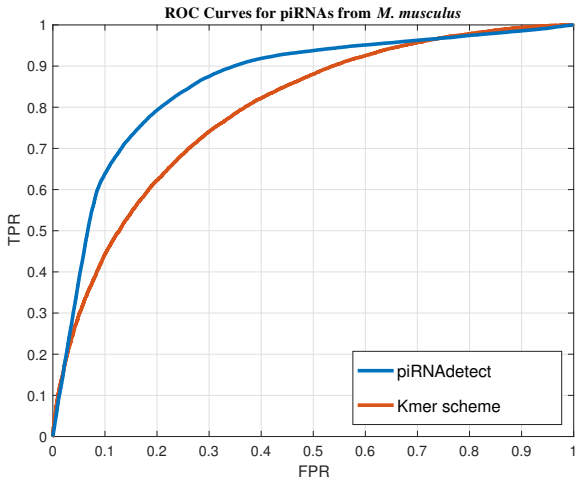


(a) ROC curve for detecting piRNAs in *H. sapiens*.

Figure 4.3: ROC curves showing the prediction performance of *piRNAdetect* and the performance of the K-mer scheme. (a) The performance for predicting piRNAs in *H. sapiens*. The false positive rate (FPR) is shown on the x-axis and the true positive rate (TPR) is shown on the y-axis.



(b) ROC curve for detecting piRNAs in *R. norvegicus*.



(c) ROC curve for detecting piRNAs in *M. musculus*.

Figure 4.3: (Continued) ROC curves showing the prediction performance of *piRNAdetect* and the performance of the K-mer scheme. (b) The prediction performance for piRNAs in *R. norvegicus*. (c) The prediction performance for piRNAs in *M. musculus*. The false positive rate (FPR) is shown on the x-axis and the true positive rate (TPR) is shown on the y-axis.

4.4 Conclusions

The piRNAs lack conserved characteristics and prominent features that could be used for recognizing them, which makes accurate prediction of piRNAs challenging. In this study, we proposed *piRNAdetect*, a novel algorithm for computational prediction of piRNAs. The proposed algorithm uses n-gram models (NGMs) to extract predictive sequence features for effective prediction of piRNAs. Besides, unlike *piRPred*, which is specifically designed for *Drosophila* and human data, our approach can be applied to identify sequences with shared sequence motifs for any given species. Comprehensive performance evaluation based on piRNAs in the piRBase database showed that *piRNAdetect* clearly outperforms the K-mer scheme, which is also a sequence-based scheme. Furthermore, despite the improved prediction accuracy, *piRNAdetect* utilizes a significantly smaller number of features compared to the K-mer scheme, which makes *piRNAdetect* more efficient and less prone to overtraining.

5. EFFICIENT COMPUTATIONAL DETECTION OF NOVEL NONCODING RNAS

5.1 Introduction

Although noncoding RNAs (ncRNAs) are not translated into proteins, many of them have been found to play important roles in diverse cellular processes such as transcriptional and post-transcriptional regulation, chromosome replication, RNA processing and modification, and protein degradation and translocation [6, 7]. It is not clearly known how many ncRNAs exist even in well-studied model organisms, but studies indicate that only a small fraction of ncRNAs may have been identified to date and that a much larger number of ncRNAs may be awaiting future discovery and investigation [5, 117, 118]. However, unlike coding genes that can be recognized by various features – *e.g.*, start/end codons, open reading frames (ORFs), composition bias – ncRNAs typically lack distinctive sequence features, making computational identification challenging. In fact, most ncRNAs are better conserved in terms of structure compared to their primary sequence [119, 120], hence it is difficult to identify ncRNAs through sequence-based methods. However, it has been also reported that a structure-based approach may not be sufficient by itself to identify ncRNAs, even though structural ncRNAs are expected to have secondary structures with higher thermodynamic stability [121]. Fortunately, comparative sequence analysis can help shed light on the detection of novel ncRNAs when coupled with a structure-based approach [122–124].

Through comparative genome analysis, several new RNA species have been found in a bacterial genome [122, 125, 126]. QRNA is one of the first approaches that detect ncRNAs through comparative sequence analysis [122, 127]. But QRNA fails

to identify ncRNAs without significant structure and the method has a relatively high false positive detection rate [128, 129]. To improve the detection accuracy, *RNAz* [129, 130], a widely used ncRNA detection package, utilizes a machine learning approach based on the thermodynamic stability of the secondary structure and the structural conservation between multiple aligned sequences. With the availability of an increasing number of sequenced genomes, more recent packages – including *RNAz* 2.0 updated with the dinucleotide models [130] and *Multifind* [131] – overcome the previous limitation on how many comparative sequences could be jointly analyzed for ncRNA detection and they also exploit additional statistical features to further enhance the detection performance.

In this study, we propose a new computational method for novel ncRNA detection called *RNAdetect*. In addition to features such as the minimum free energy (MFE) for thermodynamic stability and structural conservation index (SCI) that were shown to be useful in existing methods, *RNAdetect* incorporates novel features based on the n-gram model (NGM) and the concept of generalized ensemble defect (GED) to further enhance the ncRNA prediction accuracy. The GED metric provides an innovative feature that evaluates the conformation of a given structure to an ensemble of other structures, and utilization of the NGM enhances the assessment of sequence homology across the genome sequences that are jointly analyzed to detect conserved ncRNAs. *RNAdetect*, proposed in this study, extracts sequence-based and structure-based features that capture critical information across a set of comparative genome sequences and incorporate the features in a predictive model that uses the support vector machine (SVM) to detect novel ncRNAs. *RNAdetect* does not have any restriction on the number of sequences for comparative analysis, which can lead to further performance enhancement as a larger number of related sequences become available for joint analysis. We compare *RNAdetect* with other leading ncRNA

identification algorithms – *RNAz*, *RNAz* 2.0, and *Multifind* – based on benchmarks built from the Rfam database [113, 114], and the genomes of *Escherichia coli* [132] and *Streptomyces coelicolor* [133]. The performance assessment results demonstrate the efficiency and the accuracy of *RNAdetect*, which clearly outperform the existing state-of-the-art detection methods.

5.2 Materials and Methods

The overall structure of *RNAdetect* is shown in Figure 5.1. *RNAdetect* first screens the input of aligned genome sequences with a sliding window and calculates the various features needed for ncRNA detection by analyzing the presence of thermodynamically stable secondary structure in the given sequences and assessing the degree of structure conservation and sequence homology across the sequences. *RNAdetect* utilizes the packages ViennaRNA [134] and RNAstructure [135] for analyzing the input sequences and uses the LIBSVM package [112] to implement the SVM based on the calculated features. *RNAdetect* identities ncRNAs and estimates the corresponding probabilities using the constructed SVM.

5.2.1 Features for ncRNA Detection

For ncRNAs, their structure is often better conserved than their primary sequence, hence structural properties provide important clues for identifying ncRNAs. The SCI, which evaluates structural conservation, has been commonly utilized as a feature by ncRNA detection methods, but the effectiveness of SCI is affected by the sequence identity of the input sequences as well as the quality of the sequence alignment. To address this issue, we introduce a new measure based on the concept of GED to evaluate structure conservation. As will be shown later, GED analyzes the consensus structure between sequences based on their alignment probability and thereby improves the identification of structural ncRNAs. Noncoding RNAs with

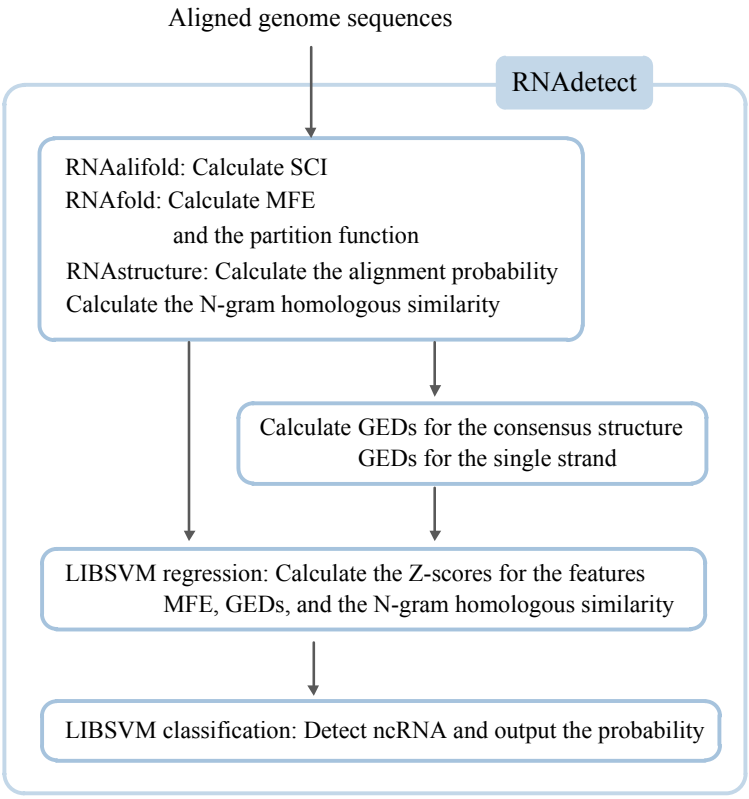


Figure 5.1: Schematic overview of *RNAdetect* for novel ncRNA detection. The aligned genome sequences are screened using a sliding window to detect ncRNAs. *RNAdetect* calculates the features MFE, SCI, GEDs, and NGM through the comparative analysis of the input sequences. Based on the Z-scores of the extracted features, *RNAdetect* predicts potential ncRNAs and estimates their confidence probabilities using an SVM.

sparse folding structure (*e.g.*, U6 snRNAs and C/D box snoRNAs) are in general not easy to detect using structure-based approaches [136, 137]. To accurately identify various types of ncRNAs, including those with sparse structure, the following features are incorporated in *RNAdetect*: average Z-score of MFE, SCI, average Z-score of the minimum GEDs, and the maximum Z-score of the sequence homology measured using NGMs. In the following subsections, each of these features used in *RNAdetect* is discussed in further details.

5.2.1.1 MFE and SCI

- To detect novel ncRNAs, *RNAdetect* utilizes RNA structure prediction methods to extract characteristic structural features commonly observed in many ncRNAs. In RNA secondary structure prediction, the minimum free energy (MFE) is an important measure to compute, as it reflects the thermodynamic stability of the RNA secondary structure [138, 139]. Although the MFE alone may not be sufficient for distinguishing ncRNAs from the genomic background in a single sequence [121, 140], the discriminative power can be significantly improved when used in a comparative setting where related sequences are jointly analyzed [122–124]. In order to remove the effect of base composition, *RNAdetect* computes the Z-score of the MFE as follows:

$$Z(E_i, q_i) = \frac{E_i - \mu}{\sigma}, \tag{5.1}$$

where E_i is MFE of a single sequence q_i, and the parameters μ and σ are the average and the standard deviation of the MFE computed based on random sequences with the same base composition as q_i. The MFE is calculated by RNAfold in the ViennaRNA package [134, 141], and the Z-score of the MFE is estimated as described by Washietl et al. [129]. First, the statistical averages and the standard deviations of the MFE are computed based on 10,648 randomly generated sequences obtained from Monte Carlo shuffling simulations [111]. The lengths of the test sequences range from 50 to 150 nucleotides with a step size of 50. The base composition ratios GC/AT, G/GC and A/AU of the test sequences range from 25% to 75% with a step size of 5%. Finally, the Z-score of the MFE is estimated by SVM regression analysis based on the statistical averages and the standard deviations.

Furthermore, the structural conservation index (SCI) is computed to estimate the

structural conservation across the aligned sequences. The SCI is defined as:

$$SCI = E_{cons}/E_{single}, \tag{5.2}$$

where E_{cons} is the consensus MFE for the sequence alignment and E_{single} is the average MFE of the sequences in the alignment. The SCI will be high (SCI $\simeq 1$) if the sequences in the alignment can fold into a stable common structure. Otherwise, the SCI will be low (SCI $\simeq 0$). The consensus MFE for a given sequence alignment can be estimated using RNAalifold in the ViennaRNA package [134, 142]. However, the effectiveness of SCI for ncRNA prediction is known to be affected by the quality of the sequence alignment. The GED proposed in the following subsection aims to address this shortcoming of SCI and complement it to improve ncRNA detection performance.

5.2.1.2 Generalized Ensemble Defect

- Since RNAs do not necessarily fold into the most stable structure predicted by thermodynamical models, an ensemble-based analysis can be useful for RNA detection rather than considering only the single most stable structure [131, 143, 144]. The *ensemble defect* of a given RNA sequence measures the average distance from one structure to an ensemble of secondary structures [145, 146]. We can measure the distance between two RNA secondary structures (s_1, s_2) based the discrepancy between the base pair conformations as follows:

$$d(s_1, s_2) = N - \sum_{\substack{1 \le i \le N \\ 1 \le j \le N+1}} S_{i,j}(s_1) S_{i,j}(s_2), \tag{5.3}$$

where N is the sequence length, and $S_{i,j}(s_1) \in \{0,1\}$ are entries of the *structure matrix* $S(s_1)$ that describes the base pairs in the structure s_1. For $1 \leq j \leq N$, we have $S_{i,j}(s_1) = 1$ if the structure s_1 contains the base pair (i,j), and $S_{i,j}(s_1) = 0$ otherwise. For the case $j = N+1$, $S_{i,N+1}(s_1) = 1$ if base-i is unpaired, otherwise we have $S_{i,N+1}(s_1) = 0$. Based on this distance metric over structures, the ensemble defect for a given structure s can be defined as:

$$\begin{aligned} n(s;\Omega) &= \sum_{\sigma \in \Omega} p(\sigma;\Omega) d(s,\sigma) \\ &= N - \sum_{\substack{1 \leq i \leq N \\ 1 \leq j \leq N+1}} S_{i,j}(s) P_S(i,j), \end{aligned} \tag{5.4}$$

where Ω is the *structure ensemble* that consists of the potential structures for a given RNA sequence, and $p(\sigma;\Omega)$ is the probability of the structure σ in the ensemble Ω. The structure probability $P_S(i,j)$ denotes the equilibrium probability of the base pair (i,j) for $1 \leq j \leq N$. For $j = N+1$, $P_S(i,j)$ is the equilibrium probability that base-i will remain unpaired in the structure ensemble Ω [145, 147]. The ensemble defect $n(s;\Omega)$ estimates how many bases in structure s are structurally different from the stable structures in the ensemble Ω on average.

In this work, we further generalize the concept of ensemble defect to make it more effective for ncRNA detection. For this purpose, we incorporate a *loop scale parameter* α (≥ 0) into the structure matrix, which is to scale the distance of unmatched loops between structures as follows:

$$S^{(1)}_{i,j}(s_1) = \begin{cases} 1, & \text{if } 1 \leq j \leq N, \text{ and base (i,j)} \in \mathcal{P} \\ \sqrt{\alpha}, & \text{if } j = N+1, \text{ and base-}i \in \mathcal{L} \\ 0, & \text{otherwise,} \end{cases} \tag{5.5}$$

where $\mathcal{P}$ is the index set of the paired bases and $\mathcal{L}$ is the index set of the unpaired loops in structure s_1. Furthermore, we generalize the structure matrix so that it can be used for the joint analysis of multiple related sequences that may potentially share a common consensus structure. Given a reference sequence with length N and a set of related sequences $\{q_m\}$, the entries of the *consensus structure matrix* based on the structure s_m for the sequence q_m with length N_m are defined as $S^{(m)}_{i,j}(s_m) = \sum_{1\le k,l\le N_m} P_A(i,k)P_A(j,l)S_{k,l}(s_m)$ for $1 \le j \le N$, and $S^{(m)}_{i,N+1}(s_m) = \sum_{1\le k\le N_m} P_A(i,k)\sqrt{\alpha}S_{k,N_m+1}(s_m)$, where $P_A(i,k)$ is the alignment probability between base-i of the reference sequence and base-k of sequence q_m. The generalized consensus structure matrix $S^{(m)}(s_m)$ denotes the potential consensus structure of the reference sequence inferred from structure s_m. Subsequently, by adopting the generalized consensus structure matrix, we can generalize the concept of ensemble defect of a given structure s_1 for a set of multiple related sequences.

Given M sequences with the structure ensemble set $\Omega^M = \bigcup_{n=1}^{M} \Omega_n$, the generalized ensemble defect (GED) is defined as

$$\begin{aligned} n(s_1;\Omega^M) = & \frac{1}{N}\sum_{n=1}^{M}\Big\{w_n \times \sum_{\sigma_n\in\Omega_n} p(\sigma_n;\Omega_n)d(s_1,\sigma_n)\Big\} \\ = & \quad 1 - \frac{1}{N}\sum_{\substack{1\le i\le N \\ 1\le j\le N+1}} S^{(1)}_{i,j}(s_1)P_{cons}(i,j), \end{aligned} \tag{5.6}$$

where w_n is the non-negative weight parameter for the distance metric according to the structure ensemble for sequence q_n such that $\sum_{n=1}^{M} w_n = 1$. Note that the GED defined in (5.6) is normalized by the length of the reference sequence.

The *consensus structure score* is defined as

$$P_{cons}(i,j) = \begin{cases} \sum_{n=1}^{M} \Big\{ w_n \times \sum_{1 \le k,l \le N_n} P_A(i,k) P_A(j,l) P_S(k,l) \Big\}, \\ \qquad\qquad \text{if } 1 \le j \le N, \\ \sum_{n=1}^{M} \Big\{ w_n \times \sum_{1 \le k \le N_n} P_A(i,k) \sqrt{\alpha} P_S(k, N_n + 1) \Big\}, \\ \qquad\qquad \text{if } j = N + 1. \end{cases}$$

For the case when only one sequence is given, the structure with the minimum GED is equivalent to the maximum expected accuracy (MEA) structure with respect to the loop scale parameter [148, 149]. The MEA-based approach was shown to outperform the traditional MFE-based approach in some cases [148, 150], where the main strength of MEA-based prediction lies in the fact that it considers all potential base pairs in the structure ensemble rather than predicting a single best structure based on thermodynamic stability.

The Z-scores of the minimum GED for a single sequence with the loop scale parameters $\alpha = \{0.5, 1.0, 2.0\}$ are used as features in *RNAdetect*, where the loop scale parameters are selected based on the Pearson correlation and the F-score [151, 152] in order to improve the performance and reduce the complexity of the SVM classification. The structure probabilities can be estimated and the structure with the minimum GED can be predicted using RNAfold in the ViennaRNA package [134]. In order to also consider the consensus structure of the multiple related input sequences for detecting ncRNAs, the Z-scores of the minimum GED for the given set of sequences are computed with the loop scale parameters $\alpha = \{0.5, 1.0, 2.0, 4.0\}$ and the resulting Z-scores are used as features in *RNAdetect*. As before, the parameters are selected according to the Pearson correlation and the F-score. Given M sequences, the alignment probabilities are first estimated using the RNAstructure package [135], and

each sequence in the set is used as a reference sequence to evaluate the average Z-score of the minimum GED for the consensus structure. Since the GED features separately evaluate the ensemble defects for individual sequences and multiple sequences, during the process of evaluating the consensus structure, the reference sequence is excluded by setting the corresponding weight to 0 and then using a uniform weight $1/(M-1)$ for the other sequences.

5.2.1.3 N-gram Model

- Although the secondary structure is often better conserved than the primary sequence for many ncRNAs, there are ncRNAs whose structure is sparse and mainly consist of unpaired bases, which are difficult to detect using structure-based approaches [137, 153]. To improve the detection performance for such ncRNAs, *RNAdetect* utilizes the n-gram model (NGM) to incorporate additional features based on sequence homology. NGM is widely used in a variety of domains, including language analysis [154], protein classification [105, 155], and genome sequence analysis [108]. The NGM provides a simple yet effective way of statistically evaluating the similarity between nucleotide sequences with tolerance for insertions, deletions, and substitutions [108]. Since the NGM is essentially an $(n-1)$th-order Markov model, the occurrence probability of the nucleotide at a given sequence position only depends on the last $(n-1)$ nucleotides. Based on NGM, the log-likelihood of a substring of length L in sequence $\underline{b}$ can be computed by:

$$R(\underline{b}, k) = \log P(b_{k+1,k+n-1}) + \sum_{i=k+n}^{k+L} \log P(b_i | b_{i-n+1,i-1}), \tag{5.7}$$

where k is the offset of the substring in the sequence, b_i represents the i^{th} base in sequence $\underline{b}$, and $b_{i,j}$ represents the substring $(b_i, b_{i+1}, \cdots, b_j)$ in $\underline{b}$. The log-likelihood

$R(\underline{b}, k+1)$ can be efficiently updated from $R(\underline{b}, k)$ when scanning the sequence $\underline{b}$ to search for potential sequence homology.

In order to learn the NGMs to be used in *RNAdetect*, we selected 116 families with high average Z-scores of MFE from Rfam database 12.1 [113, 114]. For each ncRNA family, an NGM was constructed by estimating the probabilities based on the sequences in the given family, where pseudo-counts were added to account for potential variations not observed in the training data. In this work, hexagrams were used to have high sensitivity and reasonable computational complexity. Based on each NGM, the maximum $R(\underline{b}, k)$ is used as the sequence homology score, where the substring length L was set to the minimum sequence length within the corresponding ncRNA family. The Z-score is computed from $R(\underline{b}, k)$ to reduce unwanted bias, and *RNAdetect* incorporates the maximum Z-score as an additional feature to utilize sequence homology for detecting ncRNAs.

5.2.2 Implementation for Classification

In order to train the SVM classifier for ncRNA detection, sequences have been drawn from the Rfam database 12.1 [113, 114] to construct datasets with positive samples and negative samples for training and assessment. RNA Families that contain over 25 members and whose average length is less than 400 bases were selected in the dataset, including 396 families in total. Sequence sets composed of 3 to 6 sequences (that do not contain unknown bases) were randomly drawn from each family. In total, the constructed benchmark contains 44,096 sequence sets and 198,432 RNA sequences. The sequences in each test set were aligned using ClustalW [156] to obtain positive samples. The negative samples were obtained by randomly shuffling the aligned sequences following a similar strategy proposed before [123]. Using the LIBSVM package [112], a c-support vector classification (c-SVC) model with the radial basis

function (RBF) kernel was trained to detect potential ncRNAs and compute the confidence probability for the prediction.

5.2.3 Computational Complexity

Here, we briefly analyze the computational complexity of *RNAdetect*. Given M aligned sequences of length N, the computational complexity for computing MFE and SCI using RNAfold and RNAalifold requires is $O(MN^3)$ [141, 142]. Furthermore, the time complexity for calculating the minimum GED is $O(MN^3)$, as the computation time is dominated by the partition function calculation step [148]. The computational cost for computing the maximum similarity score based on K NGMs for different RNA families is $O(KN)$, since the similarity for each family can be iteratively calculated in linear time. Although training the SVM can be time-consuming and depends on the size of the training data, the computational complexity for the SVM regression and classification is proportional to the number of features [157].

5.3 Results and Discussion

We first evaluate the effectiveness of the novel GED and NGM features for ncRNA detection. Next, we assess the performance of *RNAdetect* based on the benchmark constructed from the Rfam database and compare the proposed method with the current state-of-the-art methods. Finally, we evaluate the efficacy of *RNAdetect* for detecting known ncRNAs buried in genomes, by using a comprehensive benchmark constructed from *E. coli* and *S. coelicolor* genomes.

5.3.1 Effectiveness of GED and NGM Features

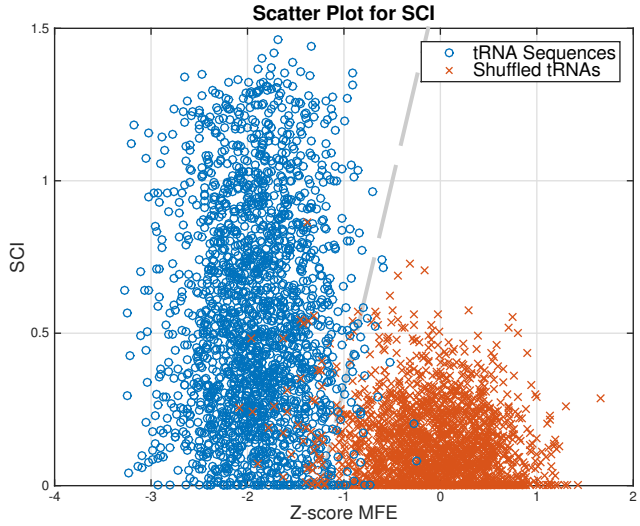
In this section, we proposed a novel predictive feature for ncRNA detection based on the concept of generalized ensemble defect. To test the efficacy of this new feature when combined with the MFE feature commonly used in existing RNA detection

methods, we performed the following simulations. For evaluation, we constructed 2,000 sequence sets, where each set is composed of 4 sequences that are randomly drawn from the tRNA family in the Rfam database. Every sequence set was aligned using ClustalW [156] to obtain positive samples and negative samples were obtained by shuffling the sequence alignment. Figure 5.2(a) shows the scatter plot based on MFE and SCI, the two features that are used for ncRNA prediction in existing methods. In comparison, Figure 5.2(b) shows the scatter plot for the case when SCI is replaced by the new GED feature (loop scale parameter set to $\alpha = 0.5$). Figure 5.2 clearly shows that the GED feature can separate the positive samples from the negative samples much more effectively compared to the traditional SCI feature, when combined with MFE.

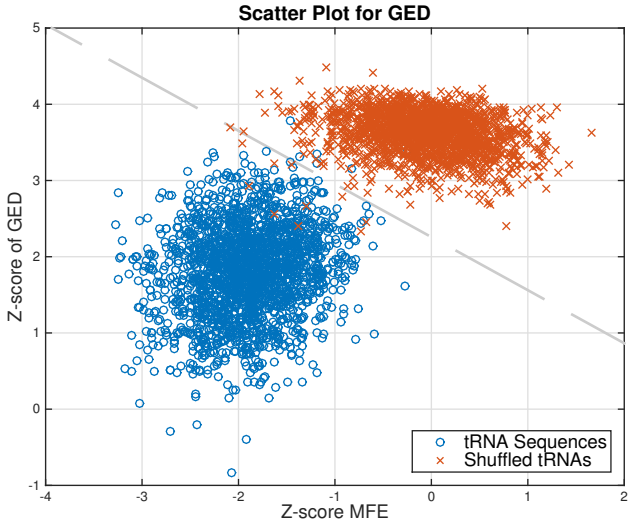
Table 5.1: Prediction accuracy of the SCI-based classifier and GED-based classifier.

		GED		
	SCI	α =0.5	α =1.0	α =1.5
TPR	0.976	0.997	0.995	0.986
FPR	0.029	0.006	0.010	0.016
Accuracy(%)	97.33	99.58	99.25	98.50

We further compared the effectiveness of SCI and GED by building linear SVM classifiers optimized by the c-SVC model (with the cost parameter set to be one) [112, 158] and performing classification experiments. The ncRNA detection performance was assessed in terms of the true positive rate (TPR)=$\frac{\text{TP}}{\text{TP+FN}}$ and the false positive rate (FPR)=$\frac{\text{FP}}{\text{TN+FP}}$. TP denotes the number of correctly identified ncRNAs, FP denotes the number of negative samples incorrectly classified as ncRNAs, and FN



(a) Scatter plot for SCI



(b) Scatter plot for GED with $\alpha = 0.5$

Figure 5.2: Scatter plots that compare the effectiveness of SCI (a) and GED (b) for separating true ncRNAs from randomized negative samples, when combined with MFE.

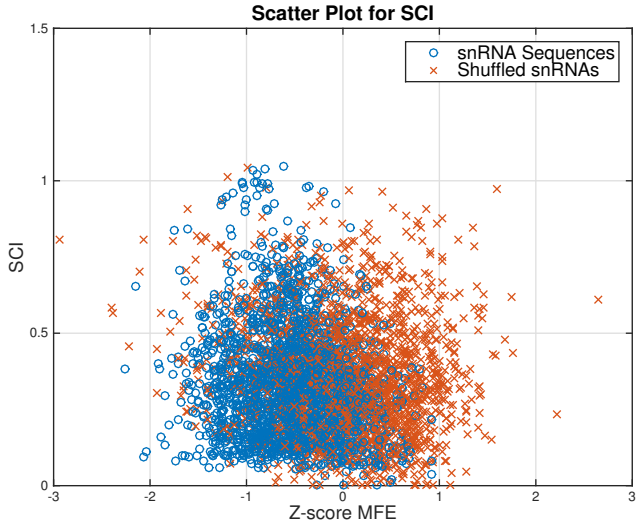
denotes the number of ncRNAs that are missed. The results are summarized in Table 5.1, which compares the prediction performance of the SCI-based classifier

and that of the GED-based classifier for three different loop scale parameters $\alpha = 0.5, 1.0, 1.5$. As shown in the table, all three GED-based classifiers outperformed the SCI-based classifier. A smaller value for α encourages the formation of a larger number of base pairs in the RNA secondary structure predicted by MEA. Although the linear SVM classifier with a smaller α shows better prediction performance, incorporating GED obtained by using several different loop scale parameter values can improve the overall performance of ncRNA detection.

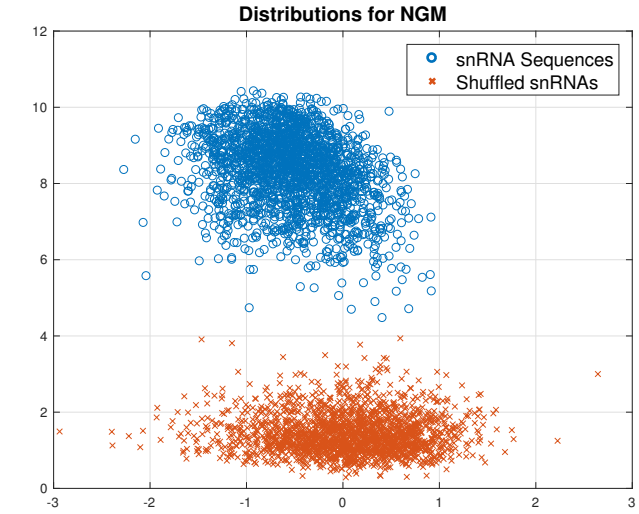
To evaluate the effectiveness of the NGM-based features for detecting ncRNAs with sparse folding structure, we again compared the prediction performance between NGM and SCI, when combined with MFE. For this experiment, we constructed 2,000 sequence sets, where each set is composed of 4 sequences randomly drawn from the U6 snRNA family in the Rfam database. As before, the sequences in each set were aligned to obtain positive samples and negative samples were obtained by random shuffling of the aligned sequences. Figure 5.3(a) shows the scatter plot based on SCI and MFE while Figure 5.3(b) shows the scatter plot based on NGM and MFE. As we can see from Figure 5.3, SCI may not be very useful for detecting ncRNAs with a sparse folding structure (such as U6 snRNAs), where many bases remain unpaired. Furthermore, the ncRNA detection performance with the different size of n-gram was evaluated by linear SVM classifiers optimized by the c-SVC model, and the results are summarized in Table 5.2. For such ncRNAs, incorporating NGM-based features may remarkably enhance the overall prediction accuracy.

5.3.2 Performance on the ncRNA Benchmark Constructed from Rfam Database

In order to assess the ncRNA detection performance of *RNAdetect*, we used the ncRNA benchmark that was constructed from the Rfam database as described before and performed 4-fold cross-validation (CV) experiments to estimate the prediction



(a) Scatter plot for SCI



(b) Scatter plot for NGM

Figure 5.3: Scatter plots that compare the effectiveness of SCI (a) and NGM (b) for separating true ncRNAs from randomized negative samples, when combined with MFE.

accuracy. In the CV experiments, the benchmark was randomly partitioned into four folds of identical size for each family, where three folds were used for training

Table 5.2: Prediction accuracy of the NGM-based classifier with the different size of n-gram.

	NGM			
n-gram	$n = 2$	$n = 3$	$n = 4$	$n = 5$
TPR	0.827	0.907	0.998	0.999
FPR	0.299	0.105	0.003	0.001
Accuracy(%)	76.40	90.12	99.78	99.95

Table 5.3: Performance evaluation on the ncRNA benchmark.

	Area under curve (AUC)	Log_{10}(Time)
RNAdetect	97.82	5.06
RNAz 1.0	90.18	3.83
RNAz 2.1	92.74	6.13
Multifind	86.51	6.89

the NGMs and the SVM and the remaining one fold was used for evaluation. For comparison, *RNAz* 1.0 [129], *RNAz* 2.0 [130], and *Multifind* [131] were also evaluated on the same benchmark. Figure 5.4 shows the receiver-operator characteristic (ROC) curves for the four methods. As shown in Figure 5.4, *RNAdetect* always shows higher TPR at a given FPR, clearly outperforming the existing methods. The AUC (area under ROC curve) is the largest for *RNAdetect*, reflecting the best overall prediction performance among the four methods.[1] Table 5.3 summarizes the AUC and the overall computation time (in seconds) for the respective ncRNA detection methods evaluated based on the ncRNA benchmark.

[1]Please note that *Multifind* skipped some of the test sets as they led to simulation errors.

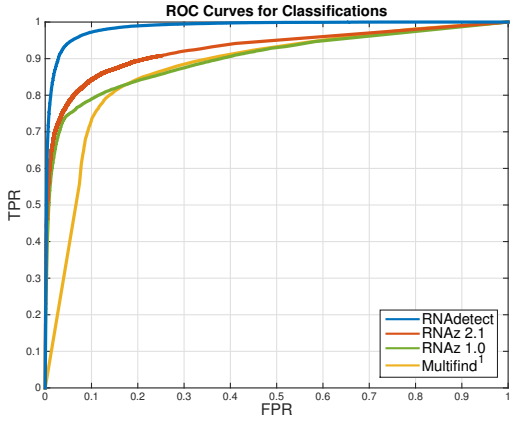


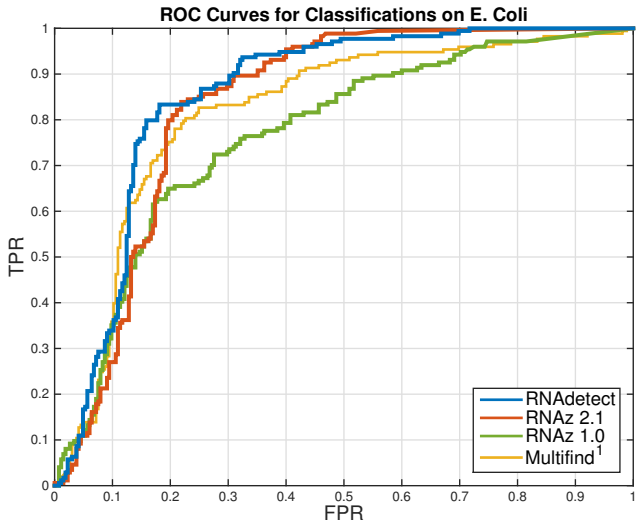
Figure 5.4: ROC curves that show the ncRNA classification performance on the ncRNA benchmark. *RNAdetect* clearly outperforms other existing ncRNA prediction methods.

Table 5.4: Performance evaluation based on the bacterial genome benchmark.

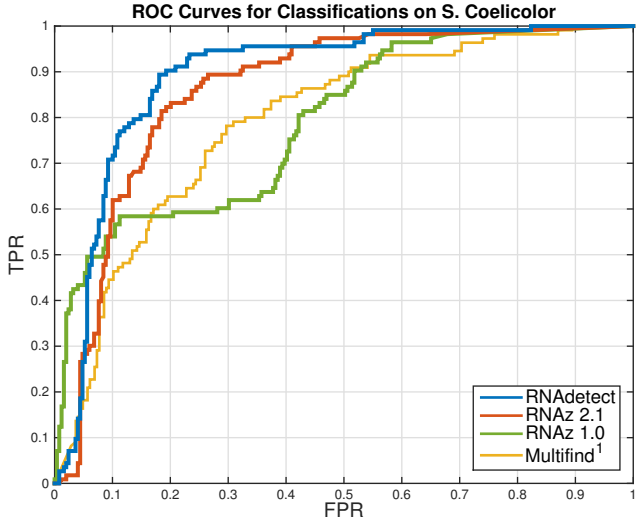
	E. coli		*S. coelicolor*	
	Area under curve	Log_{10}(Time)	Area under curve	Log_{10}(Time)
RNAdetect	83.83	2.79	88.76	2.66
RNAz 1.0	76.69	2.10	78.93	1.97
RNAz 2.1	83.32	2.76	86.07	3.03
Multifind	81.64	4.77	79.09	4.82

5.3.3 Predicting ncRNAs in Bacterial Genomes

To further evaluate the ncRNA detection performance of *RNAdetect*, we performed experiments to predict ncRNAs in the genomes of *Escherichia coli* (NCBI: NC000913) and *Streptomyces coelicolor* (NCBI: NC003888). In order to predict ncRNAs in *E. coli*, we first aligned the *E. coli* genome with the genomes of *Klebsiella pneumoniae* (NCBI: NC011283), *Salmonella paratyphi* (NCBI: NC011147), *Salmonella typhi* (NCBI: NC004631), and *Shigella boydii* (NCBI: NC010658. For this purpose, we used the multiple genome



(a) ROC curves for ncRNA detection in *E. Coli*.



(b) ROC curves for ncRNA detection in *S. Coelicolor*.

Figure 5.5: ROC curves that show the ncRNA detection performance on the bacterial genome benchmark.

alignment algorithm called progressiveMauve [159]. Similarly, to predict ncRNAs in *S. coelicolor*, we aligned the genome of *S. coelicolor* with those of *Streptomyces avermitilis* (NCBI: NC003155), *Streptomyces griseus* (NCBI: NC010572), *Streptomyces*

scabiei (NCBI: NC013929), and *Streptomyces venezuelae* (NCBI: NC018750). Although not all ncRNA loci are known for these genomes, we collected recognized ncRNA loci from the Ensembl Genomes database[160] and the related papers [161, 162] so that we can use them as a reference for performance evaluation. By taking segments of the genome alignment that contain known ncRNAs, we built a genome benchmark that consists of 113 genome alignment segments, where each segment is at least 450 bases in length. Aligned segments that contain unknown bases or segments that consist of fewer than 3 sequences were not included in the benchmark.

For performance evaluation, we used *RNAdetect* to screen the sequence alignments in the genome benchmark using a window of length 150 bases and sliding it by 50 bases at a time. The size of the sliding window was chosen such that it is long enough to detect local structures, but not overly so to avoid excessive inclusion of irrelevant flanking sequences [163]. To avoid ambiguity in detection, windows that partially overlap with any ncRNA were excluded. Figure 5.5(a) shows the ROC curve for ncRNA prediction in *E. coli* and Figure 5.5(b) shows the ROC curve for ncRNA prediction in *S. coelicolor*. In both plots, we can see that *RNAdetect* generally outperforms the other three methods for a wide range of FRP, resulting in the largest AUC (also see Table 5.4). *Multifind*, which incorporates additional statistical features, performs better than *RNAz* 1.0, while the latest version of *RNAz* (v2.1) generally outperforms both *RNAz* 1.0 and *Multifind*, especially when the FPR is not too low (*i.e.*, FPR $> 0.1 \sim 0.2$).

The overall prediction performance (measured in terms of AUC) and the total run time (in seconds) are summarized in Table 5.4 for the tested methods. The computation time was measured on an iMac (Intel Core i7 3.5GHz, 32 GB RAM, OS X 10.9.5). The performance evaluation results in Table 5.4 show that the proposed algorithm *RNAdetect* can accurately detect ncRNAs and is also computationally

efficient.

5.4 Conclusions

In this study, we presented a novel method, *RNAdetect*, for efficient and reliable detection of novel ncRNAs in comparative genome sequences. To improve the overall detection performance, *RNAdetect* incorporates novel predictive features based on the concept of generalized ensemble defect (GED) and the n-gram models (NGM). The GED provides an effective way of assessing structure conservation and conformation to the consensus structure, while NGM can effectively capture sequence homology, which can be especially useful for detecting ncRNAs that have a sparse folding structure with many unpaired bases. Unlike *RNAz*, which limits the number of sequences in the input alignment, *RNAdetect* has virtually no restriction on the number of input sequences that can be jointly analyzed, hence allowing us to include a larger number of related genome sequences to improve the detection performance. Extensive performance evaluation based on the ncRNA benchmark constructed from RNA families in Rfam database and the genome benchmark constructed from bacterial genomes clearly show that *RNAdetect* outperforms other existing ncRNA detection methods in terms of prediction accuracy. Furthermore, *RNAdetect* is also computationally efficient, often outperforming existing methods in addition to its higher accuracy.

6. CONCLUSIONS AND FUTURE WORKS

6.1 Whole Genome Reconstruction and Cost Minimization

In the first part of our dissertation, we discuss the feasibility conditions and cost minimization strategy for the optimal hybrid sequencing and assembly. We derive the conditions for whole genome reconstruction from multiple read sources at a given confidence level and also introduce the optimal strategy for combining reads from different sources to minimize the overall sequencing cost. We show that the optimal read set, which simultaneously satisfies the feasibility conditions for genome reconstruction and minimizes the sequencing cost, can be effectively predicted through constrained discrete optimization. Through extensive evaluations based on several genomes and different read sets, we verify the derived feasibility conditions and demonstrate the performance of the proposed optimal hybrid sequencing and assembly strategy. In this work, we simplify the read model and focus on deriving feasible bounds and optimal sequencing strategies for complete genome reconstruction with error-free reads. With regard to the future work, the genome reconstruction and cost minimization strategy can be extended to the reads with errors and the paired-reads. The paired-reads provide a lower-cost alternative to bridge repetitive sequences in the sequencing and it can be seen as long reads with erasures in the analysis. In the presence of sequencing errors, the minimum coverage depth required for complete genome reconstruction is bound to increase, in order to effectively correct the errors for accurate assembly. Assembly feasibility conditions for hybrid reads with potential sequencing errors require further analysis in the future. However, the overall concept and strategy for optimal hybrid sequencing and assembly in our study can be carried over to the case when sequencing errors are present.

6.2 Efficient Computational Detection of Novel Noncoding RNAs

In the second part of our dissertation, we discuss the problem of computational detection of novel noncoding RNAs. For many RNA families, it is known that their RNA secondary structures are better conserved than the RNA sequences themselves. Hence the RNA structural alignment can be employed to explore related structured ncRNAs. In this work, we propose an innovate method, *TOPAS*, for RNA structural alignment through topological networks. The computational complexity of our proposed method is significantly lower than the dynamic programming approach, while resulting in favorable alignment results. Furthermore, the proposed method is not restricted to the nested structures, and hence it can effectively handle RNAs with pseudoknots. We demonstrate the performance of the proposed method through extensive evaluation and comparison with state-of-the-art methods based on benchmark RNA families.

In order to efficiently search for novel ncRNAs in genomes, we develop a new approach by utilizing the n-gram model to classify the sequences that share similar sequence motifs and extract effective features to capture sequence homology. In this study, we propose a novel method, *piRNAdetect*, for reliable computational prediction of piRNAs in genome sequences. We demonstrate the effectiveness of the proposed *piRNAdetect* algorithm through extensive performance evaluation based on piRNAs in different species and show that *piRNAdetect* outperforms the current advanced methods in terms of efficiency and accuracy. Moreover, we propose *RNAdetect*, a novel computational method for accurate detection of ncRNAs through efficient comparative genome analysis. *RNAdetect* enhances the accuracy of ncRNA detection by incorporating n-gram model and additional predictive features based on the concept of generalized ensemble defect, which assesses the degree of structure conservation

across multiple related sequences and the conformation of the individual folding structures to a common consensus structure. Extensive performance evaluation based on the Rfam database and bacterial genomes demonstrate that RNAdetect can accurately and reliably detect novel ncRNAs, outperforming the current up-to-date methods. In regard to the future work, it is possible to incorporate more elaborate models, like hidden Markov model, stochastic context-free grammar model [164, 165], with the similar approach to detect ncRNAs. Furthermore, the detection approach through comparative genome analysis can apply to eukaryotic genomes with multiple chromosomes and predict more complex ncRNAs for the further biological research.

REFERENCES

[1] K. Darty, A. Denise, Y. Ponty, VARNA: Interactive drawing and editing of the RNA secondary structure, Bioinformatics 25 (15) (2009) 1974–1975.

[2] O. T. Avery, C. M. MacLeod, M. McCarty, Studies on the chemical nature of the substance inducing transformation of Pneumococcal types, Journal of experimental medicine 79 (2) (1944) 137–158.

[3] F. Crick, On Protein Synthesis, Symposia of the Society for Experimental Biology (1958) 138–163.

[4] N. Delihas, Discovery and characterization of the first non-coding RNA that regulates gene expression, micF RNA: A historical perspective, World journal of biological chemistry 6 (4) (2015) 272.

[5] S. R. Eddy, Non–coding RNA genes and the modern RNA world, Nature Reviews Genetics 2 (12) (2001) 919–929.

[6] G. Storz, An expanding universe of noncoding RNAs, Science 296 (5571) (2002) 1260–1263.

[7] J. S. Mattick, I. V. Makunin, Non-coding RNA, Human molecular genetics 15 (suppl 1) (2006) R17–R29.

[8] S. M. Elbashir, J. Harborth, W. Lendeckel, A. Yalcin, et al., Duplexes of 21-nucleotide RNAs mediate RNA interference in cultured mammalian cells, nature 411 (6836) (2001) 494.

[9] D. P. Bartel, MicroRNAs: genomics, biogenesis, mechanism, and function, cell 116 (2) (2004) 281–297.

[10] E.-M. Weick, E. A. Miska, piRNAs: from biogenesis to function, Development 141 (18) (2014) 3458–3471.

[11] L. P. Lim, N. C. Lau, P. Garrett-Engele, A. Grimson, et al., Microarray analysis shows that some microRNAs downregulate large numbers of target mRNAs, Nature 433 (7027) (2005) 769.

[12] R. W. Carthew, E. J. Sontheimer, Origins and mechanisms of miRNAs and siRNAs, Cell 136 (4) (2009) 642–655.

[13] A. Mencía, S. Modamio-Høybjør, N. Redshaw, M. Morín, F. Mayo-Merino, L. Olavarrieta, L. A. Aguirre, I. del Castillo, K. P. Steel, T. Dalmay, et al., Mutations in the seed region of human miR-96 are responsible for nonsyndromic progressive hearing loss, Nature genetics 41 (5) (2009) 609–613.

[14] L. de Pontual, E. Yao, P. Callier, L. Faivre, V. Drouin, S. Cariou, A. Van Haeringen, D. Geneviève, A. Goldenberg, M. Oufadem, et al., Germline deletion of the miR-17 [sim] 92 cluster causes skeletal and growth defects in humans, Nature genetics 43 (10) (2011) 1026–1030.

[15] M. Esteller, Non-coding RNAs in human disease, Nature reviews. Genetics 12 (12) (2011) 861.

[16] M. T. McManus, MicroRNAs and cancer, in: Seminars in cancer biology, vol. 13, Elsevier, 253–258, 2003.

[17] J. R. Prensner, A. M. Chinnaiyan, The emergence of lncRNAs in cancer biology, Cancer discovery 1 (5) (2011) 391–407.

[18] M. Abdelrahim, S. Safe, C. Baker, A. Abudayyeh, RNAi and cancer: Implications and applications, Journal of RNAi and gene silencing: an international journal of RNA and gene targeting research 2 (1) (2006) 136.

[19] Y. Mei, D. Clark, L. Mao, Novel dimensions of piRNAs in cancer, Cancer letters 336 (1) (2013) 46–52.

[20] K. W. Ng, C. Anderson, E. A. Marshall, B. C. Minatel, K. S. Enfield, H. L. Saprunoff, W. L. Lam, V. D. Martinez, Piwi-interacting RNAs in cancer:

emerging functions and clinical utility, Molecular cancer 15 (1) (2016) 5.

[21] F. Sanger, S. Nicklen, A. R. Coulson, DNA sequencing with chain-terminating inhibitors, Proceedings of the national academy of sciences 74 (12) (1977) 5463–5467.

[22] F. Sanger, G. M. Air, B. G. Barrell, N. L. Brown, A. R. Coulson, J. C. Fiddes, C. Hutchison, P. M. Slocombe, M. Smith, Nucleotide sequence of bacteriophage φX174 DNA, nature 265 (5596) (1977) 687–695.

[23] M. Kircher, J. Kelso, High-throughput DNA sequencing–concepts and limitations, Bioessays 32 (6) (2010) 524–536.

[24] T. P. Niedringhaus, D. Milanova, M. B. Kerby, M. P. Snyder, A. E. Barron, Landscape of next-generation sequencing technologies, Analytical chemistry 83 (12) (2011) 4327–4341.

[25] J. R. Miller, A. L. Delcher, S. Koren, E. Venter, B. P. Walenz, A. Brownley, J. Johnson, K. Li, C. Mobarry, G. Sutton, Aggressive assembly of pyrosequencing reads with mates, Bioinformatics 24 (24) (2008) 2818–2824.

[26] F. J. Ribeiro, D. Przybylski, S. Yin, T. Sharpe, S. Gnerre, A. Abouelleil, A. M. Berlin, A. Montmayeur, T. P. Shea, B. J. Walker, et al., Finished bacterial genomes from shotgun sequence data, Genome research 22 (11) (2012) 2270–2277.

[27] V. Deshpande, E. D. Fung, S. Pham, V. Bafna, Cerulean: A hybrid assembly using high throughput short and long reads, in: International Workshop on Algorithms in Bioinformatics, Springer, 349–363, 2013.

[28] A. Bashir, A. A. Klammer, W. P. Robins, C.-S. Chin, D. Webster, E. Paxinos, D. Hsu, M. Ashby, S. Wang, P. Peluso, et al., A hybrid approach for the automated finishing of bacterial genomes, Nature biotechnology 30 (7) (2012) 701–707.

[29] S. Koren, G. P. Harhay, T. P. Smith, J. L. Bono, D. M. Harhay, S. D. Mcvey, D. Radune, N. H. Bergman, A. M. Phillippy, Reducing assembly complexity of microbial genomes with single-molecule sequencing, Genome biology 14 (9) (2013) 1.

[30] A. S. Motahari, G. Bresler, N. David, Information theory of DNA shotgun sequencing, IEEE Transactions on Information Theory 59 (10) (2013) 6273–6289.

[31] G. Bresler, M. Bresler, D. Tse, Optimal assembly for high throughput shotgun sequencing, BMC bioinformatics 14 (5) (2013) 1.

[32] E. S. Lander, M. S. Waterman, Genomic mapping by fingerprinting random clones: a mathematical analysis, Genomics 2 (3) (1988) 231–239.

[33] L. Ilie, F. Fazayeli, S. Ilie, HiTEC: accurate error correction in high-throughput sequencing data, Bioinformatics 27 (3) (2011) 295–302.

[34] X. Yang, K. S. Dorman, S. Aluru, Reptile: representative tiling for short read error correction, Bioinformatics 26 (20) (2010) 2526–2533.

[35] X. Yang, S. P. Chockalingam, S. Aluru, A survey of error-correction methods for next-generation sequencing, Briefings in bioinformatics 14 (1) (2013) 56–66.

[36] X. Zhao, L. E. Palmer, R. Bolanos, C. Mircean, D. Fasulo, G. M. Wittenberg, EDAR: an efficient error detection and removal algorithm for next generation sequencing data, Journal of computational biology 17 (11) (2010) 1549–1560.

[37] I. Leitch, Genome sizes through the ages, Heredity 99 (2) (2007) 121–122.

[38] NHGHI, Human genome sequence quality standards .

[39] J. C. Marioni, C. E. Mason, S. M. Mane, M. Stephens, Y. Gilad, RNA-seq: an assessment of technical reproducibility and comparison with gene expression arrays, Genome research 18 (9) (2008) 1509–1517.

[40] E. Ukkonen, Approximate string-matching with q-grams and maximal matches,

Theoretical computer science 92 (1) (1992) 191–211.

[41] S. Onn, Convex discrete optimization Convex Discrete Optimization, in: Encyclopedia of Optimization, Springer, 513–550, 2008.

[42] M. Lubin, E. Yamangil, R. Bent, J. P. Vielma, Extended formulations in mixed-integer convex programming, in: International Conference on Integer Programming and Combinatorial Optimization, Springer, 102–113, 2016.

[43] S. Boyd, L. Vandenberghe, Convex optimization, Cambridge university press, 2004.

[44] J. Tarhio, E. Ukkonen, A greedy algorithm for constructing shortest common superstrings, in: International Symposium on Mathematical Foundations of Computer Science, Springer, 602–610, 1986.

[45] J. R. Miller, S. Koren, G. Sutton, Assembly algorithms for next-generation sequencing data, Genomics 95 (6) (2010) 315–327.

[46] P. E. Compeau, P. A. Pevzner, G. Tesler, How to apply de Bruijn graphs to genome assembly, Nature biotechnology 29 (11) (2011) 987–991.

[47] R. M. Idury, M. S. Waterman, A new algorithm for DNA sequence assembly, Journal of computational biology 2 (2) (1995) 291–306.

[48] J. Shendure, H. Ji, Next-generation DNA sequencing, Nature biotechnology 26 (10) (2008) 1135–1145.

[49] P. A. Pevzner, H. Tang, M. S. Waterman, An Eulerian path approach to DNA fragment assembly, Proceedings of the National Academy of Sciences 98 (17) (2001) 9748–9753.

[50] S. L. Salzberg, A. M. Phillippy, A. Zimin, D. Puiu, T. Magoc, S. Koren, T. J. Treangen, M. C. Schatz, A. L. Delcher, M. Roberts, et al., GAGE: A critical evaluation of genome assemblies and assembly algorithms, Genome research 22 (3) (2012) 557–567.

[51] M. Boetzer, W. Pirovano, SSPACE-LongRead: scaffolding bacterial draft genomes using long read sequence information, BMC bioinformatics 15 (1) (2014) 1.

[52] M. A. Quail, M. Smith, P. Coupland, T. D. Otto, S. R. Harris, T. R. Connor, A. Bertoni, H. P. Swerdlow, Y. Gu, A tale of three next generation sequencing platforms: comparison of Ion Torrent, Pacific Biosciences and Illumina MiSeq sequencers, BMC genomics 13 (1) (2012) 1.

[53] S. B. Needleman, C. D. Wunsch, A general method applicable to the search for similarities in the amino acid sequence of two proteins, Journal of molecular biology 48 (3) (1970) 443–453.

[54] O. Gotoh, An improved algorithm for matching biological sequences, Journal of molecular biology 162 (3) (1982) 705–708.

[55] P. P. Gardner, A. Wilm, S. Washietl, A benchmark of multiple sequence alignment programs upon structural RNAs, Nucleic acids research 33 (8) (2005) 2433–2439.

[56] C. Zwieb, C. Glotz, R. Brimacombe, Secondary structure comparisons between small subunit ribosomal RNA molecules from six different species, Nucleic Acids Research 9 (15) (1981) 3621–3640.

[57] C. Glotz, C. Zwieb, R. Brimacombe, K. Edwards, H. Kössel, Secondary structure of the large subunit ribosomal RNA from Escherichia coli, Zea mays chloroplast, and human and mouse mitochondrial ribosomes, Nucleic Acids Research 9 (14) (1981) 3287–3306.

[58] H. Raué, J. Klootwijk, W. Musters, Evolutionary conservation of structure and function of high molecular weight ribosomal RNA, Progress in biophysics and molecular biology 51 (2) (1988) 77–129.

[59] E. K. Freyhult, J. P. Bollback, P. P. Gardner, Exploring genomic dark matter: a

critical assessment of the performance of homology search methods on noncoding RNA, Genome research 17 (1) (2007) 117–125.

[60] P. Johnsson, L. Lipovich, D. Grandér, K. V. Morris, Evolutionary conservation of long non-coding RNAs; sequence, structure, function, Biochimica et Biophysica Acta (BBA)-General Subjects 1840 (3) (2014) 1063–1071.

[61] I. Tinoco, C. Bustamante, How RNA folds, Journal of molecular biology 293 (2) (1999) 271–281.

[62] C. Flamm, W. Fontana, I. L. Hofacker, P. Schuster, RNA folding at elementary step resolution, Rna 6 (03) (2000) 325–338.

[63] W. J. Greenleaf, K. L. Frieda, D. A. Foster, M. T. Woodside, S. M. Block, Direct observation of hierarchical folding in single riboswitch aptamers, Science 319 (5863) (2008) 630–633.

[64] D. Sankoff, Simultaneous solution of the RNA folding, alignment and protosequence problems, SIAM Journal on Applied Mathematics 45 (5) (1985) 810–825.

[65] D. H. Mathews, D. H. Turner, Dynalign: an algorithm for finding the secondary structure common to two RNA sequences, Journal of molecular biology 317 (2) (2002) 191–203.

[66] J. H. Havgaard, R. B. Lyngsø, G. D. Stormo, J. Gorodkin, Pairwise local structural alignment of RNA sequences with sequence similarity less than 40%, Bioinformatics 21 (9) (2005) 1815–1824.

[67] A. O. Harmanci, G. Sharma, D. H. Mathews, PARTS: Probabilistic Alignment for RNA joinT Secondary structure prediction, Nucleic acids research 36 (7) (2008) 2406–2417.

[68] Y. Fu, G. Sharma, D. H. Mathews, Dynalign II: common secondary structure prediction for RNA homologs with domain insertions, Nucleic acids research

42 (22) (2014) 13939–13948.

[69] S. Will, K. Reiche, I. L. Hofacker, P. F. Stadler, R. Backofen, Inferring noncoding RNA families and classes by means of genome-scale structure-based clustering, PLoS Comput Biol 3 (4) (2007) e65.

[70] S. Will, C. Otto, M. Miladi, M. Möhl, R. Backofen, SPARSE: Quadratic time simultaneous alignment and folding of RNAs without sequence-based heuristics, Bioinformatics 31 (15) (2015) 2489–2496.

[71] I. L. Hofacker, S. H. Bernhart, P. F. Stadler, Alignment of RNA base pairing probability matrices, Bioinformatics 20 (14) (2004) 2222–2227.

[72] D. P. Goldenberg, Finding the right fold., Nature structural biology 6 (11).

[73] A. Mashaghi, R. J. van Wijk, S. J. Tans, Circuit Topology of Proteins and Nucleic Acids, Structure 22 (9) (2014) 1227–1237.

[74] A. Gursoy, O. Keskin, R. Nussinov, Topological properties of protein interaction networks from a structural perspective, Biochemical Society Transactions 36 (6) (2008) 1398–1403.

[75] W. R. Pearson, An introduction to sequence similarity (homology) searching, Current protocols in bioinformatics (2013) 3–1.

[76] R. Singh, J. Xu, B. Berger, Global alignment of multiple protein interaction networks with application to functional orthology detection, Proceedings of the National Academy of Sciences 105 (35) (2008) 12763–12768.

[77] M. Leordeanu, M. Hebert, A spectral technique for correspondence problems using pairwise constraints, in: Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on, vol. 2, IEEE, 1482–1489, 2005.

[78] B.-J. Yoon, X. Qian, S. M. E. Sahraeian, Comparative analysis of biological networks: Hidden markov model and markov chain-based approach, IEEE Signal Processing Magazine 1 (29) (2012) 22–34.

[79] L. Page, S. Brin, R. Motwani, T. Winograd, The PageRank citation ranking: bringing order to the web. .

[80] J. S. McCaskill, The equilibrium partition function and base pair binding probabilities for RNA secondary structure, Biopolymers 29 (6-7) (1990) 1105–1119.

[81] D. H. Mathews, Using an RNA secondary structure partition function to determine confidence in base pairs predicted by free energy minimization, Rna 10 (8) (2004) 1178–1190.

[82] D. H. Turner, D. H. Mathews, NNDB: the nearest neighbor parameter database for predicting stability of nucleic acid secondary structure, Nucleic acids research (2009) gkp892.

[83] B.-J. Yoon, Hidden Markov models and their applications in biological sequence analysis, Current genomics 10 (6) (2009) 402–415.

[84] D. W. Mount, Using hidden Markov models to align multiple sequences, Cold Spring Harbor Protocols 2009 (7) (2009) pdb–top41.

[85] B.-J. Yoon, Sequence alignment by passing messages, BMC genomics 15 (1) (2014) 1.

[86] A. Chakraborty, S. Bandyopadhyay, FOGSAA: Fast optimal global sequence alignment algorithm, Scientific reports 3.

[87] J. S. Reuter, D. H. Mathews, RNAstructure: software for RNA secondary structure prediction and analysis, BMC bioinformatics 11 (1) (2010) 1.

[88] S. Griffiths-Jones, A. Bateman, M. Marshall, A. Khanna, S. R. Eddy, Rfam: an RNA family database, Nucleic acids research 31 (1) (2003) 439–441.

[89] A. Wilm, I. Mainz, G. Steger, An enhanced RNA alignment benchmark for sequence alignment programs, Algorithms for molecular biology 1 (1) (2006) 1.

[90] N. C. Lau, A. G. Seto, J. Kim, S. Kuramochi-Miyagawa, T. Nakano, D. P.

Bartel, R. E. Kingston, Characterization of the piRNA complex from rat testes, Science 313 (5785) (2006) 363–367.

[91] A. A. Aravin, G. J. Hannon, J. Brennecke, The Piwi-piRNA pathway provides an adaptive defense in the transposon arms race, science 318 (5851) (2007) 761–764.

[92] A. G. Seto, R. E. Kingston, N. C. Lau, The coming of age for Piwi proteins, Molecular cell 26 (5) (2007) 603–609.

[93] S. S. Lakshmi, S. Agrawal, piRNABank: a web resource on classified and clustered Piwi-interacting RNAs, Nucleic acids research 36 (suppl 1) (2008) D173–D177.

[94] A. Aravin, D. Gaidatzis, S. Pfeffer, M. Lagos-Quintana, P. Landgraf, N. Iovino, P. Morris, M. J. Brownstein, S. Kuramochi-Miyagawa, T. Nakano, et al., A novel class of small RNAs bind to MILI protein in mouse testes, Nature 442 (7099) (2006) 203–207.

[95] Y. Kirino, Z. Mourelatos, The mouse homolog of HEN1 is a potential methylase for Piwi-interacting RNAs, Rna 13 (9) (2007) 1397–1401.

[96] D. Betel, R. Sheridan, D. S. Marks, C. Sander, Computational analysis of mouse piRNA sequence and biogenesis, PLoS Comput Biol 3 (11) (2007) e222.

[97] Y. Zhang, X. Wang, L. Kang, A k-mer scheme to predict piRNAs and characterize locust piRNAs, Bioinformatics 27 (6) (2011) 771–776.

[98] A. Girard, R. Sachidanandam, G. J. Hannon, M. A. Carmell, A germline-specific class of small RNAs binds mammalian Piwi proteins, Nature 442 (7099) (2006) 199–202.

[99] S. Yamanaka, M. C. Siomi, H. Siomi, piRNA clusters and open chromatin structure, Mobile DNA 5 (1) (2014) 22.

[100] A. A. Erwin, M. A. Galdos, M. L. Wickersheim, C. C. Harrison, K. D.

Marr, J. M. Colicchio, J. P. Blumenstiel, piRNAs are associated with diverse transgenerational effects on gene and transposon expression in a hybrid dysgenic syndrome of D. virilis, PLoS Genet 11 (8) (2015) e1005332.

[101] D. Rosenkranz, H. Zischler, proTRAC-a software for probabilistic piRNA cluster detection, visualization and analysis, BMC bioinformatics 13 (1) (2012) 5.

[102] I. Jung, J. C. Park, S. Kim, piClust: a density based piRNA clustering algorithm, Computational biology and chemistry 50 (2014) 60–67.

[103] J. Brayet, F. Zehraoui, L. Jeanson-Leh, D. Israeli, F. Tahi, Towards a piRNA prediction using multiple kernel fusion and support vector machine, Bioinformatics 30 (17) (2014) i364–i370.

[104] P. Zhang, X. Si, G. Skogerbø, J. Wang, D. Cui, Y. Li, X. Sun, L. Liu, B. Sun, R. Chen, et al., piRBase: a web resource assisting piRNA functional study, Database 2014 (2014) bau110.

[105] B. Y. M. Cheng, J. G. Carbonell, J. Klein-Seetharaman, Protein classification based on text document classification techniques, Proteins: Structure, Function, and Bioinformatics 58 (4) (2005) 955–970.

[106] Q. Dong, K. Wang, X. Liu, Identifying the missing proteins in human proteome by biological language model, BMC Systems Biology 10 (4) (2016) 393.

[107] I. Salvador, J.-M. Benedi, RNA modeling by combining stochastic context-free grammars and n-gram models, International Journal of Pattern Recognition and Artificial Intelligence 16 (03) (2002) 309–315.

[108] A. Tomović, P. Janičić, V. Kešelj, N-Gram-based classification and unsupervised hierarchical clustering of genome sequences, Computer methods and programs in biomedicine 81 (2) (2006) 137–153.

[109] J. Brennecke, A. A. Aravin, A. Stark, M. Dus, M. Kellis, R. Sachidanandam, G. J. Hannon, Discrete small RNA-generating loci as master regulators of

transposon activity in Drosophila, Cell 128 (6) (2007) 1089–1103.

[110] E. Beyret, N. Liu, H. Lin, piRNA biogenesis during adult spermatogenesis in mice is independent of the ping-pong mechanism, Cell research 22 (10) (2012) 1429–1439.

[111] B. F. Manly, Randomization, bootstrap and Monte Carlo methods in biology, vol. 70, CRC Press, 2006.

[112] C.-C. Chang, C.-J. Lin, LIBSVM: A library for support vector machines, ACM Transactions on Intelligent Systems and Technology 2 (2011) 27:1–27:27, URL `http://www.csie.ntu.edu.tw/ cjlin/libsvm`.

[113] S. Griffiths-Jones, A. Bateman, M. Marshall, A. Khanna, S. R. Eddy, Rfam: an RNA family database, Nucleic acids research 31 (1) (2003) 439–441.

[114] E. P. Nawrocki, S. W. Burge, A. Bateman, J. Daub, R. Y. Eberhardt, S. R. Eddy, E. W. Floden, P. P. Gardner, T. A. Jones, J. Tate, et al., Rfam 12.0: updates to the RNA families database, Nucleic acids research (2014) gku1063.

[115] A. P. Bradley, The use of the area under the ROC curve in the evaluation of machine learning algorithms, Pattern recognition 30 (7) (1997) 1145–1159.

[116] C. X. Ling, J. Huang, H. Zhang, AUC: a better measure than accuracy in comparing learning algorithms, in: Conference of the Canadian Society for Computational Studies of Intelligence, Springer, 329–341, 2003.

[117] P. P. Amaral, M. E. Dinger, T. R. Mercer, J. S. Mattick, The eukaryotic genome as an RNA machine, science 319 (5871) (2008) 1787–1789.

[118] T. Doniger, R. Katz, C. Wachtel, S. Michaeli, R. Unger, A comparative genome-wide study of ncRNAs in trypanosomatids, BMC genomics 11 (1) (2010) 615.

[119] E. K. Freyhult, J. P. Bollback, P. P. Gardner, Exploring genomic dark matter: a critical assessment of the performance of homology search methods on noncoding RNA, Genome research 17 (1) (2007) 117–125.

[120] G. Bussotti, C. Notredame, A. J. Enright, Detecting and comparing non-coding RNAs in the high-throughput era, International journal of molecular sciences 14 (8) (2013) 15423–15458.

[121] E. Rivas, S. R. Eddy, Secondary structure alone is generally not statistically significant for the detection of noncoding RNAs, Bioinformatics 16 (7) (2000) 583–605.

[122] E. Rivas, R. J. Klein, T. A. Jones, S. R. Eddy, Computational identification of noncoding RNAs in *E. coli* by comparative genomics, Current biology 11 (17) (2001) 1369–1373.

[123] S. Washietl, I. L. Hofacker, Consensus folding of aligned sequences as a new measure for the detection of functional RNAs by comparative genomics, Journal of molecular biology 342 (1) (2004) 19–30.

[124] J. S. Pedersen, G. Bejerano, A. Siepel, K. Rosenbloom, K. Lindblad-Toh, E. S. Lander, J. Kent, W. Miller, D. Haussler, Identification and classification of conserved RNA secondary structures in the human genome, PLoS Comput Biol 2 (4) (2006) e33.

[125] L. Argaman, R. Hershberg, J. Vogel, G. Bejerano, E. G. H. Wagner, H. Margalit, S. Altuvia, Novel small RNA-encoding genes in the intergenic regions of *Escherichia coli*, Current Biology 11 (12) (2001) 941–950.

[126] K. M. Wassarman, F. Repoila, C. Rosenow, G. Storz, S. Gottesman, Identification of novel small RNAs using comparative genomics and microarrays, Genes & development 15 (13) (2001) 1637–1651.

[127] E. Rivas, S. R. Eddy, Noncoding RNA gene detection using comparative sequence analysis, BMC bioinformatics 2 (1) (2001) 1.

[128] J. P. McCutcheon, S. R. Eddy, Computational identification of non-coding RNAs in *Saccharomyces cerevisiae* by comparative genomics, Nucleic Acids

Research 31 (14) (2003) 4119–4128.

[129] S. Washietl, I. L. Hofacker, P. F. Stadler, Fast and reliable prediction of noncoding RNAs, Proceedings of the National Academy of Sciences of the United States of America 102 (7) (2005) 2454–2459.

[130] I. Hofacker, P. F. Stadler, RNAz 2.0: improved noncoding RNA detection, in: Pacific Symposium on Biocomputing, vol. 15, 69–79, 2010.

[131] Y. Fu, Z. Z. Xu, Z. J. Lu, S. Zhao, D. H. Mathews, Discovery of Novel ncRNA Sequences in Multiple Genome Alignments on the Basis of Conserved and Stable Secondary Structures, PloS one 10 (6) (2015) e0130200.

[132] M. Riley, T. Abe, M. B. Arnaud, M. K. Berlyn, F. R. Blattner, R. R. Chaudhuri, J. D. Glasner, T. Horiuchi, I. M. Keseler, T. Kosuge, et al., *Escherichia coli* K-12: a cooperatively developed annotation snapshot–2005, Nucleic acids research 34 (1) (2006) 1–9.

[133] S. D. Bentley, K. F. Chater, A.-M. Cerdeno-Tarraga, G. L. Challis, N. Thomson, K. D. James, D. E. Harris, M. A. Quail, H. Kieser, D. Harper, et al., Complete genome sequence of the model actinomycete Streptomyces coelicolor A3 (2), Nature 417 (6885) (2002) 141–147.

[134] R. Lorenz, S. H. Bernhart, C. H. Zu Siederdissen, H. Tafer, C. Flamm, P. F. Stadler, I. L. Hofacker, ViennaRNA Package 2.0, Algorithms for Molecular Biology 6 (1) (2011) 1.

[135] J. S. Reuter, D. H. Mathews, RNAstructure: software for RNA secondary structure prediction and analysis, BMC bioinformatics 11 (1) (2010) 1.

[136] H. S. Shibata, A. Minagawa, H. Takaku, M. Takagi, M. Nashimoto, Unstructured RNA is a substrate for tRNase Z, Biochemistry 45 (17) (2006) 5486–5492.

[137] S. H. Bernhart, I. L. Hofacker, From consensus structure prediction to RNA gene finding, Briefings in functional genomics & proteomics 8 (6) (2009) 461–471.

[138] M. Zuker, P. Stiegler, Optimal computer folding of large RNA sequences using thermodynamics and auxiliary information, Nucleic acids research 9 (1) (1981) 133–148.

[139] D. H. Mathews, J. Sabina, M. Zuker, D. H. Turner, Expanded sequence dependence of thermodynamic parameters improves prediction of RNA secondary structure, Journal of molecular biology 288 (5) (1999) 911–940.

[140] C. Workman, A. Krogh, No evidence that mRNAs have lower folding free energies than random sequences with the same dinucleotide distribution, Nucleic acids research 27 (24) (1999) 4816–4822.

[141] I. L. Hofacker, W. Fontana, P. F. Stadler, L. S. Bonhoeffer, M. Tacker, P. Schuster, Fast folding and comparison of RNA secondary structures, Monatshefte für Chemie/Chemical Monthly 125 (2) (1994) 167–188.

[142] S. H. Bernhart, I. L. Hofacker, S. Will, A. R. Gruber, P. F. Stadler, RNAalifold: improved consensus structure prediction for RNA alignments, BMC bioinformatics 9 (1) (2008) 1.

[143] Y. Wang, A. Manzour, P. Shareghi, T. I. Shaw, Y.-W. Li, R. L. Malmberg, L. Cai, Stable stem enabled Shannon entropies distinguish non-coding RNAs from random backgrounds, BMC bioinformatics 13 (5) (2012) 1.

[144] S. M. ElGokhy, M. ElHefnawi, A. Shoukry, Ensemble-based classification approach for micro-RNA mining applied on diverse metagenomic sequences, BMC research notes 7 (1) (2014) 1.

[145] J. N. Zadeh, B. R. Wolfe, N. A. Pierce, Nucleic acid sequence design via efficient ensemble defect optimization, Journal of computational chemistry 32 (3) (2011) 439–452.

[146] J. S. Martin, Describing the Structural Diversity within an RNA's Ensemble, Entropy 16 (3) (2014) 1331–1348.

[147] J. S. McCaskill, The equilibrium partition function and base pair binding probabilities for RNA secondary structure, Biopolymers 29 (6-7) (1990) 1105–1119.

[148] Z. J. Lu, J. W. Gloor, D. H. Mathews, Improved RNA secondary structure prediction by maximizing expected pair accuracy, Rna 15 (10) (2009) 1805–1813.

[149] F. Lou, P. Clote, Maximum expected accurate structural neighbors of an RNA secondary structure, in: Computational Advances in Bio and Medical Sciences (ICCABS), 2011 IEEE 1st International Conference on, IEEE, 123–128, 2011.

[150] M. Hajiaghayi, A. Condon, H. H. Hoos, Analysis of energy-based algorithms for RNA secondary structure prediction, BMC bioinformatics 13 (1) (2012) 1.

[151] I. Guyon, A. Elisseeff, An introduction to variable and feature selection, Journal of machine learning research 3 (Mar) (2003) 1157–1182.

[152] S. Ding, Feature selection based F-score and ACO algorithm in support vector machine, in: Knowledge Acquisition and Modeling, 2009. KAM'09. Second International Symposium on, vol. 1, IEEE, 19–23, 2009.

[153] R. Lorenz, M. T. Wolfinger, A. Tanzer, I. L. Hofacker, Predicting RNA secondary structures from sequence and probing data, Methods .

[154] T. Dunning, Statistical identification of language, Computing Research Laboratory, New Mexico State University, 1994.

[155] B. R. King, C. Guda, ngLOC: an n-gram-based Bayesian method for estimating the subcellular proteomes of eukaryotes, Genome biology 8 (5) (2007) 1.

[156] M. A. Larkin, G. Blackshields, N. Brown, R. Chenna, P. A. McGettigan, H. McWilliam, F. Valentin, I. M. Wallace, A. Wilm, R. Lopez, et al., Clustal W and Clustal X version 2.0, bioinformatics 23 (21) (2007) 2947–2948.

[157] A. Bordes, S. Ertekin, J. Weston, L. Bottou, Fast kernel classifiers with online and active learning, Journal of Machine Learning Research 6 (Sep) (2005)

1579–1619.

[158] N. Deng, Y. Tian, C. Zhang, Support vector machines: optimization based theory, algorithms, and extensions, CRC press, 2012.

[159] A. E. Darling, B. Mau, N. T. Perna, progressiveMauve: multiple genome alignment with gene gain, loss and rearrangement, PloS one 5 (6) (2010) e11147.

[160] P. J. Kersey, J. E. Allen, I. Armean, S. Boddu, B. J. Bolt, D. Carvalho-Silva, M. Christensen, P. Davis, L. J. Falin, C. Grabmueller, et al., Ensembl Genomes 2016: more genomes, more complexity, Nucleic acids research 44 (D1) (2016) D574–D580.

[161] P. Sætrom, R. Sneve, K. I. Kristiansen, O. Snøve, T. Grünfeld, T. Rognes, E. Seeberg, Predicting non-coding RNA genes in *Escherichia coli* with boosted genetic programming, Nucleic acids research 33 (10) (2005) 3263–3270.

[162] M.-P. Vockenhuber, C. M. Sharma, M. G. Statt, D. Schmidt, Z. Xu, S. Dietrich, H. Liesegang, D. H. Mathews, B. Suess, Deep sequencing-based identification of small non-coding RNAs in Streptomyces coelicolor, RNA biology 8 (3) (2011) 468–477.

[163] S. Washietl, I. L. Hofacker, M. Lukasser, A. Hüttenhofer, P. F. Stadler, Mapping of conserved RNA secondary structures predicts thousands of functional noncoding RNAs in the human genome, Nature biotechnology 23 (11) (2005) 1383–1390.

[164] P. Baldi, Y. Chauvin, T. Hunkapiller, M. A. McClure, Hidden Markov models of biological primary sequence information., Proceedings of the National Academy of Sciences 91 (3) (1994) 1059–1063.

[165] Y. Sakakibara, M. Brown, R. Hughey, I. S. Mian, K. Sjölander, R. C. Underwood, D. Haussler, Stochastic context-free grammers for tRNA modeling, Nucleic

acids research 22 (23) (1994) 5112–5120.

APPENDIX A

MATHEMATICAL PROOFS OF THE PROPOSITIONS

Proof of Proposition 1.1

Suppose the read locations follow a Poisson arrival process with the arrival rate $\lambda = N/G$ (event intensity) [30], and the length G of each genome sequence satisfies $G \gg L_i$ so that we may ignore the terminal effect in the sequence. Based on this model, the arrival process is **memoryless** (i.e.. $P(t > t_1 + t_2 | t > t_1) = P(t > t_2)$), so any read can be seen as a new arrival read in the process. Furthermore, the arrival gap between each pair of consecutive reads follows an exponential distribution with the rate parameter λ. Hence the probability that the arrival gap is larger than T is $P(t > T) = e^{-\lambda T}$. Consider an arrival read of length L_a that is the last read in a contig. For notational simplicity, we denote the length of the read that is longer than L_a as L_i. Otherwise, it is denoted as L_j. Hence, the relationship between the read lengths L_i, L_j, and L_a is $L_i > L_a \geq L_j$. For the read with the length L_a to be the last read, any read with a longer length L_i must start at least $L_i - L_a$ bases before this read or start at least $L_a - K$ bases after the start of this read, so that there is no read in the interval $[L_a - L_i, L_a - K]$. Based on the Poisson model, the probability that there will be no read with the longer length to cover or overlap with the last read is $e^{-\sum_{L_i > L_a} \frac{N_i}{G}(L_i - K)}$. Similarly, there is no read with the shorter or equal length L_j in the interval $[L_a - L_j, L_a - K]$ and the probability that there is no reads with the length L_j overlap with the read is $e^{-\sum_{L_j \leq L_a} \frac{N_j}{G}(L_j - K)}$. Therefore the probability that the arrival read is the last read in a contig is $P_L = \sum_a \frac{N_a}{N} e^{-\sum_n \frac{N_n}{G}(L_n - K)} = e^{-\frac{N\overline{L}}{G} \sum_n \frac{N_n}{N}(\frac{L_n}{\overline{L}} - \theta)} = e^{-C(1-\theta)}$. This leads

to the probability that there exists reads without valid overlap can be bounded as $P_{\text{overlap}}(N, \overline{L}) \leq Ne^{-\frac{N\overline{L}}{G}(1-\theta)} = Ne^{-C(1-\theta)}$.

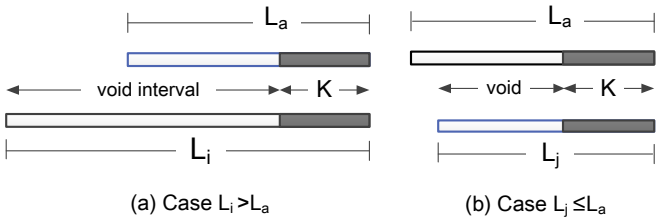


Figure A.1: The overlap patterns between reads. (a) Read length is greater than the last read length. (b) Read length is not greater than the last read length.

Proof of Proposition 1.2

Since the number of contigs equals the number of the last reads in contigs, the expected number of contigs is E[number of contigs]$= \sum_i P_L = Ne^{-C(1-\theta)}$.

Proof of Proposition 1.3

By treating the arrival reads as a geometric process, a read is either overlapped with the succeeding read or the end of the contig. So the expected number of reads in the contig is $1/P_L = e^{C(1-\theta)}$.

Proof of Proposition 1.4

Suppose the last arrival read with the read length L_i arrives at base position X_i, and G is sufficiently large. By checking the positions before the last read arrival, the possibility is either no arrival read or an arrival read with valid overlap. The probability of no arrival read is $1 - N/G$, while the probability of arrival read with valid overlap is $\frac{N}{G}(1 - P_L)$. Hence the random variable X_i has the distribution

$P(X_i = m) = (1 - \frac{N}{G}P_L)^{m-1}\frac{N_i}{G}P_L$. The expected length of the contig can be represented as the summation of arithmetic-geometric series with a correction term for the terminal effect, and can be further simplified for the long genome sequence ($G \gg 1$) as follows:

$$\begin{aligned} E[\text{length of contig}] &= \sum_{m,i} P(X_i = m)(m - 1 + L_i) \\ &= \sum_{m=1}^{G} (1 - \frac{N}{G}P_L)^{m-1}\frac{N}{G}P_L(m - 1 + \overline{L}) \\ &\quad - \sum_{i,m=1}^{L_i-1} P(X_i = G - L_i + m + 1)(G + m) \\ &\simeq \frac{G}{N}e^{C(1-\theta)} + \overline{L} \end{aligned}$$

Proof of Proposition 2.1

According to the Poisson arrival model, the repeat with the length m is not bridged if there is no longer read arriving in the preceding segment $(L - m - 1)$. Hence, the probability of the unbridged repeat with the length m is $P^{(1)} = e^{-\lambda(L-m-1)}$, where $\lambda = N/G$ is the read arrival rate. Here the read length is $L > m + 1$, otherwise $P^{(1)} = 1$.

Considering an interleaved repeat with repeat lengths m and n, under the assumption $m \geq n$ without loss of generality, it is not bridged if both the repeat pairs are not bridged. For the case when the read length is $L_1 > m + 1$, the probability that the interleaved repeat is not bridged by any read with the length L_1 is $P_{L_1}^{(2)} = e^{-2\frac{N}{G}(L_1-m-1)}e^{-2\frac{N}{G}(L_1-n-1)}$. For the case when the read length L_2 satisfies $m \geq L_2 - 1 > n$, the longer repeat cannot be bridged. Therefore, the unbridged probability becomes $P_{L_2}^{(2)} = e^{-2\frac{N}{G}(L_2-n-1)}$. By combining the results of the above cases for multiple read sources, the probability of unbridged interleaved repeat can be bounded by the following: P(unbridged interleaved repeats)= $P_{L_1}^{(2)} \cap P_{L_2}^{(2)} \leq P_{\text{bridged}}^{(2)}$

$= \sum_{mn} b_{mn} e^{-2\sum_i \frac{N_i}{G}[(L_i-m-1)^+ + (L_i-n-1)^+]}$, where b_{mn} is the number of interleaved repeats in the target genome sequence with $m \geq n$ and the step function $(L-n-1)^+ = \max(L-n-1, 0)$.

Proof of Proposition 2.2

Considering triple repeats in a genome sequence, let d_m be the number of triple repeats with the length m and ℓ_{triple} be the longest triple repeat length. The previous unbridged repeat formula with the read arrival rate $\lambda = N/G$ can be further extended to the case that triple repeats are all unbridged. Apply the union bound over triple repeats: $P_{\text{all}}^{(3)} = \sum_m d_m e^{-3\lambda(L-m-1)}$. For the case with multiple read sources, $P_{\text{all}}^{(3)} = 1$ if the read length $L \leq \ell_{\text{triple}} + 1$. Regarding triple repeats involved in other longer repeats in the genome sequence, they may still lead to ambiguity even when some repeats are bridged. Take Figure A.2 as an example. The triple repeat TGGCT is involved in the longer repeat TGGCTGTT, and forms a structure similar to interleaved repeats. Even with a bridging read that bridges the first repeat, there are still two possible candidate sequences, resulting in ambiguity. In this case, there is a unique corresponding sequence if either of the last two repeats is bridged. Therefore, the probability that unbridged triple repeats can result in ambiguity is bounded by the sum of the probabilities that all triple repeats are not bridged ($P_{\text{all}}^{(3)}$) and the correction term ($P_{\text{comb}}^{(3)}$) as shown below:

$$P_{\text{bridged}}^{(3)} = \sum_m d_m e^{-3\sum_i \frac{N_i}{G}(L_i-m-1)^+} + P_{\text{comb}}^{(3)}.$$

Here the correction term includes the cases when some repeating segments in triple repeats are bridged but the remaining unbridged segments still result in ambiguities in the assembly (e.g., the remaining unbridged segments of a triple repeat are involved in another interleaved repeat).

Figure A.2: The triple repeats involved in longer repeats. There are two candidate sequences even though the triple repeat segments are bridged by the reads ATGGCTG and GTGGCTG.

Proof of Proposition 2.3

Through **Propositions 2.1** and **2.2**, the unbridged probability bound is an immediate consequence of the union bound given a read set $(\mathbf{N}, \mathbf{L})$ as $P_{\text{bridged}}(\mathbf{N}, \mathbf{L}) = P^{(2)}_{\text{bridged}} \cup P^{(3)}_{\text{bridged}} \leq P^{(2)}_{\text{bridged}} + P^{(3)}_{\text{bridged}}$.

Proof of Proposition 3

Given c_m, the number of self-repeats with the length m in a genome sequence, let c_{m_1} be those self-repeats with individual repeat patterns. The remaining set c_{m_2} consists of the self-repeats with repeat patterns that appear more than once in the genome sequence. The probability of uncovered self-repeats in the set c_{m_1} is bounded by $\sum_{m_1} c_{m_1} e^{-\sum_i \frac{N_i}{G}(L_i - m_1 + 1)^+}$ while the probability of unbridged self-repeats in the set c_{m_2} is bounded by $\sum_{m_2} c_{m_2} e^{-\sum_i \frac{N_i}{G}(L_i - m_2 - 1)^+}$. By combining these two separate sets and further replacing covering reads with bridging reads, the probability of ambiguous self-repeats can be bounded as follows:

$$P_{\text{self}} \leq \sum_{m_1} c_{m_1} e^{-\sum_i \frac{N_i}{G}(L_i - m_1 + 1)^+} + \sum_{m_2} c_{m_2} e^{-\sum_i \frac{N_i}{G}(L_i - m_2 - 1)^+}$$
$$\leq \sum_{m} c_m e^{-\sum_i \frac{N_i}{G}(L_i - m - 1)^+}$$

Proof of Proposition 4

Finding the best solution for an optimization problem over a combinatorial set leads to convex discrete optimization if both the objective function and the constraint functions are convex [43]. In the following proofs, we first consider the case when we are given the read lengths $\mathbf{L}$ and then the case when using a fixed read cost to minimize the bound of the feasibility probability with the constraint $\forall L_i \geq 2\overline{L}/C(1-\theta)$.

Given read lengths $\mathbf{L}$, the objective function $\sum_i N_i L_i C_i$ is a linear function, hence it is convex with respect to $\mathbf{N}$. For the constraint functions, we can rearrange the formula into a summation of exponential functions as $P_{\text{feasible}}(\mathbf{N}, \mathbf{L}) = 1^T \mathbf{N} e^{-\underline{\alpha}^T \mathbf{N}} + \sum_m \gamma_m e^{-\underline{\beta}_m^T \mathbf{N}}$. It can be shown that the function is convex through the convex properties by taking the first- and second-order derivatives. By the **composition convex function** property: A composition function $f(x) = h \circ g(x)$ is convex if $h(x)$ is convex and non-increasing and $g(x)$ is a concave function. Thus the composition function $f(\underline{x}) = e^{-\underline{\alpha}^T \underline{x}}$ is a convex function if we have $h(x) = e^{-x}$ and $g(\underline{x}) = \sum_i \alpha_i x_i$. By the **sum of convex function** property: The sum of two convex functions $f(x) = h(x) + g(x)$ is convex if both $h(x)$ and $g(x)$ are convex. As a consequence of the above convex properties, the function $\sum_m \gamma_m e^{-\underline{\beta}_m^T \mathbf{N}}$ is a convex function. Hence, the constraint function $P_{\text{feasible}}(\mathbf{N}, \mathbf{L}) \leq \epsilon$ is convex if the product function $f(\underline{x}) = 1^T \underline{x} \cdot e^{-\alpha^T \underline{x}}$ is convex. This requires the Hessian matrix of $f(x)$ to be positive semi-definite, i.e. $H = \nabla^2 f(\underline{x}) \succ 0$. We can compute the Hessian matrix

of $f(x)$: $H = (\underline{\alpha}\,\underline{\alpha}^T 1^T \underline{x} - 1\underline{\alpha}^T - \underline{\alpha} 1^T) e^{-\underline{\alpha}^T \underline{x}}$, which is positive semi-definite when $(1^T \underline{x} - 1)\underline{\alpha} \geq 2$. Hence, $f(\underline{x})$ is a convex function if $(1^T \underline{x} - 1)\underline{\alpha} \geq 2$. Consequently, the condition for $P_{\text{feasible}}(\mathbf{N}, \mathbf{L})$ to be a convex function is $L_i \geq \frac{2}{C}\overline{L}\frac{N}{(N-1)(1-\theta)} = \frac{2}{C}\overline{L}\rho$, where $\rho = \frac{N}{(N-1)(1-\theta)}$. This condition can be further approximated by $L_i > \frac{2}{C}\overline{L}$ if $N \gg 1$ and $\theta \ll 1$. Therefore, the cost optimization is a convex discrete optimization problem when $\mathbf{L} \succ \frac{2G}{N}$.

For the case when the objective is to minimize the feasibility probability bound based on a fixed read cost, the optimization problem can be formulated in an epigraph form

$$\begin{aligned} \underset{\mathbf{N}=\{N_i\},E}{\text{minimize}} \quad & E \\ \text{s.t.} \quad & P_{\text{feasible}}(\mathbf{N}, \mathbf{L}) \leq E; \\ & \sum_i N_i L_i w_i = F \\ & N_i \in \mathbb{N}, \forall i. \end{aligned}$$

where F is the fixed read cost, and E is an auxiliary variable for optimizing the feasibility probability bound. As in our previous analysis, we arrive at an optimization problem, where the goal is to minimize the auxiliary variable E, which is a convex discrete optimization problem under the read length constraint.

Performance Gap Compared to the Feasibility Bound

The gap between the performance of the enhanced multi-bridging algorithm and the theoretical feasibility bound results from two error patterns when the triple repeats are involved in longer repeats with improper bridging reads. The first error pattern occurs when the X-node for triple repeats are bridged with reads but solved as simple repeats as illustrated in Figure A.3(a). The triple repeat TGGCT is

involved in the longer repeats ATTGGCT and TGGCTCT, and the X-node TGGCT is mistaken as a simple repeat by incompetent bridging reads in the X-node resolution step. In the worst case, even though the triple repeat is bridged by two bridging reads for the last two repeat segments, it can still lead to ambiguity in the assembly. The second error pattern occurs when the triple repeat is embedded in the front longer repeat, and the unresolved X-node is marked but still misleads the path traversal as illustrated in Figure A.3(b). The triple repeat TGGCT is embedded in the front longer repeats ATTGGCT, and the X-node TGGCT is marked by one bridging read for the last segment of repeat. Since the bridging reads of the first and last repeat segments are the same, the path traversal is misdirected to the last segment from the first segment, causing the assembly to fail in this case. Let g_{m_1} be the number of the first error pattern with the triple repeat length m_1 and g_{m_2} be the number of the second error pattern with the triple repeat length m_2. Based on the above error patterns, the gap to the feasibility bound can be upper bounded as

$$P_{\text{gap}} \leq \sum_{m_1} g_{m_1} P_0(m_1)(1 - P_0^2(m_1)) + \sum_{m_2} g_{m_2} P_0^2(m_2) P_1(m_2),$$

where $P_0(m) = e^{-\sum_i \frac{N_i}{G}(L_i - m - 1)^+}$ and $P_1(m) = 1 - P_0(m)$.

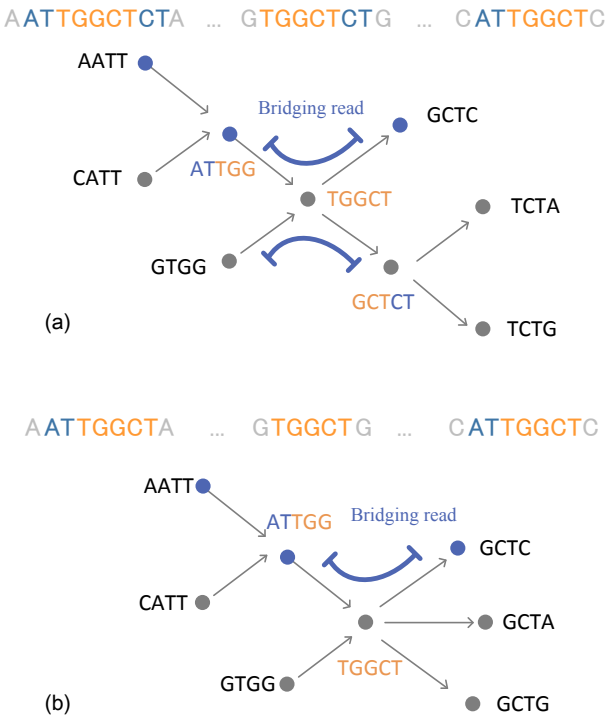


Figure A.3: Two error patterns of triple repeats.