Do RNA-dependent polymerases share common ancestry? A bioinformatic approach

Julianna Hudak

University of Nevada, Las Vegas

Follow this and additional works at: https://digitalscholarship.unlv.edu/rtds

Repository Citation

https://digitalscholarship.unlv.edu/rtds/1079

This Thesis is brought to you for free and open access by Digital Scholarship@UNLV. It has been accepted for inclusion in UNLV Retrospective Theses & Dissertations by an authorized administrator of Digital Scholarship@UNLV. For more information, please contact digitalscholarship@unlv.edu.
INFORMATION TO USERS

This manuscript has been reproduced from the microfilm master. UMI films the text directly from the original or copy submitted. Thus, some thesis and dissertation copies are in typewriter face, while others may be from any type of computer printer.

The quality of this reproduction is dependent upon the quality of the copy submitted. Broken or indistinct print, colored or poor quality illustrations and photographs, print bleedthrough, substandard margins, and improper alignment can adversely affect reproduction.

In the unlikely event that the author did not send UMI a complete manuscript and there are missing pages, these will be noted. Also, if unauthorized copyright material had to be removed, a note will indicate the deletion.

Oversize materials (e.g., maps, drawings, charts) are reproduced by sectioning the original, beginning at the upper left-hand corner and continuing from left to right in equal sections with small overlaps.

Photographs included in the original manuscript have been reproduced xerographically in this copy. Higher quality 6" x 9" black and white photographic prints are available for any photographs or illustrations appearing in this copy for an additional charge. Contact UMI directly to order.

Bell & Howell Information and Learning
300 North Zeeb Road, Ann Arbor, MI 48106-1346 USA

UMI®
800-521-0600

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
The Thesis prepared by

Julianna Hudak

Entitled

Do RNA-Dependent Polymerases Share Common Ancestry?

A Bioinformatic Approach.

is approved in partial fulfillment of the requirements for the degree of

Master of Science

[Signatures]

Examination Committee Chair

Dean of the Graduate College

Examination Committee Member (Research/Thesis Advisor)

Examination Committee Member

Graduate College Faculty Representative
ABSTRACT

Do RNA-Dependent Polymerases Share Common Ancestry? A Bioinformatic Approach

by

Julianna Hudak

Dr. George Plopper, Examination Committee Chair
Assistant Professor of Biology
University of Nevada, Las Vegas

Dr. Marcella McClure, Research and Thesis Advisor
Associate Professor of Biology
Montana State University, Bozeman

Bioinformatics is the use of computational methods to perform hypothesis-driven research that generates new knowledge from existing biological databases. For any bioinformatic analysis, it is important that the most accurate method(s) be used. The first portion of this thesis is a comparative evaluation of six programs designed for the local alignment of protein sequences. The results demonstrate that two of the programs, MEME and PROBE, outperform all other programs (BLOCKMAKER, ITERALIGN, MATCHBOX, and PIMA). The second portion of this thesis uses MEME and PROBE in an attempt to locate an ordered-series-of-motifs (OSM) among two groups of RNA-dependent polymerases, the large (L) protein from viruses in the order Mononegavirales and the reverse transcriptase (RT) protein from retroviruses and retroid agents. An OSM was not detected among the L and RT proteins, suggesting that they are not homologs. This result also supports the hypothesis that all RNA-dependent polymerases do not share common ancestry.
# TABLE OF CONTENTS

ABSTRACT ............................................................................................................................................................ iii  
LIST OF TABLES .................................................................................................................................................... v  
INTRODUCTION ...................................................................................................................................................... 1  
CHAPTER 1 A COMPARATIVE ANALYSIS OF COMPUTATIONAL MOTIF-DETECTION METHODS ................................................. 9  
CHAPTER 2 A PERFORMANCE AND RELIABILITY EVALUATION OF SEVERAL LOCAL ALIGNMENT METHODS ................................. 22  
CHAPTER 3 DO RNA-DEPENDENT POLYMERASES SHARE COMMON ANCESTRY? ......................................................... 46  
CONCLUDING REMARKS ........................................................................................................................................ 55  
BIBLIOGRAPHY ....................................................................................................................................................... 57  
APPENDIX I SMALL DATA SET OSMS .................................................................................................................. 60  
APPENDIX II GLOSSARY ....................................................................................................................................... 62  
VITA ........................................................................................................................................................................ 64  

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
LIST OF TABLES

Table 1.1: Computational Motif-Detection Programs ................................................................. 16
Table 1.2: Motif Scores and Parameters Options ......................................................................... 16
Table 2.1: Sequence Length, Percent Identity, and Distance Values .............................................. 25
Table 2.2: Program Results and Parameters for GLOB12 Data Set ............................................. 30
Table 2.3: Program Results and Parameters for KIN12 Data Set ................................................. 31
Table 2.4: Program Results and Parameters for PRO12 Data Set .................................................. 31
Table 2.5: Program Results and Parameters for RH12 Data Set ................................................... 32
Table 2.6: Program Results and Parameters for RT20 Data Set .................................................... 32
Table 2.7: Summary of Small Data Set Results ........................................................................... 34
Table 2.8: Program Results and Parameters for GLOB174 Data Set .......................................... 36
Table 2.9: Program Results and Parameters for KIN186 Data Set .............................................. 37
Table 2.10: Program Results and Parameters for PRO114 Data Set ............................................ 37
Table 2.11: Program Results and Parameters for RH169 Data Set .............................................. 38
Table 2.12: Program Results and Parameters for RT178 Data Set .............................................. 38
Table 2.13: Summary of Large Data Set Results .......................................................................... 40
Table 3.1: Results for L16, RT16, and LRT32 OSM Detection ....................................................... 52

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
INTRODUCTION

1 Bioinformatics Defined

Biological data are being generated at a phenomenal rate. Currently, the majority of this information is genomic sequence and 3-D structural data [M. Kanehisa, 1998]. Computational databases have become essential for handling and storing the increasing accumulation of data. Unfortunately, our ability to make biological sense out of the data lags far behind our ability to collect and store data. The need to analyze vast amounts of biological data has resulted in the emergence of an entirely new field, bioinformatics. Bioinformatics is the use of computational tools to perform hypothesis-driven research that generates new knowledge from existing biological databases. Genomics, for example, is one branch of bioinformatics that uses computers to analyze genomic sequence data.

Due to its interdisciplinary nature, bioinformatics can be divided into two parts: technical and analytical. The technical aspect involves the development and testing of algorithms, computer programs and databases. A variety of scientists contribute to the technical side of bioinformatics: mathematicians, statisticians, systems scientists, computer scientists, and even biologists. The computational tools created by these developers are an essential means for interpreting the massive amounts of biological data.

The analytical part of bioinformatics is the collection, analysis and interpretation of data. This hypothesis-driven research includes a wide range of topics such as molecular evolution, protein structure, and molecular interactions. A specific example is the use of bioinformatic tools to make predictions, such as the amino acid residues involved in the catalytic site of an enzyme, which can then be confirmed by crystallographic techniques. It should be noted that the validity of any bioinformatic analysis depends on the reliability and appropriateness of the computational methods being used.
An increasing number of new computer programs are available for the analysis of biological data, particularly sequence data. Although new, many of these methods are not improved. Methods may be statistically and mathematically robust, but not take into account important biological information. Many new programs appear to work only because they are tested on a small number and limited variety of data [M.A. McClure, 1994]. Thus, these methods may not perform as well on data sets that were not used to test the program. It is essential that the reliability of new methods be rigorously tested with a diverse set of data before assuming that the results are always biologically meaningful. Unfortunately, many program developers have not sufficiently tested their methods [M.A. McClure, 1994].

Even if a program has been adequately tested, it is still not clear whether it performs better than other methods that are available. Very few program developers have comparatively evaluated their methods against other similar programs to determine if they are more accurate. Comparative studies of bioinformatic tools are essential to the field of bioinformatics. With numerous methods available, it becomes impossible for biologists to make timely advances in their research if they have to evaluate the performance of every method before starting their analyses. However, without this stringent choice of methodology, inferior programs may inadvertently be used, making the analyses less biologically informative. Comparative studies of computational programs offer a foundation for biologists to choose the most appropriate method to use for their analyses. These studies also give feedback to program developers for producing improved versions of their programs. Thus, computer scientists are given a better understanding of what is needed and biologists are given a better understanding of what is available. [For recent reviews of Bioinformatics see M. Boguski, 1998 and P.F. Baldi et al., 1998].

2 From Information to Knowledge

The main driving force behind the increase in biological sequence data has been the advent of new, efficient experimental techniques, primarily DNA sequencing, combined with the advance of numerous genome sequencing projects. As a result, there are a large number of publicly available sequence databases via the Internet. The GenBank database is an example. Since its inception in 1982, GenBank grew to almost 100,000 sequences by 1989 and currently contains about 2,532,359 sequences [M. Boguski, 1998].
Genomic sequence data is used to derive knowledge of genome structure, function and evolution. When coined in 1986, the term "genomics" defined the science of mapping, sequencing and analyzing genomes. Today, the goal of genomics is to extract knowledge of genome function from the wealth of sequence information [P. Hieter and M. Boguski, 1997]. Since the use of computational methods has become an integral part of this process, genomics can be thought of as a branch of bioinformatics.

Bioinformatic tools can be used to assess the function of an uncharacterized gene product, or protein, by comparison to a gene product whose function is known. This is accomplished by establishing homology, or common ancestry, between the gene or protein sequences. Homologous proteins almost always retain the same function throughout the course of evolution. Therefore, if homology can be demonstrated between two sequences, common function can also be reliably predicted (R.F. Doolittle, 1986).

Before discussing how sequence homology is established, it should be mentioned that the analysis of genomic sequence data can take place at the nucleic acid or amino acid level. The type of sequence data used depends on the percentage of pairwise identity among the sequences and the type of analysis being performed. When a group of genes are greater than 70% identical at the nucleic acid level, very little change has occurred to the corresponding amino acid sequences. At this percent identity, amino acid sequences do not provide any evolutionary information and sequence analysis should be done at the nucleic acid level. This thesis is not concerned with sequence data in this range because this type of data would clearly be considered homologous according to the Dayhoff Criteria (discussed below). When a group of genes are less than 70% identical at the nucleic acid level, the sequences have changed so that some of the phylogenetic signal is masked behind the degeneracy of the genetic code. However, the corresponding amino acid sequences may still reflect common encoded regions, or phylogenetic remnants that can be used for sequence analysis. This thesis evaluates protein sequence data that is, on average, less than 30% identical at the amino acid level. At this percent identity, it is much more difficult to demonstrate homology and sequence analysis should take place at the amino acid level.

The computational approach to establishing homology between protein sequences involves amino acid sequence comparisons, or alignments. There are two types of sequence alignments, pairwise and multiple. A pairwise alignment compares two sequences to each other, generating the percent identity and similarity.
shared between the sequences. A similarity score is a measure of similarity between two sequences that incorporates the probability of substitution between amino acids based on their physicochemical properties and the ease of converting from one codon to another. A multiple sequence alignment compares three or more sequences to each other.

The percent identity that is determined by pairwise alignment can be used to claim homology between the sequences. Protein sequences that are greater than 30% identical are considered homologous. This is based on the Dayhoff criteria for homology, which was established through Monte Carlo simulations of protein sequence relationships [W.C. Barker and M.O. Dayhoff, 1972]. A Monte Carlo simulation is a comparison of observed sequences versus randomized, or shuffled, sequences. Alignments are created for each group of sequences and alignment scores are calculated (similarity scores, for example). Dayhoff observed that when the mean alignment score of the observed sequences is 3-5 standard deviations above the mean alignment score of the randomized sequences, the sequences can reasonably be expected to be homologous. Extensive use of this method has demonstrated that protein sequences consistently meet the criteria for homology when they are greater than 30% identical.

When protein sequences are less than 30% identical, homology is much more difficult to establish. Multiple sequence comparisons must be used to detect similar patterns among sequences that may support common ancestry. The idea of similar regions was first introduced by Dayhoff in 1983 as islands of amino acids that are conserved among sequences in a protein family [M.O. Dayhoff, 1983]. Today, a region of amino acids (approximately 1 to 9 residues) that remains conserved among sequences of a protein family is called a motif. When multiple motifs are present in a colinear order among a group of sequences, they are referred to as an Ordered-Series-of-Motifs (OSM) [M.A. McClure, 1991]. The amino acids comprising an OSM are essential to the structural and functional integrity of a protein and, therefore, are conserved throughout evolution [M.A. McClure, 1994]. These evolutionary constraints allow for an OSM to be designated as a protein family signature that can be used to classify proteins.

Ultimately, the detection of a common OSM among a group of protein sequences supports homology. The other possible explanation for a common OSM would be the convergence of two unrelated sequences, rather than the divergence from a common ancestor. Although we cannot formally distinguish between
sequence convergence and divergence at this level, it is less likely to be convergence since this would require the independent acquisition of the correct number of motifs in the same exact linear order. It is more likely that an OSM is the remnant of a common ancestral sequence whose regions of structural or functional importance have remained, while the unconstrained regions in between have changed.

Computational methods for detecting an OSM involve the comparison of multiple sequences. Methods for multiple sequence alignment are of two types, global and local. Global methods attempt to align the entire length of the protein sequences, while local alignment methods only locate the most similar regions of amino acids among the sequences. Since local alignment methods seek to locate the conserved amino acids that comprise an OSM, they can be used to detect potential homology among protein sequences that are less than 30% identical.

Once homology is established among a group of protein sequences, it can be predicted that the proteins maintain similar functions. In other words, the probable function of a protein can be deduced if it is found to share an OSM with a protein of known function. However, the putative function must be confirmed using biochemistry techniques. By using computational tools to determine which amino acids are essential to an OSM, “wet bench” biologists are given a starting point for structural and functional mutagenesis experiments.

On the other hand, just because two proteins possess similar functions does not mean they are homologous. The identification of an OSM among these types of sequences would support homology. However, in the case of proteins without an OSM, it would be more likely that the similar functions arose through convergent evolution rather than common ancestry. The strategy for establishing protein sequence homology and potential protein function is illustrated in Figure 1.

It is not surprising that the evolutionary relationships among proteins have a direct correlation to their function. Functional and structural constraints are the primary limiting factors of genome evolution. With the emergence of bioinformatics, we are finally able to investigate and reveal the complex relationships that exist between genome structure, function, and evolution.
Figure 1. Strategy for establishing protein sequence homology and potential protein function. Amino acid sequences that are greater than 30% identical are considered homologous according to the Dayhoff criteria for homology (Barker and Dayhoff, 1972). Sequences that are less than 30% identical must possess an Ordered-Series-of-Motifs (OSM) in order to demonstrate homology. Common function can be predicted by homology and confirmed by biochemical analysis. However, proteins that maintain similar functions are not always homologous. In this case, the detection of an OSM via local alignment is also used to support homology.
3 Thesis Objectives

An OSM is an invaluable instrument for biologists since it provides knowledge of potential protein structure, function, and evolutionary relationships. Thus, the methods used to locate an OSM are essential bioinformatic tools. In order to produce biologically meaningful results, it is important to use the most robust local alignment method. A major goal of this thesis is to determine the most reliable OSM-detection method via comparative analyses of the available computer programs. The performance of local alignment methods can be assessed by using benchmark data sets that were used in a previous study to evaluate sequence alignment methods [M.A. McClure et al., 1994]. The ability of the programs to correctly identify the OSM in the benchmark data is used to evaluate program performance. Chapters 1 and 2 of this thesis are assessments of program performance for local alignment methods. Chapter 3 is an application of the most robust motif-detection methods to address a biological hypothesis.

Chapter 1 is a pilot study of the comparative analysis. It evaluates the performance of seven different local alignment programs on a single data set consisting of twenty highly divergent protein sequences. A poster of this work was presented at the Pacific Symposium on Biocomputing, 1999, and Chapter 1 was published in the meeting Proceedings [J. Hudak and M.A. McClure, 1999]. Chapter 2 is an expanded version of the pilot study. It evaluates the sensitivity of motif-detection programs to a variety of data sets. By using unbiased and biased variable-sized data sets from five different protein families, a more accurate assessment of program reliability can be made. This analysis will also serve as a guide for biologists to choose the most reliable program for their study.

Chapter 3 discusses the application of local alignment methods to determine whether common ancestry can be supported among two groups of highly divergent RNA-dependent polymerases. These polymerases are functionally analogous, in that they both use an RNA template to synthesize nucleic acid (one group of polymerases produces RNA while the other produces DNA). However, the polymerase proteins vary in size from ~300 to ~2200 amino acids and share significantly limited sequence similarity. A series of four small regions of amino acids have been suggested to represent an OSM among these proteins [O. Poch et al., 1989]. Yet, statistical analyses of this and several other phylogenetic studies have raised question as to whether there is enough phylogenetic signal to claim homology between RNA-dependent polymerases
Chapter 3 is an attempt to address the hypothesis that all RNA-dependent polymerases are not related and the ongoing evolutionary debate by using the latest, and most reliable, statistically-based methods to locate a common OSM among the RNA-dependent polymerases.

References

11. O. Poch et al., The EMBO Journal, 8, 3867 (1989).
Abstract

The detection of motifs within and among families of protein sequences can provide useful information regarding the function, structure and evolution of a protein. With the increasing number of computer programs available for motif detection, a comparative evaluation of the programs from a biological perspective is warranted. This study uses a set of 20 reverse transcriptase (RT) protein sequences to test and compare the ability of 7 different computational methods to locate the ordered-series-of-motifs that are well characterized in the RT sequences. The results provide insight to biologists as to the usage, value, and reliability of the numerous methods available.

1 Introduction

Early work in protein pattern recognition suggested that islands of amino acids may be conserved in the same order of a given protein family [M.O. Dayhoff et al., 1983]. Today, a region of amino acids that is conserved throughout the evolution of a protein family is called a motif. Motifs can be present among protein sequences either as a set of unique motifs or as a set of repeated motifs. When motifs occur in a specific order among a set of sequences, they can be thought of as an ordered-series-of-motifs (OSM). [M.A. McClure, 1991] or protein signature. The designation of protein signature refers to the OSM that characterizes a particular family of proteins.
There are two aspects of motif detection worth clarifying. The first is the initial recognition of a unique motif pattern, or OSM, that defines a protein family. The second is the use of known motifs to identify potential functions in uncharacterized sequences. We are interested in new computational methods for the initial inference of an OSM. Our approach to motif detection is an attempt to find the OSM among highly divergent sequences in order to provide insight into the function, structure and evolution of the protein family.

OSMs are selectively constrained throughout the evolution of a protein family as a result of their importance to function and structure. Thus, an OSM can be defined in more than one biologically meaningful way. A functional OSM can be described by the residues of a catalytic site, e.g., the Asp-Asp (DD) motif of the reverse transcriptase (RT) protein sequence. An OSM may also define structural patterns, e.g., α-helices or β-sheets. A functional OSM can be superimposed on structural domains, e.g., the RT OSM location within the fingers, palm and thumb structural domains of the RT (figure 1) [L.A. Kohlstaedt et al., 1992]. Regardless of how the OSM is defined, function and structure is maintained only when all motifs of the OSM are present and in the appropriate order relative to one another.

In retroviruses, the RT constitutes one functional domain of the RNA-dependent DNA-polymerase (RDDP). The other domain is the ribonuclease-H (figure 1). Primary sequence analysis shows that all known RT sequences contain an ordered series of six characteristic motifs (figure 2) [M.A. McClure, 1993].

![Diagram of Reverse Transcriptase and Ribonuclease H](image)

Figure 1. The RNA-dependent DNA polymerase (RDDP) is comprised of two functional domains, RT and RH. The most highly conserved residues of the OSM of the RT functional domain [M.A. McClure, 1993] are placed within the structural domains (fingers, palm, fingers, palm, thumb, and connection) identified by the HIV-1 RT crystal [L.A. Kohlstaedt et al., 1992]. The most highly conserved residues of the ordered-series-of-motifs of the RH domain [M.A. McClure, 1991] are placed within the two RH structural domains based upon comparison of the HIV-1 and *E. coli* RH crystal structures [J.F.I. Davies et al., 1991].

The crystal structure of the RT reveals the location of the structural folds confirming the functional importance of the OSM [L.A. Kohlstaedt et al., 1992]. The individual motifs of the OSM have varying
levels of conservation. The order of conservation for the motifs, from high to low, is as follows: IV > II > VI > III > I or V. Since the OSM in the RT protein is well-characterized, the RT sequences can be used to evaluate the performance of motif detection methods.

<p>| | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>I</td>
<td>II</td>
<td>III</td>
<td>IV</td>
<td>V</td>
<td>VI</td>
</tr>
<tr>
<td>HT13</td>
<td>pvkKa--</td>
<td>t-IDLkdaaf</td>
<td>-LPQQ-fk</td>
<td>qYMDDI11</td>
<td>shGL--</td>
</tr>
<tr>
<td>NNV0</td>
<td>ikkK---</td>
<td>tiLDIgday</td>
<td>-LPQQ-wk</td>
<td>-YMDDIyi</td>
<td>qYMPP--</td>
</tr>
<tr>
<td>SFV1</td>
<td>pvpKp--</td>
<td>tttLDLncgf</td>
<td>-LPQQ-f1</td>
<td>aTVDDIyi</td>
<td>naGYVv</td>
</tr>
<tr>
<td>HERVC</td>
<td>pvpKp--</td>
<td>tCLDLkdaaf</td>
<td>-LPQR-fk</td>
<td>qTVDDR11</td>
<td>hvGIRc</td>
</tr>
<tr>
<td>GMM1</td>
<td>mvrKa--</td>
<td>ttkVDVraaf</td>
<td>-CPFF-la</td>
<td>aYLDILIi</td>
<td>--GLN--</td>
</tr>
<tr>
<td>GM17</td>
<td>v-pKKqd</td>
<td>ttIDLakgf</td>
<td>-NPPF-lk</td>
<td>vYLDIIiv</td>
<td>--NLK--</td>
</tr>
<tr>
<td>MDS1</td>
<td>lvpKKs1</td>
<td>sclDLmsgf</td>
<td>-LPPF-lk</td>
<td>lYMDDVv</td>
<td>--NLK--</td>
</tr>
<tr>
<td>MORG</td>
<td>vvrKk--</td>
<td>tttDLqngf</td>
<td>-APPF-fk</td>
<td>lYMDDIiv</td>
<td>--GLK--</td>
</tr>
<tr>
<td>CAT1</td>
<td>lvdKpdk</td>
<td>eqMDVktaf</td>
<td>kSLTG-lk</td>
<td>lYMDDMl</td>
<td>--EMR--</td>
</tr>
<tr>
<td>CMC1</td>
<td>titKrpe</td>
<td>hqMDVktaf</td>
<td>kAIYG-lk</td>
<td>lYMDDVv</td>
<td>---KR--</td>
</tr>
<tr>
<td>CST4</td>
<td>ftkKrng</td>
<td>t-LDIhaf</td>
<td>kALYG-lk</td>
<td>vYMDDCvi</td>
<td>inKLK--</td>
</tr>
<tr>
<td>C1095</td>
<td>ftcrKrdg</td>
<td>tGLDIssay</td>
<td>kSLTG-lk</td>
<td>lYMDDMl</td>
<td>itTLKk</td>
</tr>
<tr>
<td>NDMO</td>
<td>mihKt--</td>
<td>afLDIqqa</td>
<td>gVPQGSvl</td>
<td>tyADDTav</td>
<td>tsGL--</td>
</tr>
<tr>
<td>NL13</td>
<td>lipKp--</td>
<td>s-IDAekaf</td>
<td>gTRQGcpl</td>
<td>lPADDmiv</td>
<td>vsGYK--</td>
</tr>
<tr>
<td>NLOA</td>
<td>fipKaN--</td>
<td>afLDIegaf</td>
<td>gCPQGGvl</td>
<td>gYADDIVi</td>
<td>evGLN--</td>
</tr>
<tr>
<td>NTC0</td>
<td>vlrKp--</td>
<td>amLDGnay</td>
<td>gVRQGmv1</td>
<td>aYLDDTVv</td>
<td>alIIE--</td>
</tr>
<tr>
<td>ICD0</td>
<td>eipKp--</td>
<td>vdIDIg-gf</td>
<td>gTPQGg1</td>
<td>rYADDfki</td>
<td>rlDLDi</td>
</tr>
<tr>
<td>IAG0</td>
<td>fkkKrK--</td>
<td>ieGDIks-f</td>
<td>gVPQGGII</td>
<td>rYADDLvl</td>
<td>eiKTT1</td>
</tr>
<tr>
<td>ICS0</td>
<td>wipKp--</td>
<td>ldADIck-c</td>
<td>gTPQGVy</td>
<td>rYADDVvi</td>
<td>emGLLv</td>
</tr>
<tr>
<td>IPL0</td>
<td>yipKs--</td>
<td>leADI-r-gf</td>
<td>gVPQGGpi</td>
<td>rYADDFVV</td>
<td>srGLV1</td>
</tr>
</tbody>
</table>

Figure 2. The six motifs of the RT OSM are indicated by roman numerals (I-VI) [M.A. McClure, 1993]. The bold and capitalized letters represent the core amino acids of each motif used to score the programs in this study. Dashes represent gaps in the alignment. Abbreviations on the left side bar are defined in materials and methods.

With the increase in available sequence data, there has been an increase in computer programs created to define new motifs. Computational methods that attempt to identify an OSM without regard to the intervening regions are referred to as local alignment methods. Methods that attempt to align the entire length of a set of sequences are referred to as global alignment methods. A previous study of global and local methods revealed that global methods outperform local methods in identifying motifs [M.A. McClure et al., 1994]. Another comparative study of global methods and Hidden Markov Model (HMM) approaches concluded that HMMs were as good or better at motif detection than classical dynamic programming methods. Although HMMs display improved performance, they are not 100% accurate [M.A. McClure and
R. Raman, 1995; M.A. McClure, 1996]. With the increase in new computational methods for local alignment, a current comparative analysis is warranted. This study evaluates six motif-detection methods and the HMM approach to determine whether recently developed local alignment methods are superior to HMMs.

From a biologist's perspective, choosing a computational motif-detection method is not simple, especially with the many different methods available. Once a method has been chosen, how does one know what parameters should be altered to produce optimal results? Comparative analyses of computational methods assist biologists in choosing and using the best method for their studies.

2 Materials and Methods

All analyses were performed on a Sun SPARCstation Ultra 1 running SUN OS 5.6.

2.1 Biological data

The RT test sequences were obtained from GenBank, with the exception of one sequence (C1095) from the Saccharomyces Genome Database. Initially, more than 500 RT sequences were retrieved from the databases. Using a program that generates pairwise similarity scores based on the Needleman-Wunsch algorithm, [S.B. Needleman and C.D. Wunsch, 1970] and CLUSTER, an in-house hierarchical clustering method, 20 representative RT sequences were selected from this collection. The pairwise sequence identity among the test set of sequences ranges from 7-48%. As calculated by the Needleman-Wunsch algorithm, sequence similarity is also low for this data set. The data set includes an even distribution of RT sequences from the following groups: retroviruses (HT13, NVV0, SFV1, HERVC); gypsy retrotransposons (GMGl, GM17, MDGl, MORG); copia retrotransposons (CATl, CMC1, CST4, C1095); non-long terminal repeat retroposons (NDMO, NL13, NLOA, NTC0); and retropincons (ICO0, IAG0, ICS0, IPL0). GenBank accession numbers are L36905, M60610, X54482, M10976, M77661, X01472, X59545, Z27119, X53975, X02599, M94164, M22874, L19088, X60177, M62862, X98606, U41288, X71404, and Z48620. When necessary, the nucleic acid in the GenBank file was translated and the RT protein was obtained from the translation.

2.2 Motif-identification programs
Seven computer programs were included in this study (table 1.1). With the exception of SAM, all of these programs are local alignment methods that are not search engines for motif databases. Although SAM is a global alignment method, it is included in this study because it was found to perform at least as well as global methods that are better than local methods [M.A. McClure et al., 1994]. Brief descriptions of each program are provided below.

**BLOCKMAKER**, [S. Henikoff et al., 1995] the downloaded version, implements the Motif algorithm [R.F. Smith and T.F. Smith, 1990]. Motif searches the sequences for conserved triplets of amino acids, or blocks, that are separated by a user-specified length. If the triplet is found in enough sequences, an alignment is created that maximizes the block score. From the best alignments, the triplets are merged and the alignment is extended to get the highest score for the blocks.

**ITERALIGN** uses the symmetric-iterative protocol [L. Brocchieri and S. Karlin, 1998]. It starts by aligning the sequences according to the significant segment pair alignment method. Improved sequences and, eventually, consensus sequences are generated until they converge. Regions of amino acid similarity are derived from the alignment of the consensus sequences. The reported motifs are defined by a consensus residue and conservation index.

**MATCHBOX** implements a scanning algorithm [E. Depiereux et al., 1997]. It begins the search using a 9-residue running window that moves across the sequences in search of a match. A match is based on the number of identical amino acids and the sum of the distances observed between matched residues. A database of matches/boxes is created and boxes are deleted based on their length or selected based on the residual length and gap cost ratio.

The **PIMA** (Pattern-Induced Multi-sequence Alignment) program starts by constructing a binary tree based on pairwise similarity scores [R.F. Smith and T.F. Smith, 1992]. The tree is reduced to one pattern by replacing nodes with a common pattern node that is generated by an alignment based on the Smith-Waterman (SW) algorithm [T.F. Smith and M.S. Waterman, 1981]. Common patterns are constructed from the alignment using amino acid class-covering hierarchy patterns.

The **PROBE** program implements the SW algorithm that performs transitive searches to find regions of sequence similarity [A.F. Neuwald et al., 1997]. The sequences collected from this search are purged to
eliminate unequal representation of the data and then aligned co-linearly using the Gibbs sampling algorithm [C.E. Lawrence et al., 1993; A.F. Neuwald et al., 1995]. The Gibbs sampling algorithm starts at a random position for all of the sequences except one. The excluded sequence is aligned to the others. This process is reiterated until the information content score is maximized. After Gibbs sampling, a genetic algorithm is used to recombine a randomly selected alignment and choose the best alignment produced. This alignment is used to search for more sequences, which are included in another iteration starting with the Gibbs sampling step, until no more new sequences are found.

Both MEME (Multiple Expectation Maximization for Motif Elicitation) and SAM (Sequence Alignment and Modeling) locate motifs by estimating the parameters for a model that maximizes the likelihood of the data. MEME starts by breaking up the data into overlapping sequences of specified length [T.L. Bailey and C. Elkan, 1994]. The MM (Mixture Model) algorithm creates a finite mixture model of the new data set that consists of two components, the motifs and the motif-background probabilities. The EM (Expectation Maximization) algorithm estimates and maximizes the expected log likelihood value of the model parameters.

The SAM program is a linear HMM that implements the Baum-Welch algorithm [A. Krogh et al., 1994; R. Hughey and A. Krogh, 1996]. The estimated parameters are the transition and observation probabilities. Once the model converges, a multiple alignment can be created and motifs detected.

Several programs are not included in this study for a variety of reasons. In a previous study, MACAW [G.D. Schuler et al., 1991] and PRALIGN [M.S. Waterman and R. Jones, 1990] were found to give suboptimal results [M.A. McClure et al., 1994]. MOTIF [H.O. Smith et al., 1990] was not included because it is only available for DOS and a modified version, Motifj, is implemented in the BLOCKMAKER program. The FILTER program was not suitable for this study due to a maximum sequence limit of 16 [M. Vingron and P. Argos, 1990; M. Vingron and P. Argos, 1991]. PRATT was not included because detected motifs are based on PROSITE patterns [I. Jonassen et al., 1995; A. Brazma et al., 1996]. The EMOTIF program did not suit this study because it requires the input sequences to be aligned [C.G. Nevill-Manning et al., 1997]. The TEIRESIAS program is not readily available [I. Rigoutsos and A. Floratos, 1997].

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
Initially, the GIBBS program was included. However, our analysis of GIBBS clearly indicates that the authors' most recent program, PROBE, performs better.

All programs were initially run at the default parameter settings to establish baseline results. Range studies for user-specified parameter options were conducted for all methods analyzed. Parameters were changed according to the description of their function and default values. A range of values for each parameter was chosen to determine the effects on motif detection. The best results for each program were determined by a motif-scoring scheme.

2.3 Motif Scoring

Program performance was assessed by manually scoring the detected motifs. Individual program scores consist of six values, one for each motif of the OSM. The value for each motif is equal to the number of sequences correctly identified, with the highest score being the number of sequences (20) used to test the programs. The correct identification of a motif is based on the residues that represent the motifs (figure 2).

3 Results

The best results from these studies are presented in table 1.2. Of all the programs evaluated, ITERALIGN, MEME, PROBE, and SAM were the only ones that detected the entire OSM (figure 2). The highly conserved motif IV was the only pattern detected to some degree by all methods. The degree to which other motifs could be detected varied from method to method.

The webserver version of BLOCKMAKER implements both the MotifJ and Gibbs sampling algorithms, without the option of changing parameters. The results for either algorithm are not any better than the downloaded version of MotifJ with parameter changes. The best run of MotifJ only detects the two most highly conserved motifs (figure 2; II and IV), with a high score of 19 for motif IV. The ITERALIGN program finds the entire OSM with motif VI (figure 2) having the highest occurrence of detection at 14 sequences. Parameter changes are not available for the webserver version of MATCHBOX. The only result obtained from this program is the detection of the most conserved motif IV in all 20 sequences. The highest scores (20) for MEME are for the two most conserved motifs (figure 2; II and IV). MEME also reports high scores for motifs I, III, and VI. PIMA detects all of the motifs except the highly divergent
Table 1.1: Computational Motif-Detection Programs

<table>
<thead>
<tr>
<th>PROGRAM</th>
<th>ALGORITHM*</th>
<th>MATRIX</th>
<th>INDEL PENALTY*</th>
<th>RUN TIME</th>
<th>USER SPECIFICATIONS*</th>
</tr>
</thead>
<tbody>
<tr>
<td>BLOCKMAKER</td>
<td>Motiff</td>
<td>PAM 250</td>
<td>none</td>
<td>~1m</td>
<td>N</td>
</tr>
<tr>
<td>ITERALIGN</td>
<td>SI</td>
<td>PAM 250</td>
<td>C</td>
<td>~1h40m</td>
<td>N</td>
</tr>
<tr>
<td>MATCHBOX</td>
<td>Scanning</td>
<td>BLOSUM 62</td>
<td>none</td>
<td>~45m</td>
<td>N</td>
</tr>
<tr>
<td>MEME</td>
<td>MM/EM</td>
<td>PAM 250</td>
<td>none</td>
<td>~2m</td>
<td>Y</td>
</tr>
<tr>
<td>PIMA</td>
<td>SW</td>
<td>AACH*a</td>
<td>I + E</td>
<td>~2m</td>
<td>N</td>
</tr>
<tr>
<td>PROBE</td>
<td>SW+G+GA</td>
<td>PAM 250</td>
<td>I + E</td>
<td>~2h30m</td>
<td>N</td>
</tr>
<tr>
<td>SAM</td>
<td>BW</td>
<td>none</td>
<td>none</td>
<td>~2h20m</td>
<td>N</td>
</tr>
</tbody>
</table>

*Algorithms are: SI = Symmetric-Iterative protocol; MM = Mixture Model that uses (EM) Expectation Maximization; SW = Smith-Waterman; G = Gibbs Sampling; GA = Genetic Algorithm; and BW = Baum Welch. *AACH = Amino Acid Cluster Hierarchy (patgen, class 1; and class 2). *The insertion-deletion (indel) penalties are: C = constant and I + E = initial + extension. #MOTIFS = number of motifs to be detected; WIDTH = width of motifs to be detected; # SEQUENCES = number of sequences that contain the motif; N = user cannot specify; Ne= user cannot specify and program excludes sequences; Ni = user cannot specify, but program automatically includes all sequences; and Y = user can specify, but it is not required.

Table 1.2: Motif Scores and Parameter Options

<table>
<thead>
<tr>
<th>PROGRAM</th>
<th>I(1)</th>
<th>II(3)</th>
<th>III(4)</th>
<th>IV(5)</th>
<th>V(3)</th>
<th>VI(3)</th>
<th>PARAMETERS</th>
</tr>
</thead>
<tbody>
<tr>
<td>BLOCKMAKER</td>
<td>0</td>
<td>18</td>
<td>0</td>
<td>19</td>
<td>0</td>
<td>0</td>
<td>run type=1; sign=5; dist=5* (5-30)</td>
</tr>
<tr>
<td>ITERALIGN</td>
<td>10</td>
<td>9</td>
<td>8</td>
<td>13</td>
<td>12</td>
<td>14</td>
<td>ltw=0.99* (0.0-0.99)</td>
</tr>
<tr>
<td>MATCHBOX</td>
<td>0</td>
<td>0</td>
<td>20</td>
<td>0</td>
<td>20</td>
<td>10</td>
<td>mod oops; nmotifs=10; maxw=10*</td>
</tr>
<tr>
<td>MEME</td>
<td>16</td>
<td>20</td>
<td>19</td>
<td>20</td>
<td>10</td>
<td>17</td>
<td>mod oops; nmotifs=10; maxw=10*</td>
</tr>
<tr>
<td>PIMA</td>
<td>18</td>
<td>20</td>
<td>8+12</td>
<td>20</td>
<td>0</td>
<td>15</td>
<td>default with class 2 matrix</td>
</tr>
<tr>
<td>PROBE</td>
<td>18</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>14</td>
<td>20</td>
<td>S=500*</td>
</tr>
<tr>
<td>SAM</td>
<td>10+2</td>
<td>15</td>
<td>8+5+3+2</td>
<td>10+3+2</td>
<td>9+2</td>
<td>6+2+2+2</td>
<td>iw=2; FIMs @ 10,20,30,40,50*</td>
</tr>
</tbody>
</table>

Roman numerals indicate motifs and values in parenthesis indicate number of amino acids scored for in each motif. Values in the columns indicate the number of sequences in which the motif was correctly identified. Some methods find correct matches in more than one subset of the data without correct alignment of these subsets to one another, indicated by more than one result per motif. The parameter column indicates the changes which gave the best results. Values in parenthesis in this column indicate the range over which a parameter was tested. *run type = 1 is non-iterative mode; sign = significance level; and dist = search width. *ltw = weight assigned to lower threshold hits. *no parameter changes available on the webserver. *mod oops = motif distribution equals one occurrence per sequence; nmotifs = number of motifs to find; maxw = maximum motif width to be detected. *S = level at which to purge similar sequences. *iw = internal weight; FIMs = free insertion modules inserted at these positions; other parameters were changed according to (M.A. McClure and R. Raman, 1995, M.A. McClure, 1996).
motif V. Motifs II and IV are detected in all 20 sequences while motif III is detected as two different unaligned subsets. The SAM method locates the entire OSM. All motifs, except motif II with a score of 15, are detected as unaligned subsets.

PROBE has the highest occurrence of detection for the entire OSM. These results were obtained after running the program several times under the default parameters. Differences in the results of these runs are due to different random seeds, or starting points for the program. The best random seed runs find the four most conserved motifs, II, III, IV, and VI, for all 20 sequences. Motif I, a single residue motif, was found in 18 out of 20 sequences. In two of the copia elements (figure 2; CAT1 and CMC1), the lysine residues were not correctly aligned. The highly divergent motif, V, was correctly identified in 14 out of 20 sequences. This motif was not correctly identified in any of the copia sequences and two non-long terminal repeat elements. Nonetheless, this study clearly indicates that the PROBE program outperforms all other methods (table 1.2).

Another strength of PROBE is that the results are reported as collinear blocks of motifs. Since collinearity is definitive of an OSM and block format is readily analyzed, this makes the result format of PROBE highly efficient. Other methods, such as BLOCKMAKER, MATCHBOX, MEME also display the results in a block format. However, MEME has a tendency to report motifs regardless of their position in the sequence. This is useful when looking for repetitive motifs throughout a set of sequences, but it does not maintain the collinearity of an OSM. Collinearity of BLOCKMAKER and MATCHBOX cannot be determined since the entire OSM was not detected. Methods, such as ITERALIGN, PIMA, and SAM display the results as an alignment of the data set. The alignments are collinear, but difficult to analyze. The motifs of the ITERALIGN alignment are difficult to score because the program allows gaps and insertions within the motif. PIMA reports motifs as a consensus sequence using 60 symbols that represent the different types of substitutions per position. This is difficult to analyze without a symbol legend and an alignment of the sequences to the consensus sequence. Since SAM is not meant for local alignment, it requires much effort to search the entire global alignment for the regions of aligned motifs.
4 Discussion and Future Studies

4.1 Discussion

The purpose of this study is to find the most reliable method of motif detection currently available. Motif-detection programs are sensitive to the degree of sequence similarity among the analyzed data. A program may be robust for analysis of similar sequences, but inadequate for a highly divergent set of sequences. Methods that are able to identify motifs among highly divergent sequences are more reliable than those methods that cannot.

While all programs analyzed were able to detect the most highly conserved motif IV, four of the methods (ITERALIGN, MEME, SAM, and PROBE) were able to detect the entire OSM. All other methods (BLOCKMAKER, MATCHBOX, and PIMA) were not able to identify motif V because it is one of the most divergent motifs. This indicates that although conserved motifs are easily detected, only the most robust methods will be able to detect an entire OSM that also contains divergent motifs. These results demonstrate that motif-detection programs are sensitive to the degree of sequence similarity.

Of all methods evaluated, PROBE performed the best at detecting the OSM in the highly divergent RT sequences. The PROBE program correctly located the four most conserved motifs and was able to detect the two divergent motifs with considerable accuracy. The error in detecting motif I for two sequences is surprising, because the two correct residues are only out of column register by 1 and 2 positions, respectively. PROBE is a robust method for detecting an OSM even without making any parameter changes. This is because it is designed to locate motifs as they are found in an OSM, collinearly among a set of sequences. In this study, PROBE detected more than the six collinear motifs of the OSM. This is not an inaccuracy of the method, but a display of PROBE's superior performance. The additional motifs detected are actually recognized sub-motifs in the RT sequences [M.A. McClure, 1993]. PROBE detects both motifs and sub-motifs without any specification from the user. This is useful when the number of motifs is not known. MEME, on the other hand, requires the number of motifs to be specified. MEME performance is improved when the specified number of motifs is greater than the actual number of motifs. This generates some sub-motif detection, but not as accurately as PROBE.
Although MEME has scores almost as high as PROBE, a recent analysis of both MEME and PROBE using a data set of 497 RT sequences demonstrated that PROBE is still able to outperform MEME [M.A. McClure et al., 1998]. The data set used in the study contained an unequal distribution of sequence similarity, which resulted in some sequences, or motifs, to be over-represented. MEME will get trapped in a local optima by recognizing the biased motif as the correct motif and considering any divergent form incorrect. This results in the exclusion of the entire sequence, thus reducing the score and producing biologically uninformative results. PROBE, however, handles a biased data set by eliminating redundant sequences or sequences that are too similar to each other. Purging of sequences produces an equally distributed data set representative of the entire 497 sequences from which it can detect informative motifs with a high score.

A recent comparison of several methods that are also included in this study (ITERALIGN, BLOCKMAKER, MEME, and PIMA) came to similar conclusions about program performance [L. Brocchieri and S. Karlin, 1998]. ITERALIGN and PIMA were able to find the entire OSM of the Rec-A sequences. MEME displayed better performance than BLOCKMAKER. Contrary to our experience with MATCHBOX (table 2), the program correctly identified 6 out of 7 Rec-A motifs. With the exception of MATCHBOX, program performance was comparable between the two studies even though our study used more divergent sequences with shorter motifs.

Our study has elucidated that PROBE is a superlative method currently available for the detection of an OSM.

4.2 Future Studies

Future studies will attempt to find an OSM among a larger group of highly divergent protein sequences that share analogous function. In addition to the RT domain sequences, this data set will include sequences from the RNA-dependent RNA polymerases (RDRP) found in all other RNA viruses (e.g., HIV, Ebola, and Measles). In this case, some sequences of the data set cannot be statistically shown to share common ancestry. This raises the question of whether the observation of an OSM is due to common ancestry versus sequence convergence.
Whether or not common ancestry is responsible for the limited sequence similarity detected between the RT and RDRP sequences is an open question. Several studies suggest a common ancestry among all RNA-dependent polymerases [P. Argos, 1988; O. Poch et al., 1989; M. Delarue et al., 1990]. These studies were prompted by the detection of the highly conserved Asp-Asp motif in the RDRP of polio [G. Kamer and P. Argos, 1984] which is also found in retroviruses. Although the Asp-Asp motif is conserved among some RDRPs and the RT domain, there are only three additional residues found in common among these proteins, whose lengths vary from approximately 300 to 2000 amino acids. A recent reevaluation of the multiple alignments that suggested these relationships concludes that there is a lack of statistically significant signal remaining among the sequences to claim common ancestry [P.M. de A. Zanotto et al., 1996].

A more robust motif-detection algorithm may aid in addressing the ancestry versus convergence question regarding RDRPs and the RT domain of RDDPs. Future studies will use the most reliable motif-detection method, as determined from this study, to locate a potential OSM shared among the RDRPs and the RT domain. Finding a reliable OSM would assist in creating separate hidden Markov models (HMMs) representing the sequences of both the RDRPs and the RT domain based on a new OSM-anchoring approach [M.A. McClure and J. Kowalski, 1999]. By comparing the protein sequences of one group to the model of the other, these HMMs can be used to evaluate the possibility of common ancestry between these sequences. If the probability is significant, then it would be worthwhile to construct an HMM representing both the RDRP and RT sequences. This approach could provide statistical evidence to either support or refute common ancestry among all RNA-dependent polymerases.

Acknowledgments

This work was supported by a grant to M.A.M. from the NIH, AI 28309. We thank John Kowalski for technical computer assistance, Seanna Corro, Micah Potter, Kevin Richter, and Steven de Belle for their invaluable assistance in discussion and manuscript critique.

References

CHAPTER 2

A PERFORMANCE AND RELIABILITY
EVALUATION OF SEVERAL LOCAL
ALIGNMENT METHODS

Abstract

Computational methods for the local alignment of protein sequences have increased in number. Inevitably, these programs vary in their ability to correctly identify the regions of local alignment, or motifs. For any biological analysis, it is important that the most accurate method(s) be used. This study evaluates the performance and reliability of six local alignment programs using a range of benchmark data sets. The results provide insight to biologists for choosing which local alignment methods to use. This study also demonstrates the importance of comparative testing of bioinformatic tools.

1 Introduction

As biological sequence data rapidly accumulate, so do the number of programs designed for sequence data analysis. Inevitably, some of these methods are more accurate than others. Individual program performance may also vary with different numbers and types of data. With this in mind, how are biologists to choose the most accurate and reliable computational method for their analyses? Comparative evaluations of the computational tools for sequence analysis are essential to the field of bioinformatics. These comparisons give biologists a foundation for choosing an appropriate computational method for their analyses, and provide feedback to program developers as to the accuracy of their methods.

22

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
One of the most important tools for the analysis of sequence data is alignment methods. There are two types of multiple-sequence alignment methods, global and local. Global alignment methods seek to align the entire length of the sequences, while local alignment methods only attempt to locate and align the most similar regions of the sequences.

It has been almost two decades since it was first suggested that protein sequences of the same family contain regions of conserved amino acids [M.O. Dayhoff, 1983]. These regions of approximately 1-9 amino acid residues are called motifs. When multiple motifs are present in a specific colinear order among a group of sequences, they are referred to as an ordered-series-of-motifs, or OSM [M.A. McClure, 1991]. The amino acids of an OSM are conserved throughout evolution because they are essential to the structural and functional integrity of the protein. The regions between motifs, or motif-intervening-regions (MIRs), may evolve more rapidly because they are less restricted by protein function or structure. The evolutionary constraint on an OSM allows for it to be designated as a protein family signature. By locating a common OSM among a group of protein sequences, biologists are able to make inferences about protein structure, function and evolution. The accuracy of these inferences is increased when the most robust local alignment program is used to locate the OSM. To date, efforts are still being made to produce a general method that is accurate and reliable on most data sets. However, many newly developed methods are not any better than methods that already exist. Most program developers do not compare their methods to existing programs before making the method available to the public. Of all the methods included in this study, only the literature for MEME and ITERALIGN mention program performance in comparison to other similar programs. MEME was briefly compared to GIBBS sampler program [T.L. Bailey and C. Elkan, 1994], while the performance of ITERALIGN was compared to that of BLOCKMAKER, MATCHBOX, MEME, and PIMA [L. Brocchieri and S. Kargin, 1998]. However, the ITERALIGN study was conducted on a single data set with higher sequence similarity than the data sets used in this analysis.

The focus of this study is to test and compare the performance and reliability of six local alignment programs using a range of benchmark data sets whose OSMs are well-characterized. It should be noted that this study evaluates local alignment methods that are used for the initial recognition of an OSM among a group of sequences, not those that search motif databases to locate known motifs in an uncharacterized
sequence. The results of this study can guide biologists in their choice of local alignment methods. This study also demonstrates the importance of evaluating every available method before starting any biological analysis.

2 Methods and Materials

All analyses were performed on a Sun SPARCstation Ultra 1 running SUN OS 5.6.

2.1 Biological data

The protein sequence data sets used in this study are from five different protein families: hemoglobin (GLOB), kinase (KIN), aspartic acid protease (PRO), ribonuclease H (RH), and reverse transcriptase (RT). Sequences from these families are established benchmark data sets with well-characterized OSMs that are used for motif-detection studies [M.A. McClure et al., 1994]. The numbers of motifs per OSM that were used to test the programs are as follows: GLOB=5, KIN=8, PRO=3, RH=4, and RT=6. The number of amino acids per motif ranges from 1-9 residues. This study used the following number of amino acids per motif for each data set: GLOB=(7,5,5,5,3), KIN=(6,1,1,9,3,3,8,1), PRO=(3,5,3), RH=(3,1,3,5), and RT=(1,3,4,5,3,3). These benchmark OSMs serve as answer keys for evaluating program performance.

For each protein family, both small and large data sets were used. By using multiple types of data sets and a variable number of sequences, I was able to assess the sensitivity and reliability of each program to a range of data. The small data sets contain an even distribution of sequences, based on sequence similarity. The large data sets, however, are biased, i.e., some subsets of similar sequences may be present in greater numbers than other subsets. Realistically, these data sets are the type generated experimentally and obtained through database searching. The small data sets, GLOB12, KIN12, PRO12 and RH12 (where 12 is the number of sequences in the data set), are the same data sets that were used in an earlier comparative analysis [M.A. McClure et al., 1994]. The large data sets, GLOB174, KIN186, PRO114, RH169, were derived by random sampling of larger data sets from another study [M.A. McClure 1996]. Both RT data sets (RT20 and RT178) were obtained from GenBank and the Saccharomyces Genome Database. Initially, more than 500 RT sequences were retrieved from the databases. Random sampling of these sequences was used to
create the large RT data set. Also from the group of >500 sequences, 20 representative sequences (same
sequences as in Chapter 1) were selected using a program that calculates pairwise similarity scores from an
alignment generated by the Needleman-Wunsch algorithm [S.B. Needleman and C.D. Wunsch, 1970], and
CLUSTER, an in-house hierarchical clustering method. The percentage of pairwise sequence identity is low
for all data sets. The sequence similarity, based on the probability of amino acid substitutions and the ease
of converting from one codon to another, is also low for all data sets. Conversely, the average distance
values are high for all data sets. Distance is a measure of difference between the sequences that takes into
account the probability of amino acid substitutions and the ease of converting from one codon to another.

Table 2.1 gives the range and average sequence length, percent identity, and distance value for each data set.

Appendix I contains the OSMs for the GLOB12, KIN12, PRO12, and RH12 data sets. These data sets are
from the McClure, 1994, study and are available at the following web address: ftp://ftp.embl-
heidelberg.de/pub/databases/embl/align/ds16117.dat. The OSM and GenBank accession numbers for the
RT20 data set are given in Chapter 1. The sequences and OSMs for the large data sets will soon be
available via the Internet.

<table>
<thead>
<tr>
<th>DATA SET</th>
<th>SEQUENCE LENGTH</th>
<th>PERCENT IDENTITY²</th>
<th>DISTANCE³</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Range</td>
<td>Average</td>
<td>Range</td>
</tr>
<tr>
<td>GLOB12</td>
<td>141-153</td>
<td>147</td>
<td>13.5-83.7</td>
</tr>
<tr>
<td>KIN12</td>
<td>255-340</td>
<td>273</td>
<td>16.2-44.0</td>
</tr>
<tr>
<td>PRO12</td>
<td>98-160</td>
<td>127</td>
<td>9.0-72.0</td>
</tr>
<tr>
<td>RH12</td>
<td>126-158</td>
<td>141</td>
<td>8.7-40.9</td>
</tr>
<tr>
<td>RT20</td>
<td>297-412</td>
<td>348</td>
<td>10.6-39.6</td>
</tr>
<tr>
<td>GLOB174</td>
<td>115-161</td>
<td>145</td>
<td>9.7-99.3</td>
</tr>
<tr>
<td>KIN186</td>
<td>246-409</td>
<td>286</td>
<td>9.4-99.2</td>
</tr>
<tr>
<td>PRO114</td>
<td>97-150</td>
<td>108</td>
<td>6.6-99.2</td>
</tr>
<tr>
<td>RH169</td>
<td>122-246</td>
<td>144</td>
<td>5.3-99.5</td>
</tr>
<tr>
<td>RT178</td>
<td>288-434</td>
<td>347</td>
<td>10.2-99.7</td>
</tr>
</tbody>
</table>

²Percent identity is the percentage of identical amino acid residues among all sequence pairs.
³Distance (D) is a measure of difference between the sequences that takes into account the probability of amino acid substitutions and the ease of converting from one codon to another; $D = -\ln[(S_{\text{real}} - S_{\text{random}})/(S_{\text{chemical}} - S_{\text{random}})] \times 100$, where $S =$ similarity value. The range and average of sequence length, percent identity, and distance is given for each data set.
2.2 Assessment of program performance

Program performance was assessed by quantitative and qualitative criteria. The quantitative criterion is a measurement of how accurately a program can locate motifs. For each data set, this value was determined by comparing program results to the OSM key and assigning a motif-detection score. Individual program scores consist of several values, one for each motif of the OSM. Each value represents the number of sequences in which the motif was correctly identified. In some cases, a single motif was detected as several unaligned subsets (indicated by more than one result per motif). The total number of motifs reported by each program was also recorded.

To compare program performance for each small data set, a total score was assigned to each program. Total score was calculated by summing individual motif score values. In the case of multiple subsets per motif, the subset that contained the highest number of sequences was used for the total score summation. Although the use of total score masks information about a program’s ability to locate different types of individual motifs, it does provide a general overall score that is useful for comparing one program to another.

For the large data set analyses, it was necessary to record the total number of sequences reported since all of the input sequences were usually not included in the results. As a result, individual program scores also include the percentage of reported sequences in which the motif was correctly identified and the percentage of input sequences in which the motif was correctly identified.

In some cases, more than one run produced the highest motif-detection score. In this situation, it was necessary to use qualitative criteria in order to choose the best program run. For example, although two runs may produce the same score, the most informative result contains less false positives and is reported as the best.

Other types of qualitative criteria were taken into account when evaluating the reliability and user-friendliness of each program. As mentioned, the number of false positives reported by a program was considered. Programs that generate excessive false positives are less reliable and less informative. The parameterization required for the highest scoring runs was also a criteria used for evaluating the programs. Programs that consistently produce the highest scores with a fixed set of parameters are more reliable than
programs that produce high scores under a variety of different parameters. User-friendliness of each program was judged by the amount of user effort required to run the program and analyze the results.

2.3 Motif-identification programs

Six local alignment methods are included in this study: BLOCKMAKER, ITERALIGN, MATCHBOX, PIMA, PROBE and MEME. All of the methods, except MATCHBOX, are available for downloading through the Internet. Brief descriptions of each program are provided below.

BLOCKMAKER, version 9.0, can be downloaded from the Internet or used via a webserver [S. Henikoff et al., 1995]. The downloaded version implements the Motif algorithm [R.F. Smith and T.F. Smith, 1990]. Motif searches the sequences for conserved triplets of amino acids that are separated by a user-specified length. If the triplet is found in enough sequences, an alignment is created that maximizes the block score. From the best alignments, the triplets are merged and the alignment is extended to get the highest score for the blocks.

ITERALIGN, version 4.1.3 for SunOS, uses the symmetric-iterative protocol [L. Brocchieri and S. Karlin, 1998]. It starts by aligning the sequences according to the significant segment pair alignment method. Improved sequences and, eventually, consensus sequences are generated until they converge. Regions of similarity are derived from the alignment of the consensus. These regions are defined by a consensus residue and conservation index.

MATCHBOX, versions 1.2 and 1.3, is only available for use through a webserver [E. Depiereux et al., 1997]. This program implements a scanning algorithm. It begins the search using a 9-residue running window that moves across the sequences in search of a match. A match is based on the number of identical amino acids and the sum of the distances observed between matched residues. A database of matches/boxes is created and boxes are deleted based on their length or selected based on the residual length and gap cost ratio.

MEME (Multiple Expectation Maximization for Motif Elicitation), version 2.2., locates motifs by estimating the parameters for a model that maximizes the likelihood of the data [T.L. Bailey and C. Elkan, 1994]. First, the data is broken up into overlapping sequences of specified length. The MM (Mixture
Model) algorithm creates a finite mixture model of the new data set that consists of two components, the motifs and the motif-background probabilities. The EM (Expectation Maximization) algorithm estimates and maximizes the expected log likelihood value of the model parameters.

The PIMA (Pattern-Induced Multi-sequence Alignment), version 1.40, program starts by performing pairwise comparisons of the sequences [R.F. Smith and T.F. Smith, 1990]. The pairwise similarity scores are then used to cluster the sequences by two different linkage rules: maximal linkage (ML) and sequential branching (SB) [R.F. Smith and T.F. Smith, 1992]. In ML, the clustered scores are used to construct a binary tree. The tree is reduced to one pattern by replacing nodes with a common pattern node that is generated by an alignment based on the Smith-Waterman (SW) algorithm [T.F. Smith and M.S. Waterman, 1981]. In SB, similarity scores are sorted high-to-low. The first sequence from the highest scoring pair is used as a reference sequence. The remaining sequences are sequentially clustered according to their similarity to the reference sequence and common patterns are created. For both ML and SB, common patterns are constructed from the alignment using amino acid class-covering hierarchy patterns.

The PROBE, version 1.0, program implements the SW algorithm that performs transitive searches to find regions of sequence similarity [A.F. Neuwald et al., 1997]. The sequences collected from this search are purged to eliminate unequal representation of the data and then aligned co-linearly using the Gibbs sampling algorithm [C.E. Lawrence et al., 1993; A.F. Neuwald et al., 1995]. The Gibbs sampling algorithm starts at a random position for all of the sequences except one. The excluded sequence is aligned to the others. This process is reiterated until the information content score is maximized. After Gibbs sampling, a genetic algorithm is used to recombine a randomly selected alignment and choose the best alignment produced. This alignment is used to search for more sequences, which are included in another iteration starting with the Gibbs sampling step, until no more new sequences are found.

Several programs are not included in this study for various reasons. In a previous study, MACAW [G.D. Schuler et al., 1991] and PRALIGN [M.S. Waterman and R. Jones, 1990] were found to give sub-optimal results [M.A. McClure et al., 1994]. MOTIF [H.O. Smith et al., 1990] was not included because it is only available for DOS and a modified version of the algorithm, Motifj, is implemented in the BLOCKMAKER program. The FILTER program was not suitable for this study due to a maximum
sequence limit of 16 [M. Vingron and P. Argos, 1990; M. Vingron and P. Argos, 1991]. PRATT was not included because detected motifs are based on pre-defined patterns, which are not suitable for the data sets being used [I. Jonassen et al., 1995; A. Brazma et al., 1996]. The EMOTIF program did not suit this study because it requires the input sequences to be aligned [C.G. Nevill-Manning et al., 1997]. The TEIRESIAS program is not readily available [I. Rigoutsos and A. Floratos, 1998]. Initially, the GIBBS program was included. However, our analysis of GIBBS clearly indicates that the authors’ most recent program, PROBE, performs better.

2.4 Test implementation

For the small data sets, all six programs were initially run at the default parameter settings to establish baseline results. For each program, user-specified parameter options were altered in order to determine their effects on program performance and to establish which parameter values produce the best results. Parameters were changed according to the description of their function and default values. A range of values was used for each parameter that was changed. Since PROBE uses a random seed for each run, several default runs were performed to determine the effects of different seeds. The seed that produced the highest scoring results was fixed in order to test the effects of the other parameters.

For the large data set analysis, the three highest scoring programs from the small data set analysis were evaluated. These programs (MEME, PIMA and PROBE) were first run at default parameter settings. The parameter changes that produced the highest score for each small data set were also used in the analysis of the corresponding large data set.

For all programs and data sets, the parameter values that produced the highest motif-detection scores are indicated in Tables 2.2-2.6 and 2.8-2.12.

3 Results

3.1 Small data set analyses

The highest scoring results from the small data sets are in Tables 2.2-2.6, along with the parameters used to produce these results.
<table>
<thead>
<tr>
<th>PROGRAM</th>
<th>I</th>
<th>II</th>
<th>III</th>
<th>IV</th>
<th>V</th>
<th>TOTAL SCORE&lt;sup&gt;b&lt;/sup&gt;</th>
<th>TOTAL MOTIFS&lt;sup&gt;c&lt;/sup&gt;</th>
<th>PARAMETERS&lt;sup&gt;d&lt;/sup&gt;</th>
</tr>
</thead>
<tbody>
<tr>
<td>BLOCKMAKER</td>
<td>12</td>
<td>3+2+5</td>
<td>8+4</td>
<td>12</td>
<td>11</td>
<td>48</td>
<td>2</td>
<td>run type=3; sign=(3,5); dist=10</td>
</tr>
<tr>
<td>ITERALIGN</td>
<td>12</td>
<td>11</td>
<td>12</td>
<td>12</td>
<td>12</td>
<td>59</td>
<td>9</td>
<td>w=1; PAM 250 matrix</td>
</tr>
<tr>
<td>MATCHBOX</td>
<td>0</td>
<td>11</td>
<td>12</td>
<td>0</td>
<td>0</td>
<td>23</td>
<td>2</td>
<td>version 1.2; blosum 62 matrix</td>
</tr>
<tr>
<td>MEME</td>
<td>11</td>
<td>10</td>
<td>10</td>
<td>11</td>
<td>12</td>
<td>54</td>
<td>7</td>
<td>mod oops; nmotifs=10</td>
</tr>
<tr>
<td>PIMA</td>
<td>12</td>
<td>12</td>
<td>12</td>
<td>12</td>
<td>11</td>
<td>59</td>
<td>13</td>
<td>SB; patgen matrix</td>
</tr>
<tr>
<td>PROBE</td>
<td>12</td>
<td>11</td>
<td>12</td>
<td>9+3</td>
<td>12</td>
<td>56</td>
<td>12</td>
<td>S=700</td>
</tr>
</tbody>
</table>

<sup>a</sup>Roman numerals represent each motif of the OSM and values under each roman numeral indicate the number of sequences in which the motif was correctly identified (See Appendix 1 for OSM keys). Some methods find correct matches in more than one subset of the data without correct alignment of these subsets to one another, indicated by more than one result per motif. <sup>b</sup>TOTAL SCORE is the sum of all individual motif scores. In the case of multiple subsets per motif, the highest number per column was used for this summation. TOTAL SCORE is used to numerically rank the programs, with the highest score being the better score. <sup>c</sup>TOTAL MOTIFS is the number of regions reported as a single motif. For MEME, this is the actual number of motifs the program reported before all of the motifs of the OSM were located. The total number of motifs reported by MEME was specified by the user (nmotif parameter). For PIMA, TOTAL MOTIFS is the number of amino acid residues reported as being part of a motif. <sup>d</sup>The PARAMETER column indicates the changes from the default values which gave the best results. Abbreviations in this column are defined as follows: run type = 3 is iterative mode; sign = significance level; and dist = search width; w = weight sequences according to sequence maxscore; mod oops = motif distribution equals one occurrence per sequence; nmotifs = number of motifs to find; SB = sequential branching clustering method; and S = level at which to purge similar sequences.
Table 2.3: Program Results and Parameters for KIN12 Data Set.

<table>
<thead>
<tr>
<th>PROGRAM</th>
<th>I</th>
<th>II</th>
<th>III</th>
<th>IV</th>
<th>V</th>
<th>VI</th>
<th>VII</th>
<th>VIII</th>
<th>TOTAL SCORE</th>
<th>TOTAL MOTIFS</th>
<th>PARAMETERS</th>
</tr>
</thead>
<tbody>
<tr>
<td>BLOCKMAKER</td>
<td>12</td>
<td>5</td>
<td>0</td>
<td>12</td>
<td>10</td>
<td>10</td>
<td>12</td>
<td>0</td>
<td>61</td>
<td>3</td>
<td>run type=3; sign=(3,5); dist=20</td>
</tr>
<tr>
<td>ITERALIGN</td>
<td>12</td>
<td>10</td>
<td>12</td>
<td>12</td>
<td>12</td>
<td>12</td>
<td>8+2</td>
<td>9</td>
<td>90</td>
<td>12</td>
<td>w=3; PAM 250 matrix</td>
</tr>
<tr>
<td>MATCHBOX</td>
<td>12</td>
<td>4</td>
<td>10</td>
<td>12</td>
<td>12</td>
<td>12</td>
<td>12</td>
<td>12</td>
<td>82</td>
<td>9</td>
<td>version 1.3; PAM 250 matrix</td>
</tr>
<tr>
<td>MEME</td>
<td>12</td>
<td>11</td>
<td>10</td>
<td>12</td>
<td>12</td>
<td>12</td>
<td>8</td>
<td>92</td>
<td>9</td>
<td>10</td>
<td>mod oops; nmotifs=12; maxiter=100</td>
</tr>
<tr>
<td>PIMA</td>
<td>12</td>
<td>12</td>
<td>12</td>
<td>12</td>
<td>12</td>
<td>12</td>
<td>12</td>
<td>95</td>
<td>26</td>
<td>26</td>
<td>ML; patgen matrix</td>
</tr>
<tr>
<td>PROBE</td>
<td>12</td>
<td>11</td>
<td>9</td>
<td>12</td>
<td>12</td>
<td>12</td>
<td>12</td>
<td>12</td>
<td>91</td>
<td>11</td>
<td>S=300</td>
</tr>
</tbody>
</table>

See footnotes “a” through “d” for table 2. Abbreviations in this column are the same as in table 2, except as follows: maxiter = maximum iterations and ML = maximal linkage clustering method.

Table 2.4: Program Results and Parameters for PRO12 Data Set.

<table>
<thead>
<tr>
<th>PROGRAM</th>
<th>I</th>
<th>II</th>
<th>III</th>
<th>TOTAL SCOREb</th>
<th>TOTAL MOTIFSc</th>
<th>PARAMETERSd</th>
</tr>
</thead>
<tbody>
<tr>
<td>BLOCKMAKER</td>
<td>11</td>
<td>2</td>
<td>6</td>
<td>19</td>
<td>2</td>
<td>run type=(1,3,4); sign=(3,5); dist=(3,5,10)</td>
</tr>
<tr>
<td>ITERALIGN</td>
<td>8+4</td>
<td>0</td>
<td>0</td>
<td>8</td>
<td>6</td>
<td>ht=0.8; PAM 250 matrix</td>
</tr>
<tr>
<td>MATCHBOX</td>
<td>12</td>
<td>2+2</td>
<td>8</td>
<td>22</td>
<td>4</td>
<td>version 1.3; blosum 62 matrix</td>
</tr>
<tr>
<td>MEME</td>
<td>12</td>
<td>2</td>
<td>10</td>
<td>24</td>
<td>9</td>
<td>mod oops; nmotifs=10; adj=none</td>
</tr>
<tr>
<td>PIMA</td>
<td>12</td>
<td>0</td>
<td>8</td>
<td>20</td>
<td>7</td>
<td>ML; patgen matrix</td>
</tr>
<tr>
<td>PROBE</td>
<td>12</td>
<td>6+5</td>
<td>11</td>
<td>29</td>
<td>3</td>
<td>S=600</td>
</tr>
</tbody>
</table>

See footnotes “a” through “d” for table 2. Abbreviations in this column are the same as in table 2, except as follows: run type = 1 is non-iterative mode and 4 is shuffled, iterative mode; ht = high threshold value; adj = type of LRT adjustment; ML = maximal linkage clustering method.
Table 2.5: Program Results and Parameters for RH12 Data Set.

<table>
<thead>
<tr>
<th>PROGRAM</th>
<th>II</th>
<th>III</th>
<th>IV</th>
<th>TOTAL SCORE</th>
<th>TOTAL MOTIFS</th>
<th>PARAMETERS[^d]</th>
</tr>
</thead>
<tbody>
<tr>
<td>BLOCKMAKER</td>
<td>8</td>
<td>2</td>
<td>5</td>
<td>0</td>
<td>15</td>
<td>2, run type=(3,4); dist=(3,5,10) or run type=1; sign=(3,5); dist=(3,10)</td>
</tr>
<tr>
<td>ITERALIGN</td>
<td>0</td>
<td>6</td>
<td>5</td>
<td>0</td>
<td>11</td>
<td>4, ht=0.5; PAM 150 matrix</td>
</tr>
<tr>
<td>MATCHBOX</td>
<td>10</td>
<td>8</td>
<td>0</td>
<td>0</td>
<td>18</td>
<td>2, version 1.3; blosum 62 matrix</td>
</tr>
<tr>
<td>MEME</td>
<td>9</td>
<td>9</td>
<td>8</td>
<td>9</td>
<td>35</td>
<td>5, mod oops; nmotifs=10; width=10</td>
</tr>
<tr>
<td>PIMA</td>
<td>10</td>
<td>9</td>
<td>12</td>
<td>11</td>
<td>42</td>
<td>19, SB; ptagen matrix; d=2.0; i=22.0</td>
</tr>
<tr>
<td>PROBE</td>
<td>11</td>
<td>9</td>
<td>11</td>
<td>9</td>
<td>40</td>
<td>6, default</td>
</tr>
</tbody>
</table>

See footnotes “a” through “d” for table 2. ^d Abbreviations in this column are the same as in table 2, except as follows: ht = high threshold value; width = starting motif width; d = length dependent gap penalty; i = length independent gap penalty.

Table 2.6: Program Results and Parameters for RT20 Data Set

<table>
<thead>
<tr>
<th>PROGRAM</th>
<th>II</th>
<th>III</th>
<th>IV</th>
<th>V</th>
<th>VI</th>
<th>TOTAL SCORE</th>
<th>TOTAL MOTIFS</th>
<th>PARAMETERS[^d]</th>
</tr>
</thead>
<tbody>
<tr>
<td>BLOCKMAKER</td>
<td>0</td>
<td>18</td>
<td>0</td>
<td>19</td>
<td>0</td>
<td>37</td>
<td>2, run type=1; sign=5; dist=5 (5-30)</td>
<td></td>
</tr>
<tr>
<td>ITERALIGN</td>
<td>7</td>
<td>11</td>
<td>8</td>
<td>15</td>
<td>8</td>
<td>59</td>
<td>24, ht=0.8; PAM 250 matrix</td>
<td></td>
</tr>
<tr>
<td>MATCHBOX</td>
<td>0</td>
<td>16</td>
<td>11</td>
<td>20</td>
<td>14+2</td>
<td>20</td>
<td>81</td>
<td>7, version 1.3; blosum 62 matrix</td>
</tr>
<tr>
<td>MEME</td>
<td>19</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>14</td>
<td>19</td>
<td>112</td>
<td>8, mod oops; nmotifs=10; distance=0.1</td>
</tr>
<tr>
<td>PIMA</td>
<td>18</td>
<td>20</td>
<td>8+12</td>
<td>20</td>
<td>0</td>
<td>15</td>
<td>85</td>
<td>11, ML; class 2 matrix</td>
</tr>
<tr>
<td>PROBE</td>
<td>18</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>15</td>
<td>20</td>
<td>113</td>
<td>8, S=500</td>
</tr>
</tbody>
</table>

See footnotes “a” through “d” for table 2. ^d Abbreviations in this column are the same as in table 2, except as follows: distance = convergence criterion; ML = maximal linkage clustering method.
For the most conserved data set, GLOB12, MATCHBOX was the only program that was unable to detect all five motifs of the OSM to some degree. BLOCKMAKER and PROBE reported at least one motif as several mis-aligned subsets (as indicated by more than one score per motif in Table 2.2). ITERALIGN and PIMA produced the highest scores by correctly identifying four of the motifs in all 12 sequences and a fifth motif in 11 sequences. Although ITERALIGN produced one of the highest scores, it was not necessarily the most informative because it did not distinguish motifs from MIRs. As a result, so many false positives were reported that the entire sequence was reported as being part of a motif. Program rank based on highest to lowest total motif-detection score is as follows: PIMA and ITERALIGN, PROBE, MEME, BLOCKMAKER, and MATCHBOX.

For the KIN12 data set, BLOCKMAKER was the only program that could not locate all eight motifs of the OSM. All of the other programs identified the five most conserved motifs (I, IV, V, VI, VII) in all 12 sequences. ITERALIGN located motif VIII as two unaligned subsets. PIMA received the highest score by detecting the first seven motifs in all 12 sequences and motif VIII in 11 of the sequences. Total score ranking for the KIN12 data set is PIMA, MEME, PROBE, ITERALIGN, MATCHBOX, and BLOCKMAKER.

For the more divergent PRO12 data set, only ITERALIGN and PIMA were unable to locate all three motifs of the OSM. MATCHBOX, MEME, PIMA, and PROBE were able to detect the most conserved motif I in all 12 sequences. ITERALIGN, MATCHBOX, and PROBE reported two unaligned subsets for one motif. Although PROBE produced two unaligned subsets for motif II, it still received the highest overall score since one of the subsets contained 6 sequences (the highest number detected for motif II by any program). Total score ranking is PROBE, MEME, MATCHBOX, PIMA, BLOCKMAKER, and ITERALIGN.

For the RH12 data set, BLOCKMAKER, ITERALIGN, and MATCHBOX were not able to detect all four motifs of the OSM. ITERALIGN and MATCHBOX could only locate two of the four motifs. MEME, PIMA, and PROBE were able to locate all of the motifs to some degree. PIMA was the only program to locate a single motif (III) in all 12 sequences. PIMA scored the highest, followed by PROBE, MEME, MATCHBOX, BLOCKMAKER, and ITERALIGN.
For the RT20 data set, BLOCKMAKER, MATCHBOX, and PIMA were not able to locate all six motifs of the OSM. ITERALIGN, MEME, and PROBE were able to detect the entire OSM to some degree. MEME detected three of the motifs (II, III, and IV) in all 20 sequences and motif VI in 19 sequences, while PROBE was able to detect the four most conserved motifs (II, III, IV, and VI) in all 20 sequences. Total score ranking is PROBE, MEME, PIMA, MATCHBOX, ITERALIGN, and BLOCKMAKER.

Based on the total score ranking from all of the small data sets, the cumulative ranking of the programs is as follows: PROBE (329), MEME (317), PIMA (301), ITERALIGN (227), MATCHBOX (226), and BLOCKMAKER (180).

The BLOCKMAKER program had a tendency to report a small number of large amino acid “blocks”, some of which contained multiple motifs. Although the motifs were aligned and considered correctly identified, the BLOCKMAKER program was not able to consistently separate the motifs from the MIRs for all data sets except RT20. ITERALIGN also had a problem with separating motifs from MIRs for the more conserved data sets. Rather than reporting a small number of large blocks containing multiple motifs, ITERALIGN reported a large number of false positives that were small overlapping or connected motifs. In both instances, the entire sequence, or a large part of it, was erroneously reported as being part of a motif.

MATCHBOX is only available via a webserver and only allows for a single parameter change (amino acid substitution matrix). The low program results for MATCHBOX could not be improved since parameter changes are limited to a single parameter.

A summary of results for the small data sets is given in Table 2.12. Values represent the percentage of scores reported as percentage of sequences in which motifs were correctly identified (total score received out of the total score possible).

<table>
<thead>
<tr>
<th>PROGRAM</th>
<th>DATA SETS (in order of decreasing percent identity)</th>
<th>AVG</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>GLOB12</td>
<td>KIN12</td>
</tr>
<tr>
<td>BLOCKMAKER</td>
<td>80</td>
<td>63</td>
</tr>
<tr>
<td>ITERALIGN</td>
<td>98</td>
<td>94</td>
</tr>
<tr>
<td>MATCHBOX</td>
<td>38</td>
<td>85</td>
</tr>
<tr>
<td>MEME</td>
<td>90</td>
<td>96</td>
</tr>
<tr>
<td>PIMA</td>
<td>98</td>
<td>99</td>
</tr>
<tr>
<td>PROBE</td>
<td>93</td>
<td>95</td>
</tr>
</tbody>
</table>

Scores reported as percentage of sequences in which motifs were correctly identified (total score received out of the total score possible)

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
sequences in which the motifs were correctly identified. This percentage was calculated by the TOTAL SCORE (Tables 2.2-2.6) received out of the total score possible.

3.2 Large data set analyses

Program results on the large data sets are given in Tables 2.8-2.12, along with the parameters used to produce these results.

For the GLOB 174 data set, PIMA was only able to detect three motifs (I, II, and III) of the OSM. For GLOB 174, as well as every other large data set, PIMA separated the input sequences into two groups, based on sequence similarity, and reported motifs for each of these subgroups. The GLOB 174 data set was divided into one group of 137 sequences and another group containing the other 37 sequences. For the larger subgroup, only motif I was located in all 137 sequences. For the smaller subgroup, only motifs I and II were correctly identified in all 37 sequences. MEME and PROBE were able to detect all five motifs of the OSM, but not in all 174 input sequences. For all data sets, MEME reported a variable number of sequences for each motif. MEME's total, or cumulative, score for the GLOB 174 data set was 736 out of the possible 870 sequences. PROBE reported 173 sequences per motif and received a total score of 852. The PROBE results were taken from the .scn output file instead of the .mfe output file. The .scn (database scan) file was used because it includes over-represented sequences while the .mfe (optimum alignment) file only contains an evenly distributed, representative group of the input sequences (see Discussion for detailed explanation).

For the KIN 186 data set, every program was able to locate the eight motifs of the OSM in some of the sequences. The large subgroup reported by PIMA contained 100 sequences in which every motif, except motif III, was identified to some degree. Motifs I, IV, and V were detected in all 100 sequences. The smaller subgroup reported by PIMA contained 86 sequences in which all motifs, except V and VIII, were accurately identified. For MEME, the total score was 1438 out of a possible 1488. MEME was only able to accurately identify a single motif (IV) in all 186 sequences. For motifs V, VI, and VII, MEME correctly identified motifs in 184 out of 184, 177 out of 177, and 185 out of 185 of the sequences reported, respectively. PROBE reported all 186 sequences and received a the total score of 1454. PROBE was able to locate four of the motifs (IV, V, VI, and VII) in all 186 sequences.
Table 2.8: Program Results and Parameters for GLOB174 Data Set.

<table>
<thead>
<tr>
<th>PROGRAM</th>
<th>I*</th>
<th>II</th>
<th>III</th>
<th>IV</th>
<th>V</th>
<th>SEQS</th>
<th>TOTAL SCORE</th>
<th>TOTAL MOTIFS</th>
<th>PARAMETERS</th>
</tr>
</thead>
<tbody>
<tr>
<td>MEME</td>
<td>167/167</td>
<td>165/166</td>
<td>125/154</td>
<td>137/144</td>
<td>142/145</td>
<td>736</td>
<td>8</td>
<td>mod oops; nmotifs=10</td>
<td></td>
</tr>
<tr>
<td>% REPORTED</td>
<td>100</td>
<td>99</td>
<td>81</td>
<td>95</td>
<td>98</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>% INPUT</td>
<td>96</td>
<td>95</td>
<td>72</td>
<td>79</td>
<td>82</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PIMA</td>
<td>137</td>
<td>106</td>
<td>131</td>
<td>0</td>
<td>0</td>
<td>137</td>
<td>374</td>
<td>4</td>
<td>ML1; patgen matrix</td>
</tr>
<tr>
<td>% REPORTED</td>
<td>100</td>
<td>77</td>
<td>96</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>% INPUT</td>
<td>79</td>
<td>61</td>
<td>75</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PROBE</td>
<td>173</td>
<td>173</td>
<td>172</td>
<td>170</td>
<td>164+5</td>
<td>173</td>
<td>852</td>
<td>6</td>
<td>default (.scn file)</td>
</tr>
<tr>
<td>% REPORTED</td>
<td>100</td>
<td>100</td>
<td>99</td>
<td>98</td>
<td>95+3</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>% INPUT</td>
<td>99</td>
<td>99</td>
<td>99</td>
<td>99</td>
<td>94+3</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

*Roman numerals represent each motif of the OSM and values in these columns indicate the number or percent of sequences in which the motif was correctly identified. Some methods find correct matches in more than one subset of the data without correct alignment of these subsets to one another, indicated by more than one result per motif. # REPORTED indicates the number of sequences correctly identified. For MEME, this row is the number of sequences correctly identified out of the total number reported. For PIMA and PROBE, the total number of sequences reported is indicated in the SEQS column. % REPORTED is the percentage of sequences that are correctly identified out the number of sequences reported. % INPUT is the percentage of sequences correctly identified out of the entire input set of sequences. TOTAL SCORE is the sum of all individual motif scores. In the case of multiple subsets per motif, the highest number per column was used for this summation. TOTAL SCORE is used to numerically rank the programs, with the highest score being the better score. The TOTAL MOTIFS column indicates the total number of motifs reported by each program. For MEME, this is the actual number of motifs the program reported before all of the motifs of the OSM were located. The total number of motifs reported by MEME was specified by the user (nmotif parameter). For PIMA, TOTAL MOTIFS the number of amino acid residues reported as being part of a motif. The PARAMETER column indicates the user-specified parameter changes from the default values that produced the best results. Abbreviations in this column are defined as follows: mod oops = motif distribution equals one occurrence per sequence; nmotifs = number of motifs to find; ML1 and ML2 = maximal linkage clustering method results.
Table 2.9: Program Results and Parameters for KIN186 Data Set.

<table>
<thead>
<tr>
<th>PROGRAM</th>
<th>I</th>
<th>II</th>
<th>III</th>
<th>IV</th>
<th>V</th>
<th>VI</th>
<th>VII</th>
<th>VIII</th>
<th>SEQS</th>
<th>TOTAL SCORE</th>
<th>TOTAL MOTIFS</th>
<th>PARAMETERS</th>
</tr>
</thead>
<tbody>
<tr>
<td>MEME</td>
<td>182/182</td>
<td>180/180</td>
<td>164/166</td>
<td>186/186</td>
<td>184/184</td>
<td>177/177</td>
<td>185/185</td>
<td>180/180</td>
<td>&lt;--</td>
<td>1438</td>
<td>12</td>
<td>mod oops; nmotifs=12</td>
</tr>
<tr>
<td>% REPORTED</td>
<td>100</td>
<td>100</td>
<td>99</td>
<td>100</td>
<td>100</td>
<td>100</td>
<td>100</td>
<td>100</td>
<td>100</td>
<td>95</td>
<td>99</td>
<td>95+4</td>
</tr>
<tr>
<td>% INPUT</td>
<td>98</td>
<td>97</td>
<td>88</td>
<td>100</td>
<td>99</td>
<td>95</td>
<td>99</td>
<td>97</td>
<td>100</td>
<td>692</td>
<td>13</td>
<td>ML1; patgen matrix</td>
</tr>
<tr>
<td>PIMA</td>
<td>100</td>
<td>99</td>
<td>0</td>
<td>100</td>
<td>100</td>
<td>99</td>
<td>99</td>
<td>95+4</td>
<td>100</td>
<td>515</td>
<td>12</td>
<td>ML2; patgen matrix</td>
</tr>
<tr>
<td>% REPORTED</td>
<td>100</td>
<td>99</td>
<td>0</td>
<td>100</td>
<td>100</td>
<td>99</td>
<td>99</td>
<td>95+4</td>
<td>100</td>
<td>515</td>
<td>12</td>
<td>ML2; patgen matrix</td>
</tr>
<tr>
<td>% INPUT</td>
<td>54</td>
<td>53</td>
<td>0</td>
<td>54</td>
<td>54</td>
<td>53</td>
<td>53</td>
<td>51+2</td>
<td>86</td>
<td>46</td>
<td>46</td>
<td></td>
</tr>
<tr>
<td>PROBE</td>
<td>184</td>
<td>178</td>
<td>174</td>
<td>186</td>
<td>186</td>
<td>186</td>
<td>186</td>
<td>186</td>
<td>186</td>
<td>1454</td>
<td>12</td>
<td>default (.scn file)</td>
</tr>
<tr>
<td>% REPORTED</td>
<td>99</td>
<td>96</td>
<td>94</td>
<td>100</td>
<td>100</td>
<td>100</td>
<td>100</td>
<td>96</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>% INPUT</td>
<td>99</td>
<td>96</td>
<td>94</td>
<td>100</td>
<td>100</td>
<td>100</td>
<td>100</td>
<td>96</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

See footnotes "a" through "h" for table 7.

Table 2.10: Program Results and Parameters for PRO114 Data Set.

<table>
<thead>
<tr>
<th>PROGRAM</th>
<th>I</th>
<th>II</th>
<th>III</th>
<th>SEQS</th>
<th>TOTAL SCORE</th>
<th>TOTAL MOTIFS</th>
<th>PARAMETERS</th>
</tr>
</thead>
<tbody>
<tr>
<td>MEME</td>
<td>112/112</td>
<td>86/95</td>
<td>101/106</td>
<td>&lt;--</td>
<td>299</td>
<td>3</td>
<td>mod oops; nmotifs=10</td>
</tr>
<tr>
<td>% REPORTED</td>
<td>100</td>
<td>90</td>
<td>95</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>% INPUT</td>
<td>98</td>
<td>75</td>
<td>89</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PIMA</td>
<td>79</td>
<td>79</td>
<td>79</td>
<td>237</td>
<td>9</td>
<td>ML1; patgen matrix</td>
<td></td>
</tr>
<tr>
<td>% REPORTED</td>
<td>100</td>
<td>100</td>
<td>100</td>
<td>24</td>
<td>3</td>
<td>ML2; patgen matrix</td>
<td></td>
</tr>
<tr>
<td>% INPUT</td>
<td>69</td>
<td>69</td>
<td>69</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PROBE</td>
<td>111</td>
<td>98+2+2</td>
<td>102+2</td>
<td>111</td>
<td>3</td>
<td>default (.scn file)</td>
<td></td>
</tr>
<tr>
<td>% REPORTED</td>
<td>100</td>
<td>88</td>
<td>92</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>% INPUT</td>
<td>97</td>
<td>86</td>
<td>89</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

See footnotes "a" through "h" for table 7.
Table 2.11: Program Results and Parameters for RH169 Data Set.

<table>
<thead>
<tr>
<th>PROGRAM</th>
<th>I</th>
<th>II</th>
<th>III</th>
<th>IV</th>
<th>SEQS&lt;sup&gt;c&lt;/sup&gt;</th>
<th>TOTAL SCORE&lt;sup&gt;d&lt;/sup&gt;</th>
<th>TOTAL MOTIFS&lt;sup&gt;e&lt;/sup&gt;</th>
<th>PARAMETERS&lt;sup&gt;f&lt;/sup&gt;</th>
</tr>
</thead>
<tbody>
<tr>
<td>MEME</td>
<td># REPORTED&lt;sup&gt;b&lt;/sup&gt;</td>
<td>136/145</td>
<td>157/159</td>
<td>116/162</td>
<td>103/133</td>
<td>&lt;=</td>
<td>512</td>
<td>7</td>
</tr>
<tr>
<td></td>
<td>% REPORTED&lt;sup&gt;d&lt;/sup&gt;</td>
<td>94</td>
<td>99</td>
<td>72</td>
<td>77</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>% INPUT&lt;sup&gt;e&lt;/sup&gt;</td>
<td>80</td>
<td>93</td>
<td>69</td>
<td>61</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PIMA</td>
<td># REPORTED&lt;sup&gt;b&lt;/sup&gt;</td>
<td>73</td>
<td>75</td>
<td>0</td>
<td>0</td>
<td>96</td>
<td>148</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>% REPORTED&lt;sup&gt;d&lt;/sup&gt;</td>
<td>76</td>
<td>78</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>% INPUT&lt;sup&gt;e&lt;/sup&gt;</td>
<td>43</td>
<td>44</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td># REPORTED&lt;sup&gt;b&lt;/sup&gt;</td>
<td>100</td>
<td>100</td>
<td>100</td>
<td>100</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>% REPORTED&lt;sup&gt;d&lt;/sup&gt;</td>
<td>43</td>
<td>43</td>
<td>43</td>
<td>43</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PROBE</td>
<td># REPORTED&lt;sup&gt;b&lt;/sup&gt;</td>
<td>159</td>
<td>160</td>
<td>160</td>
<td>151</td>
<td>160</td>
<td>630</td>
<td>6</td>
</tr>
<tr>
<td></td>
<td>% REPORTED&lt;sup&gt;d&lt;/sup&gt;</td>
<td>99</td>
<td>100</td>
<td>100</td>
<td>94</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>% INPUT&lt;sup&gt;e&lt;/sup&gt;</td>
<td>94</td>
<td>95</td>
<td>95</td>
<td>89</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

See footnotes “a” through “h” for table 7. Abbreviations in this column are the same as table 7, except for the following: w = starting motif width; d = length dependent gap penalty and i = length independent gap penalty.

Table 2.12: Program Results and Parameters for RT178 Data Set.

<table>
<thead>
<tr>
<th>PROGRAM</th>
<th>I</th>
<th>II</th>
<th>III</th>
<th>IV</th>
<th>V</th>
<th>VI</th>
<th>SEQS&lt;sup&gt;c&lt;/sup&gt;</th>
<th>TOTAL SCORE&lt;sup&gt;d&lt;/sup&gt;</th>
<th>TOTAL MOTIFS&lt;sup&gt;e&lt;/sup&gt;</th>
<th>PARAMETERS&lt;sup&gt;f&lt;/sup&gt;</th>
</tr>
</thead>
<tbody>
<tr>
<td>MEME</td>
<td># REPORTED&lt;sup&gt;b&lt;/sup&gt;</td>
<td>114/151</td>
<td>175/175</td>
<td>149/172</td>
<td>177/178</td>
<td>127/155</td>
<td>156/157</td>
<td>&lt;=</td>
<td>898</td>
<td>10</td>
</tr>
<tr>
<td></td>
<td>% REPORTED&lt;sup&gt;d&lt;/sup&gt;</td>
<td>75</td>
<td>100</td>
<td>87</td>
<td>99</td>
<td>82</td>
<td>99</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>% INPUT&lt;sup&gt;e&lt;/sup&gt;</td>
<td>64</td>
<td>98</td>
<td>84</td>
<td>99</td>
<td>71</td>
<td>88</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PIMA</td>
<td># REPORTED&lt;sup&gt;b&lt;/sup&gt;</td>
<td>35</td>
<td>28</td>
<td>29</td>
<td>53</td>
<td>0</td>
<td>25</td>
<td>93</td>
<td>170</td>
<td>12</td>
</tr>
<tr>
<td></td>
<td>% REPORTED&lt;sup&gt;d&lt;/sup&gt;</td>
<td>38</td>
<td>30</td>
<td>31</td>
<td>57</td>
<td>0</td>
<td>27</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>% INPUT&lt;sup&gt;e&lt;/sup&gt;</td>
<td>20</td>
<td>16</td>
<td>16</td>
<td>30</td>
<td>0</td>
<td>15</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td># REPORTED&lt;sup&gt;b&lt;/sup&gt;</td>
<td>76</td>
<td>85</td>
<td>85</td>
<td>85</td>
<td>85</td>
<td>85</td>
<td>85</td>
<td>501</td>
<td>29</td>
</tr>
<tr>
<td></td>
<td>% REPORTED&lt;sup&gt;d&lt;/sup&gt;</td>
<td>89</td>
<td>100</td>
<td>100</td>
<td>100</td>
<td>100</td>
<td>100</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>% INPUT&lt;sup&gt;e&lt;/sup&gt;</td>
<td>43</td>
<td>48</td>
<td>48</td>
<td>48</td>
<td>48</td>
<td>48</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PROBE</td>
<td># REPORTED&lt;sup&gt;b&lt;/sup&gt;</td>
<td>130</td>
<td>176</td>
<td>176</td>
<td>176</td>
<td>118</td>
<td>132</td>
<td>176</td>
<td>908</td>
<td>10</td>
</tr>
<tr>
<td></td>
<td>% REPORTED&lt;sup&gt;d&lt;/sup&gt;</td>
<td>74</td>
<td>100</td>
<td>100</td>
<td>100</td>
<td>67</td>
<td>75</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>% INPUT&lt;sup&gt;e&lt;/sup&gt;</td>
<td>73</td>
<td>99</td>
<td>99</td>
<td>99</td>
<td>66</td>
<td>74</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

See footnotes “a” through “h” for table 7.
For the PRO114 data set, all programs were able to distinguish the three motifs to some degree. The larger subgroup reported by PIMA contained 79 sequences in which all three motifs were correctly identified. The small subgroup reported by PIMA contained 35 sequences, with the highest detection being motif I in 32 of the sequences. The total score for MEME was 299 out of a possible 342. PROBE reported 111 sequences for each motif, with a total score of 311. PROBE was able to identify motif I in all 111 sequences reported, while MEME identified it in all 112 of the sequences that it reported for motif I. However, PROBE had a higher rate of detection for the other two motifs (II and III).

For the RH169 data set, all programs were able to locate all four motifs of the OSM to a certain extent. The large subgroup reported by PIMA contained 96 sequences. However, only motifs I and II were located in 73 and 75 sequences, respectively. The small subgroup for PIMA contained 73 sequences in which all four motifs were correctly identified. The total score for MEME was 512 out of a possible 845. PROBE reported 160 sequences for each motif, with a total score of 630. PROBE was only able to locate motifs II and III in all 160 sequences reported.

For the RT178 data set, all programs were able to detect the six motifs to some degree. The large subgroup reported by PIMA contained 93 sequences. Motif V was not detected at all and the other motifs were identified in a small number of sequences. For the small subgroup containing 85 sequences, PIMA was able to locate all the motifs in all 85 sequences, except motif I which was located in 76 sequences. MEME achieved a total score of 898 out of a possible 1068. MEME was able to locate motif II in the entire 175 sequences that it reported for that motif. PROBE reported 176 sequences per motif and obtained a total score of 908. PROBE located the most conserved motifs II, III, and IV in all 176 sequences reported.

For all programs, many of the results did not include all of the sequences from the input data set. While the lower number reported will certainly affect the number of sequences in which the motif was correctly identified, it does not imply anything about how accurately a method locates motifs among the sequences that it does report. For instance, although PROBE reported 186 sequences for motif VIII of the KIN186 data set, it only correctly located the motif in 174 of these sequences. MEME, on the other hand, only reported 180 sequences for the same motif III, but it accurately identified the motif in all 180
sequences. In this case, MEME has a higher percentage of accuracy (see % REPORTED, Table 2.9).

Overall, PROBE had the highest rate of detection when considering results for all of the large data sets.

A summary of results for the large data sets is given in Table 2.13. Values represent the percentage of sequences in which the motifs were correctly identified. This percentage was calculated by the TOTAL SCORE (Tables 2.2-2.6) received out of the total score possible.

Table 2.13: Summary of Large Data Set Analyses

<table>
<thead>
<tr>
<th>PROGRAM</th>
<th>DATA SETS (in order of decreasing percent identity)</th>
<th>AVG</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>GLOB174</td>
<td>KIN186</td>
</tr>
<tr>
<td>PIMA</td>
<td>43</td>
<td>46</td>
</tr>
<tr>
<td>MEME</td>
<td>12</td>
<td>35</td>
</tr>
<tr>
<td>PROBE</td>
<td>85</td>
<td>97</td>
</tr>
</tbody>
</table>

Scores reported as percentage of sequences in which motifs were correctly identified (total score received out of the total score possible)

4 Discussion

4.1 Small data set analyses

From the small data set analyses, it is apparent that local alignment programs are sensitive to the degree of sequence similarity among the data analyzed. A program may be robust for the analysis of similar sequences (GLOB12), but inadequate when the percentage of sequence identity is low (RH12). Methods that are able to identify motifs among both types of sequence data are more reliable than those methods that cannot. Of the methods tested, BLOCKMAKER and ITERALIGN were affected the most by the low sequence similarity among the data sets. Motif detection scores dropped considerably as the sequence similarity decreased among the data sets.

Motif detection results for BLOCKMAKER and ITERALIGN were also affected by high sequence similarity among the data. These methods report a high number of false positives when sequences have a high degree of similarity, resulting in the entire sequence, or a large portion of it, being reported part of a motif. These methods are unable to tease apart the motifs from the MIRs when the sequences share a high degree of similarity.
It should be noted that most local alignment programs produce false positive motifs. Although they are not part of the OSM, some of these motifs may actually represent subfamily motifs that are not common to the entire data set. On the other hand, local alignment programs may report a motif of the OSM as multiple unaligned subsets of subfamily motifs. This indicates that the computer program is locating the correct motif among subfamilies within the data, but cannot align the subfamily motifs to each other as part of an OSM. On at least one occasion, every program, except MEME, located a single motif in multiple unaligned subsets.

MATCHBOX had a low overall score and the results did not demonstrate a detectable correlation between performance and sequence similarity. Scores could not be improved through parameterization because MATCHBOX is only available for use through a webserver and the only parameter change allowed is the type of amino acid substitution matrix used.

Although PIMA had the third highest overall score, it required the most user-manipulation for analyzing and scoring the results. Several changes had to be made to the output file so it could be printed in a useful form for visual inspection. The results for PIMA are reported as individual amino acid residues. Often, an individual motif was identified by the most conserved residue(s) within the motif, rather than all residues of the motif.

MEME had second highest score for motif detection. One aspect of MEME worth mentioning is that the user must specify the number of motifs to locate. For the purpose of this study, it is not a problem to specify the number of motifs to look for since I know the number of motifs within the test data sets. However, most program users will not have this previous knowledge and will be required to guess the number of motifs to search for. By setting this number higher than the actual number of motifs, MEME will eventually report all of the motifs of the OSM. Unfortunately, the program will also report motifs that are not common to all of the sequences, either because they are subfamily motifs or they are simply incorrect, thus requiring the user to carefully evaluate each reported motif. One positive aspect of being able to specify the number of motifs to look for is that the user can force the program to continue to search for motifs until the results indicate that no more motifs are present. For example, the user may tell the program to look for 50 motifs, but the colinearity and scores for the motifs may decrease at the 10th motif.
reported. This allows the program to exhaustively search the data for all motifs, including subfamily motifs.

PROBE had the highest overall score for motif detection. Not only is PROBE the most accurate method, the results were easy to analyze since it reports the motifs in a co-linear order, just as they are found in an OSM.

The performance of local alignment programs is not only affected by the degree of sequence similarity among the data, but also by the type of motifs comprising the OSM. Some methods are not able to locate highly diverged or single residue motifs. This study does not go into great detail about the different types of motifs within each data set or each program’s ability to locate these motifs. The total score used in this study actually masks how well a program is able to locate individual motifs. However, total score is useful for making a general comparison of program performance. From the overall total score, it is apparent that BLOCKMAKER, ITERALIGN, and MATCHBOX do not perform as well as MEME. PIMA and PROBE at locating the OSMs within the test data sets.

4.2 Large data set analyses

From the large data set analyses, it becomes obvious that some methods are better than others in dealing with biased data. It is important to test a program’s ability to deal with over-representation in the data since databases are not made up of an equally distributed group of sequences. A large, biased group of sequences is a typical result of an average database search.

The results produced by PIMA are less informative since they contain subgroups that are not aligned to each other. The reason for this subgroup separation is that PIMA recognizes the over-representation of, or bias towards, a group of sequences that have higher sequence similarity within the group. These similar sequences are put into a separate subgroup and the remaining sequences comprise the other subgroup. Motifs are then located separately for each group, but the groups are never aligned to each other. Because PIMA does not provide correction for over-representation of sequence similarity, it is not able to locate the OSM among the entire group of sequences in the large data sets. Also, by putting similar sequences in a single group, there are no outliers to help tease out the motifs from the MIRs, thus producing many false positives for the OSM of the larger subgroup (see MOTIFS column in Tables 7-11). Although considered
false positives for the purpose of this study, the high number of common residues reflects the high level of similarity among the sequences, as well as subfamily motifs that may be present.

MEME is fairly accurate at locating the OSM among a group of biased sequences. However, MEME does not report the same number of sequences for each motif of the OSM. Sequences are excluded on an individual motif basis when the motif is so divergent that MEME is unable to detect it. As a result, the set of sequences identified for one motif may not be the same set of sequences for another motif. In the end, the actual number of sequences in which the entire OSM is identified may be less than the number of sequences reported for any given motif. The task of determining which group of sequences an entire OSM has been identified in makes MEME more difficult to use when trying to determine the common OSM for an entire group of sequences. In contrast, both PIMA and PROBE report a consistent group of sequences for each motif of the OSM.

The PROBE program had the highest rate of detection for the large data sets since it is designed to handle biased data. PROBE produces several output files. Of these files, the .mtf and .scn both contain motifs that were located by the program. The .mtf file reports motifs for an equally distributed group of sequences selected from the input data set. For the small data set analysis, the .mtf file was used since all of the input sequences were included in the results. Consequently, when the input data set is biased, the .mtf file includes only one representative sequence from the over-represented data and the results do not include all of the input sequences. The .scn file, however, contains all of the sequences in which motifs could be identified, including over-represented sequences. This is because motifs within the .scn file are based on an unbiased method for determining motifs. PROBE is able to eliminate bias within the .scn file by initially locating the motifs among a representative set and then adding back the over-represented data according to which sequence it is most similar to, or its representative sequence. The only sequences excluded from the .scn file are the ones that PROBE could not locate the OSM within. Therefore, the .scn file was used to evaluate program performance on the large data sets. Of all programs, PROBE was best able to locate the OSMs among the large, biased data sets.
4.3 General discussion

This study assesses program accuracy by assigning a motif-detection score. The reliability is measured by evaluating the consistency of each method. When considering the parameterization required by the user, only PROBE and MEME reliably detected motifs with constant parameters. PROBE consistently produced the most accurate results with the same parameters, either default parameters or setting the S (purge) value high enough to include all of the sequences. Although MEME produced the highest results with a few inconsistent parameters, it was fairly reliable (only slightly lower scores) with only two parameter changes: mod oops and nmotifs. The other methods in this study did not display any continuity of performance with specific parameter changes.

While most methods are able to locate an OSM among an equally-distributed set of conserved sequences, only the most robust methods will be able to locate the OSM among a biased data set containing highly divergent sequences. The best programs are still not 100% accurate. PROBE was the most accurate at detecting motifs for both the small and large data sets used in this study. Because of its high performance across a range of data with a single consistent parameter change, PROBE is the most reliable local alignment method in this study.

Although this study demonstrates PROBE to be a good local alignment method, it is recommended that both PROBE and MEME be used for the detection of motifs among protein sequence data. It is my recommendation to run PROBE first. The number of motifs reported by PROBE can be used as a guide for specifying the number of motifs (nmotifs) that MEME should search for. However, nmotifs should be set to a number higher than the number of motifs reported by PROBE. The confidence in detected motifs is increased when supported by the results of two methods. It is also recommended that all detected motifs be carefully evaluated. Results that appear incorrect may actually contain subfamily motifs that can be useful for detecting subfamily relationships. This analysis should serve as a reference to help biologists choose a local alignment method. It should be noted that the performance of each program in this analysis may vary with data sets that were not used in this study. As more benchmark data sets become available and new computational methods rapidly develop, extensive comparative analyses will prove to be essential tools for biologists.
Acknowledgements

This work was supported by a grant to Marcella A. McClure, Ph.D., from the NIH, AI 28309. Thanks are given to John Kowalski for technical computer assistance and Marcie McClure for advice, discussion, and manuscript critique.

References

CHAPTER 3

DO RNA-DEPENDENT POLYMERASES
SHARE COMMON ANCESTRY?

Abstract

It has been suggested that all RNA-dependent polymerases (RDpols) share common ancestry. More recently, however, it was concluded that any similarity among the RDpols is not statistically significant enough to claim homology. This study uses local alignment methods in an attempt to locate a common ordered-series-of-motifs among two of the most divergent groups of RDpols. If all RDpols are homologous, then the most accurate local alignment methods should be able to detect a common OSM among these two dissimilar groups. The results of this Chapter do not provide evidence of a common OSM and, therefore, do not support common ancestry of all RDpols.

1 Introduction

RNA-dependent polymerases (RDpols) use an RNA template to produce nucleic acid, either RNA or DNA depending on the type of polymerase. RNA-dependent RNA polymerases (RDRP) are encoded in all RNA viruses, except the retroviruses. Retroviruses and retroid elements contain an RNA-dependent DNA polymerase (RDDP), also known as reverse transcriptase (RT).

The RDRPs used in this study are from unsegmented negative-strand RNA viruses (Order Mononegavirales), e.g., measles, vesicular stomatitis virus, and ebola. The RDRP of these viruses is a complex of three protein subunits and the RNA template that functions to transcribe and replicate the viral genome. In transcription, the RDRP synthesizes positive strand RNAs to be translated into proteins. In replication, the RDRP synthesizes the full-length positive strand compliment of the negative strand
genomic RNA. From the nascent positive strand, the RDRP also synthesizes the complimentary negative strand genomic RNA that will be packaged into new virions. Although the entire polymerase complex is required for transcription and replication, it is the large (L) protein subunit that possesses the majority of enzymatic polymerase activities.

The RDDPs used in this study are RT protein sequences from positive-strand RNA viruses (retroviruses), e.g., simian immunodeficiency virus and human t-cell leukemia virus, as well as several double-stranded DNA viruses (badna-, caulimo-, hepadnaviruses). In contrast to the L protein, which produces RNA from the RNA template, the RT protein synthesizes DNA from the RNA template. In retroviruses, the RT protein uses the positive-strand genomic RNA as a template to synthesize complimentary double-stranded DNA, which is integrated into the host genome. The integrated viral DNA is then transcribed into mRNAs and replicated into full-length genomic copies by RNA polymerase II. In the dsDNA viruses, the host cell polymerase produces the mRNA transcripts that are used for translation by the host-cell machinery and replication by the virally-encoded RT.

Although the L and RT proteins are analogous in function, there are several fundamental differences between them. The obvious difference is that the L protein synthesizes RNA, while the RT protein produces DNA. As mentioned above, the L protein only functions as a heterotrimer, while the RT protein does not require any accessory proteins to function. This suggests that structural and mechanistic differences may exist between the L and RT proteins. Another significant difference is that the L protein sequences are approximately 1900 amino acid residues longer than the RT protein sequences. The protein sequences also share a statistically insignificant level of sequence similarity (below 30% identical).

Despite the limited similarity between the L and RT protein sequences, four motifs have been suggested to comprise a common OSM for all RDpols [O. Poch et al, 1989]. These motifs were derived from manual alignment of database scans using RT motifs. The proposed OSM implies that the RDRPs and the RDDPs are homologous since, by definition, an OSM is the functional or structural remnant of a common ancestor and is considered a protein family signature. Whether or not the identified residues actually constitute an ancestral OSM is debatable. The proposed OSM relationship consists of only four motifs, each containing one invariant amino acid residue, and a number of semiconserved residues.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
Studies that have attempted to locate the OSM among the L protein sequences have detected many conserved motifs, each comprised of several conserved amino acid residues [O. Poch et al, 1990; D.S. Stec et al, 1991; T-J. Choi et al, 1992]. The definition of a motif is quite ambiguous among these studies. For example, the 1990 study detected six large blocks of high conservation containing a total of 31 motifs, of which four are considered highly conserved. The 1992 study, however, reported 30 conserved regions, but defined them as 12 conserved segments. As a result, it is still not clear which, if not all, conserved regions comprise the ancestral OSM. On the other hand, the OSM within the RT sequences has been well-characterized by primary sequence analysis and confirmed by crystallography [M.A. McClure, 1993; L.A. Kohlstaedt et al, 1992]. The RT OSM is comprised of six highly conserved motifs, with multiple residues per motif. The OSM proposed to be common among the L and RT proteins resides within a small stretch of about 70 amino acid residues. This is only a small portion of the 2200 and 300 residues that, respectively, comprise the L and RT proteins. Although the suggested OSM includes previously defined motifs for both the L and RT protein sequences, it does not include all of the highly conserved motifs for both data sets. Rather, the proposed L-RT OSM contains only a subset of motifs from the individual L and RT OSMs.

Based on the OSM proposed by Poch (1989), phylogenetic studies have been performed on several different groupings of RDpols [J. Bruenn, 1991; E.V. Koonin, 1991; E.V. Koonin and V.V. Dolja, 1993]. Subsequent analysis of these studies, and of the OSM proposed for all RDpols, concludes that there is not enough statistically-based evidence to support any of these relationships [P.M. de A. Zanotto et al, 1996]. Whether or not common ancestry is responsible for the limited sequence similarity detected between RDRP and RDDP proteins is still an open question.

Statistically and mathematically-based local alignment methods can be used in an attempt to address this ongoing evolutionary debate. This study uses the latest, and most reliable, methods to address the hypothesis that all RDpols are not related. The detection of an OSM among the two most diverged groups of RNA-dependent polymerases, the L and RT protein, would support common ancestry. If the most reliable local alignment methods are unable to locate a common OSM, the results would support the hypothesis that all RDpols are not homolgous.
2 Materials and Methods

All analyses were performed on a Sun SPARCstation Ultra 1 running SUN OS 5.6.

2.1 Biological data

The L protein sequences were obtained from the GenBank database. Initially, 75 sequences were retrieved. Using a program that calculates pairwise similarity scores from an alignment generated by the Needleman-Wunsch algorithm [S.B. Needleman and C.D. Wunsch, 1970], and CLUSTER, an in-house hierarchical clustering method, 16 sequences were selected for the study (L16). These test sequences represent an even distribution of the sequences that are available from the database. The pairwise sequence identity among the L16 data set ranges from 16-61%, with an average of 23%. Based on the conservative substitution of amino acids, the sequence similarity is also low. GenBank accession numbers for L16 are as follows: U23458, Z12132, M75730, U65312, Z66517, AF017149, X05399, AB000388, Z11575, M30204, M31046, K02378, X89213, AB011257, L32603, and U04608.

The RT test sequences were also obtained from GenBank. Using the same method as for the L data set, a group of 16 RT sequences (RT16) was selected to represent an even distribution of sequences from the retro-, badna-, caulimo-, and hepadnavirus families. The pairwise sequence identity among RT protein sequences ranges from 17-70%, with an average of 27%. The sequence similarity for the RT data is also low. GenBank accession numbers for RT16 are as follows: X54482, U94514, L36905, J02255, AJ002234, M10987, U26458, U29144, M12349, U59751, M90542, X57924, M37980, M37980, L04972, M22056, and D01065. When necessary, the nucleic acid in the GenBank file was translated and the protein sequence was obtained from the translation.

The data set containing L and RT sequences (LRT32) was a compilation of the L16 and RT16 data sets.

2.2 Local alignment programs

The local alignment programs used in this study are the latest, and most reliable methods available [J. Hudak and M.A. McClure, 1999; Chapter 2 of this thesis].

MEME (Multiple Expectation Maximization for Motif Elicitation), version 2.2., locates motifs by estimating the parameters for a model that maximizes the likelihood of the data [T.L. Bailey and C. Elkan, 1994]. First, the data is broken up into overlapping sequences of specified length. The MM (Mixture
Model) algorithm creates a finite mixture model of the new data set that consists of two components, the motifs and the motif-background probabilities. The EM (Expectation Maximization) algorithm estimates and maximizes the expected log likelihood value of the model parameters.

The PROBE, version 1.0, program implements the Smith-Waterman algorithm [T.F. Smith and M.S. Waterman, 1981] that performs transitive searches to find regions of sequence similarity [A.F. Neuwald et al., 1997]. The sequences collected from this search are purged to eliminate unequal representation of the data and then aligned co-linearly using the Gibbs sampling algorithm [C.E. Lawrence et al., 1993; A.F. Neuwald et al., 1995]. The Gibbs sampling algorithm starts at a random position for all of the sequences except one. The excluded sequence is aligned to the others. This process is reiterated until the information content score is maximized. After Gibbs sampling, a genetic algorithm is used to recombine a randomly selected alignment and choose the best alignment produced. This alignment is used to search for more sequences, which are included in another iteration starting with the Gibbs sampling step, until no more new sequences are found.

2.3 Methodology

PROBE was used at default parameter settings. MEME requires the user to specify the number of motifs to search for. This value was set to 20. Other parameters set for MEME specify to the program that it should search for one motif occurrence per sequence, and that the sequences are protein.

First, both programs were run on L16 and RT16 individual test sets to determine if an OSM could be produced and if the produced results were similar to the motifs described in the literature. Second, the programs were run on the combined set of L and RT sequences (LRT32) in an attempt to locate a common OSM.

3 Results

The results for OSM detection by MEME and PROBE are indicated in Table 3.1.

For the L16 data set, both methods were able to locate conserved motifs. MEME detected 20 motifs since that was the number specified to the program. Motifs are reported in the order of most conserved to least conserved. Four of the first six motifs reported contain the regions suggested to comprise a common
OSM for the L and RT sequences. At the 15th motif, the motif scores that are generated by MEME started to decline and the start sites for each motif were becoming incongruent among the sequences, indicating the decreasing conservation among the results reported after motif 15. The 16th motif reported by MEME was the only motif reported in the entire carboxy portion of the L protein. The motifs reported by MEME span a region of approximately 1500 amino acid residues. PROBE reported a total of 12 motifs for the L16 data set. Motifs are reported in the colinear order in which they are exist, rather than most conserved to least conserved. The four conserved motifs, believed to represent an OSM for the L and RT proteins, were reported as three large blocks of conserved regions, with the 2nd and 3rd motif residing in the 2nd reported block. PROBE did not detect any motifs in the entire carboxy portion of the L protein. The 12 motifs reported span a region of about 1000 amino acids, less than half of the entire L protein. In addition, about half of the 1000 amino acids were reported as being part of a motif. All motifs reported by both MEME and PROBE resemble previously described conserved regions [O. Poch et al, 1990], but not all of the previously defined motifs were located by MEME and PROBE.

For the RT 16 data, MEME was able to locate all six motifs of the OSM by the eighth motif reported. PROBE reported a total of 7 motifs, including the six that comprise the OSM.

For the LRT32 data set, MEME located 20 motifs as specified to the program. However, only the first 15 motifs contained sequences from both the L16 and RT16 data set. The first motif reported by MEME actually contained the regions of the L and RT that were reported by Poch as the 2nd conserved motif of the L-RT OSM. However, these regions were not correctly aligned to each other. The remaining motifs detected by MEME are not consistent with the OSM proposed by Poch and do not support any other common OSM among the L and RT proteins. The results produced by PROBE also do not support a common OSM. PROBE was only able to detect a single motif among the LRT32 data set. This motif contained all 16 L sequences and only two RT sequences. For the L sequences, the region reported as a motif is the same as the first motif reported by MEME, where a single Trp residue is the most conserved amino acid of the motif. The motifs reported for the two RT sequences, however, are not the same as those reported by MEME.
Table 3.1 Results for L16, RT16, and LRT32 OSM Detection

<table>
<thead>
<tr>
<th>PROGRAM</th>
<th>DATA SETS</th>
<th>PARAMETERS</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>L16</td>
<td>RT16</td>
</tr>
<tr>
<td>MEME</td>
<td>+</td>
<td>+</td>
</tr>
<tr>
<td>PROBE</td>
<td>+</td>
<td>+</td>
</tr>
</tbody>
</table>

A "+" symbol indicates the program was able to locate an OSM within the data set, while a "-" symbol indicates that the program did not find an OSM.

4 Discussion

The results of MEME and PROBE suggest that a significant OSM does not exist between the L and RT proteins. Both methods were able to identify the conserved motifs for each individual data set, but were not able to find an OSM common to both data sets. These results support the hypothesis that all RDpols are not homologous.

Could it be that the latest, and most reliable, local alignment methods do not work on these types of data sets? Although these methods are specifically designed to detect weak signals, it is possible that the methods are not able to detect similarities among highly divergent sequences with such a great disparity in length. Or, is it more likely that conserved regions, or an entire OSM, do not exist among the L and RT sequences?

Although the motifs suggested by Poch are not statistically significant enough, by several methods, to claim the presence of a common OSM, does it mean that these regions are completely insignificant to the study of protein evolution? Maybe the conserved regions suggested by Poch are present simply by chance, considering that there are only four residues conserved in a colinear order among proteins with lengths ranging from 300 to 2200 amino acids.

An entirely different point of view is that the similar regions suggested by Poch could be the result of convergence. Although convergence has not been demonstrated at the sequence level, a recent study suggests that mechanistic convergence may explain low-level similarities among sequences [T.A. Steitz, 1999]. Low-level amino acid similarities that are present, such as one conserved residue instead of four or five conserved residues per motif, may resemble the motifs of an OSM. However, these slight resemblances may not be significant enough to support the existence of a true ancestral OSM. Rather, the similarity may be the result of functional, or mechanistic, convergence. In proteins with analogous...
function, there are only so many ways that the common function can be accomplished. It is possible that the rapidly evolving RNA genomes of the RDpols have exhaustively searched all possibilities and simply came up with similar solutions. While a highly conserved OSM represents divergence from a common ancestor, small amounts of colinear sequence similarity may be the result of mechanistic convergence.

The results of this Chapter do not support the existence of a common OSM among the L and RT proteins, and consequently among all RDpols. The development of new and improved computational methods and the increase in available sequence data may eventually lead to an OSM that supports homology. On the other hand, as the concept of mechanistic convergence develops, RDpols may become known as the first demonstrated occurrence of sequence convergence. Until then, the evolutionary debate continues.

Acknowledgments

This work was supported by a grant to M.A.M. from the NIH, AI 28309. We thank John Kowalski for technical computer assistance, Marcella McClure for advice and manuscript critique, and Seanna Corro and King Jordan for their invaluable long-distance assistance.

References

CONCLUDING REMARKS

A large portion of this thesis was devoted to comparative analyses of computational tools, specifically local alignment programs. With the increasing number of methods available, it would be impossible for biologists to make advances in their research if they were required to evaluate every method before starting their analyses. In addition, many biologists do not have the expertise necessary to comparatively evaluate these types of programs. It is the responsibility of the bioinformatician to analyze the methods and provide the results to biologists.

The comparative analyses in this thesis illustrate several important points for biologists who are delving into the area of sequence analysis. The most general, and foremost, point is that all computational methods are not equal. It is essential to research all of the available programs, compare them to each other, and choose the most appropriate method. Otherwise, the results of biological analyses may be just as inaccurate as the program used. When evaluating computational tools for sequence analysis, there are several important concepts to be aware of. First, some programs are affected by the degree of similarity among the sequence data being analyzed. For instance, a program may be robust for the analysis of similar sequences, but inadequate for a highly divergent set of sequences. In addition, some computational tools are adversely affected by the bias in the data. Only the most reliable methods will be able to produce accurate results on both an equally-distributed set of sequences and a data set that is biased towards one type of sequence. One of the most important findings of this thesis is that MEME and PROBE are two of the most accurate and reliable local alignment methods available.

The second portion of this thesis discusses the application of local alignment methods to address issues of common ancestry. MEME and PROBE were used in an attempt to locate an ordered-series-of-motifs (OSM) shared among two groups of highly divergent RNA-dependent polymerases. The presence of an OSM would support homology since it is the ancestral remnant. However, both methods were unable to detect motifs common to the polymerases. An OSM for these sequences was proposed ten years ago, but...
was later determined to be statistically insignificant [O. Poch et al. 1989; P.M. de A. Zanotto, 1996]. One controversial explanation for the weakly conserved sequence similarity is mechanistic convergence, although convergence at the sequence level has never been demonstrated. This analysis provides insight as to one of the many biological applications of local alignment methods. The location of an OSM can be used to support or refute sequence homology. Individual motifs of a detected OSM can also provide a starting point for functional or structural mutagenesis experiments. Local alignment methods are an indispensable tool of biology. With the increasing number of programs becoming available, it is just a matter of determining which one to use. Through the use of bioinformatic tools, a vast amount of knowledge is accessible and awaiting discovery.

It is my recommendation that biologists use both MEME and PROBE for detecting conserved sequence motifs. First, PROBE should be used at default parameter settings. The number of motifs reported by PROBE can serve as a guide for specifying the number of motifs that MEME should search for. Although PROBE is the most accurate method, MEME can be used to further validate the motifs reported by PROBE. Motifs located by both methods are more likely to be authentic than motifs found by only one method. By using computer programs to locate the most probable motifs, wet-bench biologists are provided a reasonable starting point for mutagenesis experiments. Wet-bench experiments, such as mutagenesis and crystallography, must be used to determine whether or not the predicted motifs are important to the function or structure of the protein. The combination of local alignment programs and wet-bench experiments is extremely important since it can be used to determine protein function and structure. As the discipline of Bioinformatics develops, sequence data and computer programs will continue to rapidly increase in number. As a result, comparative analyses of computational tools will become essential to the field of bioinformatics.
BIBLIOGRAPHY


D.S. Stec et al., *Proceedings of the National Academy of Science USA* 91, 4362 (1994).


APPENDIX I

SMALL DATA SET OSMs

### Five Motifs of the GLOB12 OSM

<table>
<thead>
<tr>
<th>I</th>
<th>II</th>
<th>III</th>
<th>IV</th>
<th>V</th>
</tr>
</thead>
<tbody>
<tr>
<td>FPTKTYP</td>
<td>AQVEHGGKK</td>
<td>SALSDLMAH</td>
<td>HCLTVTLAA</td>
<td>VHAsLDMK</td>
</tr>
<tr>
<td>FPTTITYFS</td>
<td>AQIKEHGGKK</td>
<td>SALSDLMAH</td>
<td>HCLTVVLLAR</td>
<td>AHAAMDK</td>
</tr>
<tr>
<td>YPQITTYFP</td>
<td>AQIKEHGGKK</td>
<td>SKLSDLMAHQC</td>
<td>HCPLEVVAI</td>
<td>VHASLDMK</td>
</tr>
<tr>
<td>YPWTQRFFE</td>
<td>PKVKEHGGKK</td>
<td>ALTSELHCDF</td>
<td>NVLVVCLAH</td>
<td>VQAAYQK</td>
</tr>
<tr>
<td>YPWTQRFFE</td>
<td>PKVKEHGGKK</td>
<td>AKLSELHCDF</td>
<td>NVLVVCLAR</td>
<td>VQAAMQK</td>
</tr>
<tr>
<td>YPWTQRFFA</td>
<td>PMVRAHGGKQ</td>
<td>AQPLELHCDF</td>
<td>DILIVTLAA</td>
<td>CQAAMQK</td>
</tr>
<tr>
<td>HPETLEKPD</td>
<td>EDLKEHGAT</td>
<td>KPLAQSEHAT</td>
<td>ECIIQVLQS</td>
<td>AQGAAMK</td>
</tr>
<tr>
<td>HPEVRFLEFD</td>
<td>ADLKEHGAT</td>
<td>KPLAQSEHAT</td>
<td>EAIILVQLQ</td>
<td>AQGAAMK</td>
</tr>
<tr>
<td>YPDQCHKFS</td>
<td>GSFATHATR</td>
<td>SKLGDHDKA</td>
<td>TALVAYLQA</td>
<td>VAAAGNMK</td>
</tr>
<tr>
<td>APESKTVFS</td>
<td>PKLEAHAALV</td>
<td>KRLGSIELK</td>
<td>GALLSITIKE</td>
<td>MGQAAMTE</td>
</tr>
<tr>
<td>APGAKDLFS</td>
<td>PDQAAMHAGK</td>
<td>KSLGSHSVE</td>
<td>EAIILTIKE</td>
<td>LNTAWMT</td>
</tr>
<tr>
<td>HPESRFLP</td>
<td>EQPKALAMT</td>
<td>KKLAVKCHQ</td>
<td>QELLGAITE</td>
<td>ILDAWGK</td>
</tr>
</tbody>
</table>

### Eight Motifs of the KIN12 OSM

<table>
<thead>
<tr>
<th>I</th>
<th>II</th>
<th>III</th>
<th>IV</th>
<th>V</th>
<th>VI</th>
<th>VII</th>
<th>VIII</th>
</tr>
</thead>
<tbody>
<tr>
<td>LGTGSFGR</td>
<td>MKI</td>
<td>NEK</td>
<td>LIVRDLKPENL</td>
<td>-DFGL</td>
<td>LAGEV</td>
<td>WDWALGVI</td>
<td>KRM-</td>
</tr>
<tr>
<td>LGGGKFGA</td>
<td>AKV</td>
<td>LEM</td>
<td>VHELDLKPENL</td>
<td>-DFGL</td>
<td>LSPVQ</td>
<td>TDWSLGVI</td>
<td>ARM-</td>
</tr>
<tr>
<td>IGRGFSR</td>
<td>IVM</td>
<td>SEL</td>
<td>ITHHELKPENL</td>
<td>TDFGL</td>
<td>TAPEV</td>
<td>WDWALVGA</td>
<td>ARM-</td>
</tr>
<tr>
<td>VGEGTYGV</td>
<td>LKK</td>
<td>RHI</td>
<td>ILEHDLKPENL</td>
<td>-DFGL</td>
<td>RAVEV</td>
<td>VDTWSICG</td>
<td>NRI-</td>
</tr>
<tr>
<td>LGSGEGSE</td>
<td>VKK</td>
<td>QEF</td>
<td>YVEHDLKPENL</td>
<td>-DFGL</td>
<td>TAPEV</td>
<td>ADIPSLGTV</td>
<td>NRP-</td>
</tr>
<tr>
<td>LGGSGFTG</td>
<td>VKK</td>
<td>NEV</td>
<td>IREDMKSHMI</td>
<td>-DFGL</td>
<td>MAPEV</td>
<td>SDVYSTGLVL</td>
<td>EPR-</td>
</tr>
<tr>
<td>LGAGGFSG</td>
<td>INQ</td>
<td>AEL</td>
<td>IVEHDLKPENL</td>
<td>-DFGL</td>
<td>RAEVL</td>
<td>ADITYSPAAL</td>
<td>ERP-</td>
</tr>
<tr>
<td>LGGGCFGE</td>
<td>IKT</td>
<td>QEA</td>
<td>YVERDLRAANI</td>
<td>-DFGL</td>
<td>TAPEA</td>
<td>SDVVSFGILL</td>
<td>ERP-</td>
</tr>
<tr>
<td>IGRHGFGE</td>
<td>VKS</td>
<td>QEA</td>
<td>CHERDLARHC</td>
<td>-DFGL</td>
<td>TAPEA</td>
<td>SDVVSFGILL</td>
<td>QRP-</td>
</tr>
<tr>
<td>LGSGAFOG</td>
<td>VMK</td>
<td>SEL</td>
<td>CHERDLARHC</td>
<td>-DFGL</td>
<td>MARES</td>
<td>SDVVSFGILL</td>
<td>TRP-</td>
</tr>
<tr>
<td>LGSGAFGT</td>
<td>IKE</td>
<td>DKA</td>
<td>LVERDLARHC</td>
<td>-DFGL</td>
<td>MALES</td>
<td>SDVWSYGTTE</td>
<td>SRP-</td>
</tr>
<tr>
<td>LTPGSEGC</td>
<td>VKN</td>
<td>HKA</td>
<td>IHERDINKEN</td>
<td>-DFGA</td>
<td>NAPDV</td>
<td>VDWAGLVI</td>
<td>LRPN</td>
</tr>
</tbody>
</table>

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
### Three Motifs of the PRO12 OSM

<table>
<thead>
<tr>
<th>I</th>
<th>II</th>
<th>III</th>
</tr>
</thead>
<tbody>
<tr>
<td>LLDTGAG-</td>
<td>VLGAGGQTQ</td>
<td>QAIIGRD</td>
</tr>
<tr>
<td>LLDGSA-</td>
<td>IGHIGGGIP</td>
<td>GSILGRD</td>
</tr>
<tr>
<td>LLDTGAG-</td>
<td>IGGGIGFIK</td>
<td>VNIIGRN</td>
</tr>
<tr>
<td>LIDTGAG-</td>
<td>LRGIGQGNN</td>
<td>VNLNGRD</td>
</tr>
<tr>
<td>LVDTGAG-</td>
<td>-QGATGQKR</td>
<td>YPLLGRD</td>
</tr>
<tr>
<td>FVDTGGG-</td>
<td>KIADGSSIT</td>
<td>DFIXGNN</td>
</tr>
<tr>
<td>LIDTGST</td>
<td>-TNSQPLT-</td>
<td>DLLLGKR</td>
</tr>
<tr>
<td>LFDSGSP</td>
<td>GFVATSKAV</td>
<td>QLIGGLN</td>
</tr>
<tr>
<td>VLDGSAS</td>
<td>VAKQGFIFY</td>
<td>KEAGGLN</td>
</tr>
<tr>
<td>VFDTGS</td>
<td>TYGTSMTG</td>
<td>DGILGLA</td>
</tr>
<tr>
<td>IFDTS</td>
<td>AYGTGSMGG</td>
<td>DGILGLA</td>
</tr>
<tr>
<td>LFDTGSA</td>
<td>SYGDSGSSAS</td>
<td>DGLLGLA</td>
</tr>
</tbody>
</table>

### Four Motifs of the RH12 OSM

<table>
<thead>
<tr>
<th>I</th>
<th>II</th>
<th>III</th>
<th>IV</th>
</tr>
</thead>
<tbody>
<tr>
<td>FSDGSPQ</td>
<td>KGBELL</td>
<td>FLDGLY</td>
<td>TFNEYTDSL</td>
</tr>
<tr>
<td>FTDGSSA</td>
<td>LVEQAR</td>
<td>YTDGSL</td>
<td>RGNGKADLA</td>
</tr>
<tr>
<td>FTDASS</td>
<td>QAELE</td>
<td>YTDGQQ</td>
<td>--NEVQDH</td>
</tr>
<tr>
<td>YTDGSSC</td>
<td>RAELI</td>
<td>YTDGSQ</td>
<td>RGMRKADQA</td>
</tr>
<tr>
<td>YTDGSSS</td>
<td>RACQV</td>
<td>FSDSLS</td>
<td>--NEVQCDM</td>
</tr>
<tr>
<td>WTDGS</td>
<td>DKEET</td>
<td>RTDHTH</td>
<td>-DNHFADFL</td>
</tr>
<tr>
<td>ETDASDD</td>
<td>DBTL</td>
<td>RTDHTH</td>
<td>KENCVADEL</td>
</tr>
<tr>
<td>TTDASDV</td>
<td>EKELL</td>
<td>SDEHQPL</td>
<td>RENGKIAEF</td>
</tr>
<tr>
<td>PSDRLLL</td>
<td>LREIN</td>
<td>EIDSKLS</td>
<td>ALMPADDPS</td>
</tr>
<tr>
<td>FADAT</td>
<td>TABELL</td>
<td>GTDNSNV</td>
<td>YEDMGC</td>
</tr>
<tr>
<td>YVDSDWA</td>
<td>EAEM</td>
<td>YEDMGC</td>
<td>-ENQLADIF</td>
</tr>
<tr>
<td>PTDGSL</td>
<td>RMEEL</td>
<td>STDSQVY</td>
<td>--NERCDEL</td>
</tr>
</tbody>
</table>
APPENDIX II

GLOSSARY

Analogous - similar in function, but not necessarily ancestrally related

Bioinformatics - the use of computational methods to perform hypothesis-driven research that generates new knowledge from existing biological databases

Convergence - process of evolving towards a similar point

Dayhoff Criteria - protein sequences that are >30% identical are considered homologous

Divergence - process of evolving away from a common ancestor

Genomics - the analysis of genomic sequence data, especially through the use of computers

GLOB - Globin

Global Alignment method - method that attempts to align entire length of protein sequences to each other

HMM - Hidden Markov Model; statistical model that can be used to represent a multiple sequence alignment by attempting to capture common patterns of residue conservation

Homologous - related by descent from a common ancestor

KIN - Kinase

Local Alignment method - method that attempts to align/locate the most conserved regions, or motifs, among a group of protein sequences

Motif - region of approximately 1-9 residues that remains conserved among a group of sequences due to functional or structural importance

MIR - see Motif-Intervening-Region

Motif-Intervening Region (MIR) - less conserved regions of amino acids that separate motifs

Multiple Alignment Method - method that attempts to align three or more sequences to each other; global and local are two types of multiple alignment methods

Ordered-Series-of-Motifs (OSM) - multiple motifs present in a colinear order among a group of sequences

OSM - see Ordered-Series-of-Motif
Pairwise Alignment method - method that attempts to align two sequences to each other

Percent Identity - percentage of identical residues shared between two sequences

PRO - Protease

RDDP - See RNA-Dependent DNA Polymerase

RDRP - See RNA-Dependent RNA Polymerase

RH - Ribonuclease H

RNA-Dependent DNA Polymerase (RDDP) - uses an RNA template to synthesize DNA; reverse transcriptase is an example of an RDDP

RNA-Dependent RNA Polymerase (RDRP) - uses an RNA template to synthesize RNA; large (L) protein is an example of an RDRP

RT - Reverse Transcriptase

Sequence Similarity - score assigned to a pair of protein sequences that takes into account the level of conservative substitutions and the ease of converting from one codon to another.
VITA

Graduate College
University of Nevada, Las Vegas

Julianna Hudak

Local Address:
4497 McMillan Road
Las Vegas, Nevada 89121

Degrees:
Bachelor of Science, Biology, 1996
University of Nevada, Las Vegas

Special Honors and Awards:
Department of Energy,
Travel Grant, 1999

Biological Association of Graduate Students, University of Nevada, Las Vegas,
Travel Grant, 1999

Publications:


Thesis Title:
Do RNA-Dependent Polymerases Share Common Ancestry? A Bioinformatic Approach

Thesis Examination Committee:
Chairperson: Dr. George Plopper, Ph.D.
Research and Thesis Advisor: Dr. Marcella McClure, Ph.D.
Committee Member: Dr. Stephen Carper, Ph.D.
Graduate Faculty Representative: Dr. Aimee Govett, Ph.D.