Ola Snøve Jr.

Hardware-accelerated analysis of non-protein-coding RNAs

Dr.philos. thesis 2005:196

Faculty of Information Technology, Mathematics and Electrical Engineering
Department of Computer and Information Science

NTNU

Hardware-accelerated analysis of non-protein-coding RNAs
Ola Snøve Jr.

Norwegian University of Science and Technology
Faculty of Information Technology, Mathematics and Electrical Engineering
Department of Computer and Information Science

Dr.philos. thesis 2005:196

ISBN 82-471-7287-9 (printed version)
ISBN 82-471-7286-0 (electronic version)
ISSN 1503-8181

To Thea.

Abstract

A tremendous amount of genomic sequence data of relatively high quality has become publicly available due to the human genome sequencing projects that were completed a few years ago. Despite considerable efforts, we do not yet know everything that is to know about the various parts of the genome, what all the regions code for, and how their gene products contribute in the myriad of biological processes that are performed within the cells. New high-performance methods are needed to extract knowledge from this vast amount of information.

Furthermore, the traditional view that DNA codes for RNA that codes for protein, which is known as the central dogma of molecular biology, seems to be only part of the story. The discovery of many non-protein-coding gene families with housekeeping and regulatory functions brings an entirely new perspective to molecular biology. Also, sequence analysis of the new gene families require new methods, as there are significant differences between protein-coding and non-protein-coding genes.

This work describes a new search processor that can search for complex patterns in sequence data for which no efficient lookup-index is known. When several chips are mounted on search cards that are fitted into PCs in a small cluster configuration, the system's performance is orders of magnitude higher than that of comparable solutions for selected applications. The applications treated in this work fall into two main categories, namely pattern screening and data mining, and both take advantage of the search capacity of the cluster to achieve adequate performance. Specifically, the thesis describes an interactive system for exploration of all types of genomic sequence data. Moreover, a genetic programming-based data mining system finds classifiers that consist of potentially complex patterns that are characteristic for groups of sequences. The screening and mining capacity has been used to develop an algorithm for identification of new non-protein-coding genes in bacteria; a system for rational design of effective and specific short interfering RNA for sequence-specific silencing of protein-coding genes; and an improved algorithmic step for identification of new regulatory targets for the microRNA family of non-protein-coding genes.

Preface

This dissertation is submitted to the Norwegian University of Science and Technology (NTNU) in partial fulfillment of the requirements for the degree of *Doctor philosophiae*.

My main contributions are contained in the original research articles that have been, or will be published in peer-reviewed scientific journals. The first chapters of the thesis introduce the various topics, and provide readers with the context that is out of the papers's scope.

I will use this opportunity to outline the unusual history behind this work, and acknowledge the many extraordinary people who have been involved. This work is more than anything testimonial to our success as a team, and I reflect this by using the plural pronoun in the remaining parts of this thesis.

I am indebted to you all.

When I wrote my master's thesis on numerical integrators in December 2000, I rejected an offer to become a graduate student because I was reluctant to postpone my commercial ambitions, and afraid that a doctorate in mathematics would be of limited value going forward. I left NTNU thinking that my academic career was over, but a series of fortunate events have enabled me to work in mathematics, electronics, programming, and molecular biology; practice teamwork and leadership; and combine science with business.

It has been a memorable experience.

In my opinion, Fast Search & Transfer (FAST) was the most exciting Norwegian technology startup around Y2K. After a short conversation at his office, Professor Arne Halaas recommended me to Torstein Heggebø who led FAST's R&D in Trondheim at the time. I do not know what convinced them, but I presume that my Mechanical Engineering degree had nothing to do with the decision to employ me as a Systems Engineer on 1 January 2001.

I was assigned to a team of incredibly talented scientists in the hardware development group whose goal was to implement a very-large-scale

integration (VLSI) version of FAST's pattern matching chip (PMC) architecture. I can still remember how impressed I was when I met the chip's chief architect, Olaf Birkeland, whom I still regard as the best technologist I know. My first task was to implement a software simulator of the chip, which was challenging as I had limited knowledge of both C++ programming and integrated circuits at the time.

In May 2001, FAST decided that the hardware technology had to be funded and run independent from the company's core business. In an effort to attract investors, we immediately focused on proof-of-concept applications in the genomics sector, as complex pattern analysis in large volumes of unstructured data is ideal for the chip's architecture. Collaborations with several biologists emerged in the autumn of 2001, most notably with the groups of Professor Hans Krokan at NTNU and the late Professor Erling Seeberg at University of Oslo (UIO). Following successful work on molecular biology applications, Interagon AS was established on 18 January 2002, with 20 million Norwegian Kroners, or about 3 million United States Dollars, in seed capital.

John Lervik, Torstein Heggebø, Torbjørn Kanestrøm, and Børge Svingen made important contributions, in addition to Olaf Birkeland, Ståle Fjeldstad, Pål Sætrom, Magnar Nedland, and Håkon Humberset who became my colleagues in Interagon. I was given the opportunity to lead our Trondheim office and became responsible for a team that comprised Fjeldstad, Sætrom, Nedland, Humberset, and myself.

When the development of the PMC reached completion, we had already worked with domain-specific applications since Interagon's inception. Thomas Grünfeld was contracted to evaluate Interagon's business strategy, and was later employed as CEO in April 2003. Grünfeld enforced a milestone-oriented strategy that aimed to scientifically document our pattern mining technology's strengths in relation to important applications in molecular biology. We quickly became interested in the analysis of short sequences for downregulation of genes. Since then, we have attended numerous conferences, collaborated with leading researchers, and worked with international biotechnology companies on non-coding RNA with regulating properties. Unfortunately, the commercial work could not be included in this thesis, but will be the subject of upcoming publications co-authored by Interagon employees. The collection of papers that constitutes the scientific work of this thesis should nevertheless demonstrate our accomplishments in the field.

I do not know what Karl Marx meant when he said that men's ideas are the most direct emanations of their material state, but he married the daughter of a baron, and lived of money that stemmed from his collabo-

rator Engel's family business. Without further comparison, I would like to acknowledge Interagon for financing this academic pursuit. Parts of the work were also supported by the Norwegian Research Council, grants 151899/150 and 151521/330, and NTNU's bioinformatics platform in the national functional genomics programme (FUGE).

Interagon's Board of Directors, including Øyvin Brøymer, Hans Krokan, Erling Seeberg, Jens Vig, Per Thrane, and John Lervik, has been very supportive of this work. This is also the case for large shareholders such as Erik Must, Jan-Erik Hareid, and Arne Halaas.

I would like to express my deepest gratitude to Professors Hans Krokan, Arne Halaas, and Finn Drabløs who have gone far beyond their duties to support us.

While working on my master's degree, I studied the work of mathematicians and physicists that lived hundreds of years ago. In the present work, I have been fortunate to witness how a new research area has emerged. I picked up the first papers on RNA interference a few months before Science Magazine celebrated the pathway as 2002's Breakthrough of the Year. These papers caught our attention, and we started to work on the efficacy and specificity of short interfering RNAs. Since then, we have published several papers on this and related subjects, and one of these—the one that is referred to as Paper VII in the thesis—was actually among the top 20 downloaded papers in Biochemical and Biophysical Research Communications in 2004. The role of non-protein-coding RNAs can almost be viewed as a disruptive technology in that their action provides new means to understand the mechanisms that cells perform. For example, many papers have shown how microRNAs are closely related to different disease mechanisms, and that brings encouragement to those of us that are hoping for new ways to fight some of the worst genetic diseases.

These are exciting times.

It takes more than one person, and sometimes more than one team, to publish in the life sciences today. I want to express my earnest gratitude to our co-authors at NTNU and UIO, including Arne Halaas, Børge Svingen, Torgeir Holen, Ola Sætrom, Ragnhild Sneve, Knut Kristiansen, Torbjørn Rognes, and Erling Seeberg. Their contributions are greatly acknowledged.

Without the collaboration with my colleagues in Interagon, this work would not have been possible. I have always had tremendous respect and appreciation for the talent in our organization, but I realize that we are better at everything we do now than we were five years ago. This process is perhaps best illustrated by Magnar Nedland who have matured into a programmer with an admirable attitude to industrial development.

My collaboration with Pål Sætrom deserves some mention. We grew up together, played on the same soccer team for years, and I was his best man when he got married. In recent years, we have established an incredibly productive professional relationship that I hope we can maintain.

Thomas Grünfeld's focus on processes, something I suspect resulted from his years with McKinsey&Company, has been immensely valuable. I have truly enjoyed our numerous discussions about science, business, and life in general.

To my mother and father, brothers and sisters, and grandparents: I will never become the "real thing" with a white doctor's coat, as some of you had hoped for, but at least you cannot see the difference on airline tickets.

Thank you for genes and ambition; friendship, love, and support.

I moved to Oslo to be with Marianne, and much of this thesis has been written in our kitchen with one eye on the computer and the other one on her growing belly.

I am forever grateful.

Ola Snøve Jr.
Oslo, 14 April 2005

On joint authorships

The scientific responsibilities in Interagon have been divided between Olaf Birkeland, Pål Sætrom, and myself. Birkeland and Sætrom's work is mostly related to the search processor and the machine learning system, respectively, whereas my main occupation has been the research and development of biological applications. Needless to say, there have been times when others have helped to fulfill my responsibilities and *vice versa*, but as the title of the thesis suggests, my main contribution is new methods for analysis of short RNA sequences in general, and for the study of the efficacy and specificity of short interfering RNAs—the molecules that mediate silencing in mammals—in particular.

Ola Snøve Jr.
Oslo, 26 July 2005

Contents

List of figures

List of tables

List of papers

Paper I Arne Halaas, Børge Svingen, Magnar Nedland, Pål Sætrom, Ola Snøve Jr., and Olaf Renè Birkeland. A recursive MISD architecture for pattern matching. *IEEE Trans. on VLSI Syst.*, 12 (7):727–734, 2004.

Paper II Olaf René Birkeland, Ola Snøve Jr., Arne Halaas, Ståle H. Fjeldstad, Magnar Nedland, Håkon Humberset, and Pål Sætrom. A MISD architecture in a pattern-mining supercomputing cluster. *IEEE Trans. on Comp.*, 2005. Submitted.

Paper III Ola Snøve Jr., Håkon Humberset, Olaf René Birkeland, and Pål Sætrom. Sequence Explorer: interactive exploration of genomic sequence data, 2005. Manuscript.

Paper IV Pål Sætrom, Ragnhild Sneve, Knut I. Kristiansen, Ola Snøve Jr., Thomas Grünfeld, Torbjørn Rognes, and Erling Seeberg. Predicting non-coding RNA genes in *Escherichia coli* with boosted genetic programming. *Nucleic Acids Res.*, 33(10):3263–3270, 2005.

Paper V Ola Snøve Jr. and Torgeir Holen. Many commonly used siRNAs risk off-target activity. *Biochem. Biophys. Res. Commun.*, 319(1):256–263, 2004.

Paper VI Pål Sætrom and Ola Snøve Jr. A comparison of siRNA efficacy predictors. *Biochem. Biophys. Res. Commun.*, 321(1):247–253, 2004.

Paper VII Ola Snøve Jr., Magnar Nedland, Ståle H. Fjeldstad, Håkon Humberset, Olaf R. Birkeland, Thomas Grünfeld, and Pål Sætrom. Designing effective siRNAs with off-target control. *Biochem. Biophys. Res. Commun.*, 325(3):769–773, 2004.

Paper VIII Ola Sætrom, Ola Snøve Jr., and Pål Sætrom. Weighted sequence motifs as an improved seeding step in microRNA target prediction algorithms. *RNA*, 11(7):995–1003, 2005.

Chapter 1

Introduction

Doctoral theses usually comprise two parts; one that reviews the research field, and another that contains original research articles that resulted from the work. As mentioned in the preface, our work spans many subjects, and I therefore had to make difficult choices on how to present it. This chapter defines the aim of our study, outlines the thesis's structure, and lists information about papers and additional work.

1.1 Aim of study

In the broadest sense, the goal of this study has been threefold, namely to complete the development of the PMC; understand molecular biology to identify suitable problems; and implement hardware-accelerated domain-specific applications. In the process, we have defined specific aims for each part as will be described in the following.

I. *Develop a search cluster using special-purpose hardware.* Arne Halaas and and Børge Svingen's PMC search core was designed in the nineties, and Olaf Birkeland constructed the first PCI card prototypes a few years later. Large-scale screening and pattern mining typically require the performance of supercomputers, and an important part of our work was therefore to realize the architecture's full potential by building a production-scale solution.

   Our aim was to (i) develop an application-specific integrated circuit (ASIC) version of the PMC based on the existing field-programmable gate array (FPGA) prototype; (ii) integrate multiple ASICs on search cards that adhere to the PCI standard; (iii) build PMC servers with several PCI cards in each machine; (iv) obtain linear scalability with a

cluster of PMC servers; and (v) scientifically document the system, its features, and its performance.

II. *Study molecular biology in general, and siRNAs and miRNAs in particular.* Bioinformatics requires competence in molecular biology and informatics alike, and an important part of graduate studies is therefore for an informaticist to learn molecular biology, or *vice versa*. Our group consists mainly of informaticists with formal education in electronics and software, whereas I am a mechanical engineering student turned mathematician.

Important subgoals of this work was therefore to (i) learn the basic concepts of molecular biology; and (ii) identify applications suitable for our technology. Following Science Magazine's celebration of RNA interference (RNAi) as 2002's Breakthrough of the Year, we saw many potential applications within that field for our technology. Consequently, we aimed to (iii) identify the most promising applications in the analysis of short non-coding RNA (ncRNA); and (iv) get a detailed overview of the literature on ncRNA in general and short interfering RNA (siRNA) and microRNA (miRNA) in particular.

III. *Develop screening and pattern mining applications.* An ideal application for the PMC is one that deals with complex patterns in large volumes of unstructured string data. The search problem typically involves a relatively static dataset for which no efficient lookup index is known; consist of a large number of readily available queries; and is dividable into independent subproblems that can be solved in parallel. The popularity of search heuristics such as the basic local alignment search tool (BLAST) illustrates that screens for patterns that contain long stretches of perfect matches can be effectively solved in software. Short patterns with high complexity, however, is ideally suited for our technology.

Based on our technical advantage and the literature study, we therefore chose to develop applications for (i) interactive screening of complex patterns; and (ii) analysis of important sequence properties of short ncRNA.

1.2 Outline of thesis

I have tried to balance the material of this thesis to reflect our interdisciplinary work in the past years. The presentation reflects how I want

Chapter 2:	Paper I (chip)	→	Paper II (cluster)	→	Paper III (screening)
Chapter 3:	Paper IV (mining)				
Chapter 4:	Paper V (screening)	→	Paper VI (mining)	→	Paper VII (screening and mining)
Chapter 5:	Paper VIII (mining)				

Figure 1.1: The papers span four main categories, including the chip's architecture, the assembly of many chips into a high-performance cluster, and screening and mining applications for sequence analysis.

this work to be accessible to interested informaticists and biologists alike. To enable readers to approach potentially unfamiliar material, I have included a small glossary in the back.

Chapter 2 introduces Interagon's ASIC for large-scale pattern matching, and explains how the technology is used for pattern screening and mining. Topics from machine learning are discussed with particular emphasis on pattern evolution using genetic programming (GP). Note that the description of our hardware's search architecture in Paper I is very different from that in Chapter 2, as Paper I presents a mathematical framework for discussing the chip's architecture. In the interest of accessibility, I have refrained from using a rigorous mathematical formulation throughout this thesis.

Chapter 3 discusses non-protein-coding RNAs in general. Our intention is to give readers an opportunity to get an overview of the research field, and to understand how our papers fit into the bigger picture. Paper IV is about ncRNA genes in *Escherichia coli*, and the chapter therefore emphasizes the computational challenges with ncRNA gene finding.

One of several interesting non-protein-coding RNA species is siRNAs and their role in RNAi. Chapter 4 is about this class of molecules and how RNAi has become the preferred technique for sequence-specific silencing of genes with promising therapeutic applications. We have used our technology to analyze the efficacy and specificity of siRNAs, and the details are contained in Paper V, Paper VI, and Paper VII.

Finally, miRNAs are the endogenous counterparts of siRNAs, and Chapter 5 is about the genes, biogenesis, and targets of this abundant class of

ncRNA with regulating properties. We also review algorithms for gene and target prediction with particular emphasis on the latter, as principles for translational suppression was used in Paper VIII to develop an improved seeding step for such algorithms. This algorithm is based on weighted patterns that result from the GP-based approach that was also used in Paper IV and Paper VI.

Figure 1.1 shows how the different papers are connected to each other. Note that the papers are not listed chronologically, but presented in the order that naturally corresponds with the main categories we have addressed and thereby also with the disposal of the thesis.

1.3 Paper abstracts

The research that will be described throughout this thesis is contained in a subset of recent publications from our group. To make the reading easier, I will cite these papers as Paper I through Paper VIII, as defined in the List of Papers. Reprints of the original papers are attached as the second part of this thesis, and their abstracts are given in the following.

Paper I *A recursive MISD architecture for pattern matching.* Many applications require searching for multiple patterns in large data streams for which there is no preprocessed index to rely on for efficient lookups. An multiple instruction stream-single data stream (MISD) VLSI architecture that is based on a recursive divide and conquer approach to pattern matching is proposed. This architecture allows searching for multiple patterns simultaneously. The patterns can be constructed much like regular expressions, and add features such as requiring subpatterns to match in a specific order with some fuzzy distance between them, and the ability to allow errors according to prescribed thresholds, or ranges of such. The current implementation permits up to 127 simultaneous patterns at a clock frequency of 100 MHz, and does 1.024×10^{11} character comparisons per second.

Paper II *A MISD architecture in a pattern-mining supercomputing cluster.* Multiple instruction stream-single data stream (MISD) architectures have not found many practical applications in supercomputing. We present a multiple instruction stream-multiple data stream (MIMD) cluster implementation that uses MISD search processors with extreme pattern mining performance. For regular expressions, a single search processor is three orders of magnitude faster than a modern CPU

running `nr-grep`. We use PCI cards that hold sixteen search processors with local memory to build a relatively small cluster of five PCs with six PCI cards each, and this cluster can handle anything between 64 independent queries at 48 GB per second or 30,720 independent queries at 100 MB per second. The cluster's performance characteristics are such that we can easily scale the system to obtain higher performance with containable overhead. Because this may be the first commercially used MISD implementation we discuss several applications in molecular biology, seismic data processing, network surveillance, and financial transaction analysis.

Paper III *Sequence Explorer: interactive exploration of genomic sequence data.* Current solutions for complex motif searching in DNA and protein sequences are not interactive as users usually wait tens of seconds before the results can be viewed. We propose a hardware-accelerated client-server solution that is fast enough to retain the interactive feeling even when screening whole genomes. We structured our framework for interactive sequence analysis around query, dataset, filter, and result presentation modules. The query and dataset specification enable simultaneous, interactive screening of multiple complex queries against several datasets. The filters impose restrictions such as only allowing hits to be reported if they occur in coding regions, and the different result presentations include histograms and hit lists. Our results show that interactive searching is possible even though response times vary significantly depending on filter, network bandwidth and hit frequencies. With a relatively small server, we obtain response times of about one and a half second on gigabytes of data when queries are sufficiently complex to avoid network bottlenecks due to high hit frequencies.

Paper V *Many commonly used siRNAs risk off-target activity.* Using small interfering RNA (siRNA) to induce sequence specific gene silencing is fast becoming a standard tool in functional genomics. As siRNAs in some cases tolerate mismatches with the mRNA target, knockdown of genes other than the intended target could make results difficult to interpret. In an investigation of 359 published siRNA sequences, we have found that about 75% of them have a risk of eliciting non-specific effects. A possible cause for this is the popular BLAST search engine, which is inappropriate for such short oligos as siRNAs. Furthermore, we used new special purpose hardware to do a transcriptome-wide screening of all possible siRNAs, and show that many

unique siRNAs exist per target even if several mismatches are allowed. Hence, we argue that the risk of off-target effects is unnecessary and should be avoided in future siRNA design.

Paper VI *A comparison of siRNA efficacy predictors.* Short interfering RNA (siRNA) efficacy prediction algorithms aim to increase the probability of selecting target sites that are applicable for gene silencing by RNA interference. Many algorithms have been published recently, and they base their predictions on such different features as duplex stability, sequence characteristics, mRNA secondary structure, and target site uniqueness. We compare the performance of the algorithms on a collection of publicly available siRNAs. First, we show that our regularized genetic programming algorithm GPboost appears to have a higher and more stable performance than other algorithms on the collected datasets. Second, several algorithms gave close to random classification on unseen data, and only GPboost and three other algorithms have a reasonably high and stable performance on all parts of the dataset. Third, the results indicate that the siRNAs' sequence is sufficient input to siRNA efficacy algorithms, and that other features that have been suggested to be important may be indirectly captured by the sequence.

Paper VII *Designing effective siRNAs with off-target control.* Successful gene silencing by RNA interference requires a potent and specific depletion of the target mRNA. Target candidates must be chosen so that their corresponding short interfering RNAs are likely to be effective against that target and unlikely to accidentally silence other transcripts due to sequence similarity. We show that both effective and unique targets exist in mouse, fruit fly, and worm, and present a new design tool that enables users to make the trade-off between efficacy and uniqueness. The tool lists all targets with partial sequence similarity to the primary target to highlight candidates for negative controls.

Paper IV *Predicting non-coding RNA genes in Escherichia coli with boosted GP.* Several methods exist for predicting non-coding RNA (ncRNA) genes in *Escherichia coli* (*E. coli*). In addition to about sixty known ncRNA genes excluding tRNAs and rRNAs, various methods have predicted more than thousand ncRNA genes, but only 95 of these candidates were confirmed by more than one study. Here we introduce a new method that uses automatic discovery of sequence patterns to predict ncRNA genes. The method predicts 135 novel candi-

dates and confirms 152 existing predictions. We test sixteen predictions experimentally, and show that twelve of these are actual ncRNA transcripts. Six of the twelve verified candidates were novel predictions. The relatively high confirmation rate indicates that many of the untested novel predictions are also ncRNAs, and we therefore speculate that *E. coli* contains more ncRNA genes than previously estimated.

Paper VIII *Weighted sequence motifs as an improved seeding step in microRNA target prediction algorithms.* We present a new microRNA target prediction algorithm called TargetBoost, and show that the algorithm is stable and identifies more true targets than do existing algorithms. TargetBoost uses machine learning on a set of validated microRNA targets in lower organisms to create weighted sequence motifs that capture the binding characteristics between microRNAs and their targets. Existing algorithms require candidates to have (i) near-perfect complementarity between microRNAs's 5′ end and their targets; (ii) relatively high thermodynamic duplex stability; (iii) multiple target sites in the target's 3′ UTR; and (iv) evolutionary conservation of the target between species. Most algorithms use one of the two first requirements in a seeding step, and use the three others as filters to improve the method's specificity. The initial seeding step determines an algorithm's sensitivity and also influences its specificity. As all algorithms may add filters to increase the specificity, we propose that methods should be compared before such filtering. We show that TargetBoost's weighted sequence motif approach is favorable to using both the duplex stability and the sequence complementarity steps. TargetBoost is available as a web-tool from `http://www.interagon.com/demo/`.

1.4 Other publications

I have also co-authored a detailed description of PMC concepts that is available to interested readers upon request (Birkeland and Snøve Jr. 2002). Other members of our group have previously published papers on data mining in time series using early versions of our boosted genetic programming-based machine learning system (Hetland and Sætrom 2002, 2003a,b; Sætrom and Hetland 2003a,b). Sætrom (2004) published the first application of the boosted genetic programming-algorithm that we later used in Paper VI, Paper IV, and Paper VIII.

1.5 Supplementary material

For demonstration purposes, we maintain limited versions of the applications for interactive search presented in Paper III, siRNA design described in Paper VII, and microRNA target prediction introduced in Paper VIII at `http://www.interagon.com/demo/`. A tutorial on the screening application in Paper III is also available.

Chapter 2

Screening and mining

Pattern matching in strings is an important research field with many practical applications, including signal processing, text retrieval, data mining, pattern recognition, computational biology, and more (Navarro 2001). Online pattern matching refers to situations where no persistent index can be built to facilitate search algorithms that depend on efficient lookups (Navarro and Raffinot 2002). We focus on applications that require multiple patterns to be simultaneously screened against large volumes of unstructured data. Furthermore, the patterns are often approximate, which means that the matches do not have to be exact—that is, there may be various levels of discrepancy between the query and the patterns that should be matched.

A detailed overview of string matching is out of scope for this thesis, and we therefore refer interested readers to books (for instance Gusfield 1997) or review articles (for instance Michailidis and Margaritis 2002) on the subject. This chapter will introduce Interagon's pattern matching chip, and describe its role in a high-performance cluster. Furthermore, we demonstrate the cluster's potential in Sequence Explorer, an interactive screening application that represents an alternative method for motif searching in sequence data. Finally, we describe simple and advanced classification methods in relation to our choice of a genetic programming-based (GP) system for pattern mining applications.

2.1 A special-purpose search processor

The pattern matching chip (PMC) is a patented (Fast Search & Transfer ASA 2000a,b) application-specific integrated circuit (ASIC) that can screen for the occurrence of up to 64 independent patterns in a data stream at 100

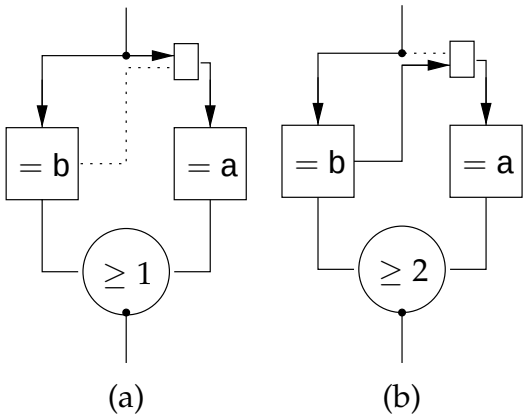

Figure 2.1: Data flow, pattern matching and result gathering for queries (a) `x|y` and (b) `xy`.

MB per second. It consists of a data distribution tree that feeds a row of processing elements with data, and a result processing tree that outputs the results. Figure 2.1 shows how the architecture matches the queries `a|b` and `ab`, that is, either an `a` or a `b` in the first query, and the string `ab` in the second query. The distribution tree uses (a) parallel or (b) sequential distribution of characters from the data stream to the processing elements that does the matching operations. In (a), the processing elements receive the same characters, and it is therefore impossible for both elements to report a match at the same time. The result processing tree may ensure that either of the characters match by checking that the sum of matches is at least one—that is, the tree node performs a boolean `OR` operation on the results. In (b), the rightmost processing element matches the first part of the expression, whereas the leftmost processing element match the consecutive character. A boolean `AND` operation on the result processing node's part ensures that the architecture matches the full expression.

Even though the queries in Figure 2.1 are simple examples, they illustrate how the architecture consists of three parts, namely a data distribution tree, some processing elements, and a result processing tree. In our final implementation, there are 1,024 processing elements that can be combined to match complex queries, or divided into at most 64 blocks that match independent queries in parallel. Additional functionality in the processing elements and result processing tree enables the architecture to match much more complex expressions, such as queries that contain wildcards, skips, and repeats in addition to requirements for n out of m subparts, one expression before or near another, and more.

Patterns are usually expressed in Interagon's query language (IQL; Interagon AS 2002), which is very similar to regular expressions (Friedl 2002). These are subsequently parsed and mapped to an intermediate language that is used to construct the binary configuration that the chip receives.

See Paper I for a thorough presentation of the PMC implementation, including a mathematical framework for discussing the architecture's functionality in terms of hit functions. The paper also describes how the functionality is implemented in hardware. The example in Figure 2.1 was adapted from Birkeland and Snøve Jr. (2002), which was written to explain the PMC's functionality to students.

2.2 Many chips in cluster

Even though a single PMC is relatively fast, the PMC's full potential is only reached when several chips are working together to solve a specific search problem. We specifically designed the PMC to have a low power consumption such that several PMCs could be fitted in a single PC using search cards that can hold sixteen chips each. The cards adhere to PCI standards, and regular workstations can hold up to six cards, bringing the total number of PMCs in every machine to 96. In Paper II, we present a cluster of five machines, but in principle, the cluster could be arbitrary large. Provided that the problem can be divided into disjoint subproblems that can be solved independently with minimal overhead, additional machines can be added to the cluster to achieve near-linear scalability.

To achieve maximum parallelism, we had to design a dense system that could fit as many PMCs as possible. On the one hand, data duplication gives our system a theoretical performance of 30,720 queries at 100 MB per second if query throughput is important. On the other hand, query duplication yields a potential search throughput of at most 64 queries at 48 GB per second if an application requires search speed. Note that the maximum query throughput depends on the number of processing elements that are needed to match the queries, whereas the maximum search speed can only be attained when the data volume is smaller than the size of each chip's individual memory, which is currently 128 MB. Note, however, that practical applications require that we strike the right balance between query throughput and search speed.

Paper II compares the cluster's performance with that of relevant algorithms for short query screening, and gives more details regarding the technical design choices we have made during the development. Possible applications for our pattern screening and mining technology include, but

is not limited to, seismic data processing, financial knowledge mining, and network surveillance, in addition to the applications in molecular biology that will be discussed throughout this thesis. In Paper II, we also identified some possible improvements to our search card that may increase the performance in existing applications, and possibly facilitate other applications than those we have identified so far. For example, an IO processor at each card makes it possible to do some computations locally to avoid some information transfer between the CPU and the cards. Furthermore, some real-time screening applications may require that the card has a network interface.

2.3 Interactive screening

In an effort to exploit the performance of our search cluster, we wanted to develop an application for interactive searching in biological data. Without further comparison, popular search heuristics such as BLAST (Altschul et al. 1990) running on large publicly funded clusters typically take tens of seconds to complete a search, at least for relatively short queries. Even though homology searching is probably the most important search problem in molecular biology, Betel and Hogue (2002) demonstrated that pattern matching was valuable when identifying characteristic genetic targets in a cancer.

Figure 2.2 shows a screenshot from our application, the Sequence Explorer, when screening two queries against human chromosomes one and two. Note that while only the query `ACTGCACT` is visible to the user in the pattern pane, the results are updated for both queries simultaneously. A filter constraints the search, for instance by requiring that all hits belong to regions annotated as mRNAs, as is the case in figure 2.2. Paper III describes the application further, but it is important to note that queries are automatically scheduled for submission to the server, and the familiar "submit" button was therefore unnecessary. To avoid excessive scheduling, we have introduced a submission delay, that is, a quiet time frame from the last character entry to query submission. Furthermore, to avoid unnecessary overhead, an ongoing search is automatically aborted if its results have not been reported at the time that its corresponding query is altered in the client.

Table 2.1 shows that the main performance bottleneck is the query's hit rate. Sequence Explorer's performance can potentially depend on seven factors that are listed in order of occurrence from query submission to result presentation:

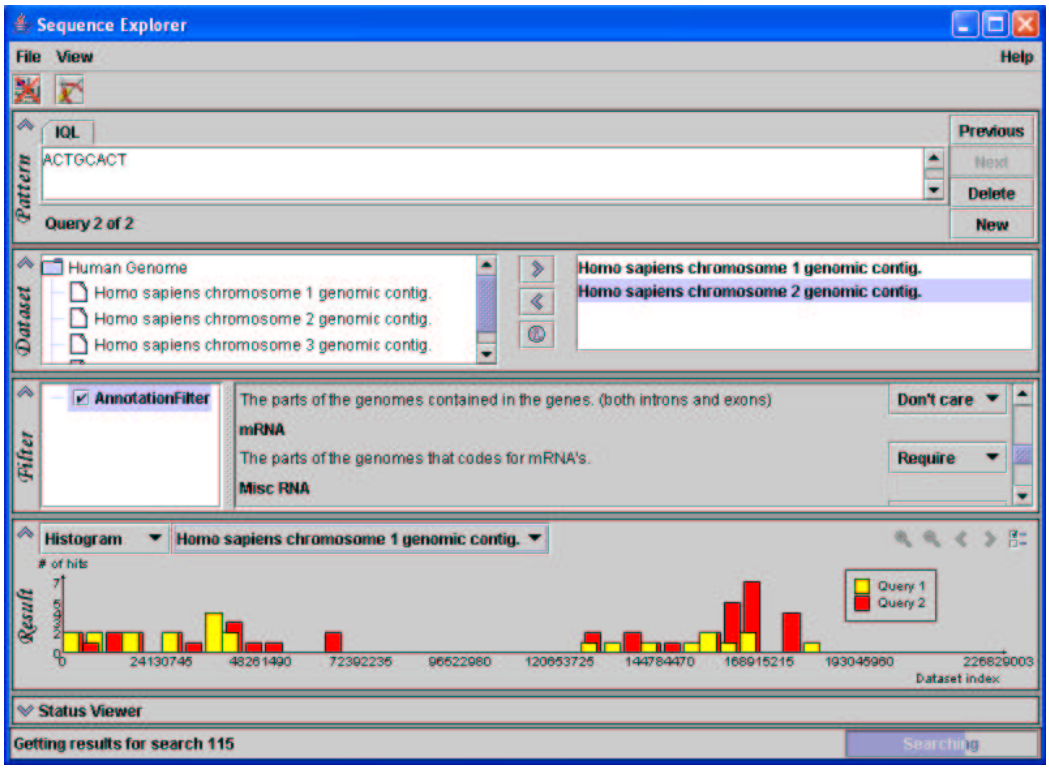

Figure 2.2: Screenshot from Sequence Explorer. The client has separate panes for pattern input, dataset selection, filter constraints, and result presentation.

Table 2.1: Sufficient query complexity is required to limit the hit rates that otherwise destroy the interactive experience. The small dataset is chromosome 1, whereas the larger dataset is the entire human genome.

	≈ 0.2 GB		≈ 3 GB	
chars	thousand hits	time	thousand hits	time
1	66,064	86.2 s	891,778	1962.4 s
3	8,413	11.1 s	115,927	251.1 s
5	1,509	2.7 s	20,320	25.8 s
7	512	1.8 s	6,598	8.7 s
11	207	1.6 s	2,596	4.1 s
≥13	<145	1.4 s	<1,820	3.3 s

(i) *Network transfer from client to server.* The search configuration, including information about active queries, datasets, filters, and result views must be transfered to the server. In our experience, this factor is negligible when disregarding the submission delay that is currently set to 0.3 seconds. This could, however, change if the application is run in a slow network environment.

(ii) *Parsing and mapping.* Queries are parsed and mapped from IQL expressions to PMC configurations by the server. In principle, we could have delegated this work to the client, but experiments where we varied the complexity of the queries showed that this factor is negligible.

(iii) *Distribution of configurations.* As a single PMC can only handle 128 MB of data, we have to use several PMCs when the datasets are larger. Configurations must therefore be distributed to the machines, cards, and chips that ultimately perform the search.

(iv) *PMC search time.* With a clock frequency of 100 MHz and 128 MB of local memory, one pass through a PMC's data takes 1.28 seconds. For now, Sequence Explorer does not support duplication of data to enable faster searching. Screens that involve more than 128 MB of data is therefore bound to use at least 1.28 seconds.

(v) *Postprocessing of results.* The server collects results from the distributed PMCs, removes hits that do not satisfy the filter requirements, and constructs the results that were requested by the client. As noted in Paper III, filtering is computationally intensive and is disregarded for queries that results in tens of thousands of hits. Table 2.1 shows that simple queries of moderate lengths will always return more hits than is tolerated with filters, and the current implementation of filters is therefore only valuable for relatively complex queries.

(vi) *Network transfer from server to client.* We designed a minimal network protocol for this purpose, but network transfer of results is still Sequence Explorer's main bottleneck. When the user opts for histogram view and no filters, interactivity can usually be maintained if there are less than one million hits per request.

(vii) *Updates to the client's graphical user interface.* In addition to result views that must be updated, the client also receives information about the search progress from the server. This factor has negligible impact on Sequence Explorer's performance.

On a practical note, we do not expect that unspecific patterns that hit millions of genomic regions have much information value. If this is the case, it can be said that Sequence Explorer is interactive for practical purposes, as this type of queries can easily be aborted by the server to improve the overall performance.

Our work on Sequence Explorer introduced us to short oligonucleotides and their importance in molecular biology. This lead to our subsequent development of specialized applications for analysis of the short regulatory RNAs that are discussed in chapters 3, 4, and 5.

2.4 Pattern classifiers

Consider strings S_i that have been labeled according to their group membership y_i. As was the case in Paper VI, the strings may for instance be siRNAs that have been labeled effective (1) or ineffective (-1) in gene silencing experiments (see chapter 3 for details).

A hard classifier $f : (S_i, y_i) \rightarrow \{-1, 1\}$ assigns binary group memberships for unseen strings. Our patterns can be used as hard classifiers, as we evaluate them on the PMC that reports a hit (1) or not (-1) in a dataset. A classifier's performance depends on its ability to capture the characteristics that distinguish one class of sequences from others. Baldi et al. (2000) reviews various measures that exist for determining a classifier's accuracy. In our case, we quantify a pattern's ability to separate between positive and negative examples using correlation. Consider a pattern that correctly classifies some effective and ineffective siRNAs, whereas others that really are ineffective are assumed to be effective, and *vice versa*. That is, there are both true positive TP, true negative TN, false positive FP, and false negative FN predictions. The correlation is given by

$$\frac{\mathrm{TP} \cdot \mathrm{TN} - \mathrm{FP} \cdot \mathrm{FN}}{\sqrt{(\mathrm{TP} + \mathrm{FN})(\mathrm{TP} + \mathrm{FP})(\mathrm{TN} + \mathrm{FP})(\mathrm{TN} + \mathrm{FN})}}.$$

Furthermore, a classifier's sensitivity reflects its ability to correctly classify the positive examples, and is expressed as the ratio of true positives to the total number of positive examples, TP/(TP+FN). An algorithm that assigns positive labels to every sample has maximum sensitivity, but is still not a good classifier, as many of the positive predictions will be wrong. A classifier's specificity reflects its ability to correctly classify the negative examples, and is expressed as the ratio of true negatives to the total number of negative examples, TN/(TN+FP).

In reality, of course, experimental results are rarely black and white, but comes in shades of gray. An siRNA may for example knock down its target by sixty or eighty-five percent, so when predicting the efficacy of unseen siRNAs, we would like our classifier to report how confident we can be about the prediction. A soft classifier $f : (S_i, y_i) \rightarrow \mathbb{R}$ usually maps an unseen example to a real-valued interval, for example $(-1, 1)$. An siRNA that receives a score close to 1 may therefore be assumed to be effective with a high degree of confidence, and *vice versa*.

When using a soft classifier to assign binary class memberships to unseen samples, the outcome depends on the score threshold that defines positive and negative predictions. Decreasing the threshold will improve the sensitivity, but at the cost of a declining specificity. A receiver operating characteristics (ROC) curve plots a classifier's sensitivity versus its specificity (or one minus the specificity to maintain familiar axes) for a range of prediction thresholds. The ROC score, which corresponds to the area under the ROC curve, can be used to estimate the overall performance of a classifier, as we do in Paper VI and Paper VIII.

2.5 Pattern mining

Generally speaking, GP uses sexual recombination and spontaneous mutations of computer programs to breed a population of problem solvers (Koza 1992). First, an initial population of randomly generated programs are assigned a fitness score that measures their ability to solve the problem. Second, programs are selected based on some selection strategy—typically such that fitter programs have a higher probability of being selected. Third, some of these programs go unchanged to the next generation, whereas others participate in recombination or undergo mutations. The second and third steps are repeated until some predefined termination criteria are satisfied.

Motivated by the fact that our evolving programs can be efficiently screened against large data volumes by the PMC, these programs are patterns specified in Interagon's query language IQL (Interagon AS 2002). The optimal parameters of a run depend on the problem, but we typically select ninety percent of the patterns that participate in creating the next generation for recombination, whereas only one percent undergo mutation. Furthermore, we generally use populations that consists of 100–1,000 patterns, and obtain the final classifier within 50–200 generations.

Note that a classifier may perform exceptionally well on training data, but still fail miserably on unseen data, in which case we say that it does

not generalize well. Ideally, we want a machine learning algorithm to strike the right balance between its output's accuracy on a given training set, and the algorithms capacity to produce an infinitely complex solution for any set (Burges 1998). We are currently not limiting the complexity of our patterns—except for the natural limitation that results from using the PMC—but measure the classifiers's ability to perform well on unseen data using a technique called k-fold cross validation (Stone 1974). The training set is divided into k disjoint subsets of equal size, and an unbiased test performance is obtained from each of the k folds by training a classifier on the remaining $k-1$ folds. We normally use ten folds, as the average of k test values have proven to be a good measure of the generalized performance when $k \geq 10$ (Martin and Hirschberg 1996).

Pattern evolution, as described here, has the characteristics of a weak learner as it does not have the capacity to learn every twist of any dataset. The next section explains how the capacity can be increased by combining several patterns.

2.6 Weighted patterns in models

To obtain soft classifiers, as described in section 2.4, we may combine T hard classifiers $h_t(S)$ into an ensemble

$$f(S) = \sum_{t=1}^{T} \alpha_t h_t(S),$$

where $\alpha_t \in \mathbb{R}$ is the weight of the classifier h_t (Meir and Rätsch 2003). In fact, classifiers that perform only slightly better than random may be combined into ensembles whose performance is only limited by the quality of the training data (Kearns and Valiant 1994). There are several possibilities for selecting the weights α_t, but bagging and boosting are the most prominent. In bagging, the weights are set $1/T$, and the ensemble's performance therefore corresponds to the average of T classifiers (Breiman 1996). Boosting algorithms attempt to assign the weights iteratively as illustrated by the AdaBoost procedure that puts more effort into learning the difficult parts of the dataset as it proceeds (Freund and Schapire 1997).

Our base classifiers h_t are patterns obtained from GP, as described in Section 2.5, whereas we have our own AdaBoost-based boosting implementation called GPboost. An ensemble may consist of hundreds of weighted classifiers, but this generally depends on the problem. For example, we used ensembles of 100 classifiers in Paper IV, and the average of ten en-

sembles that consist of 20 and 25 classifiers in Paper VI and Paper VIII, respectively.

In the interest of good generalization, we may put several constraints on the algorithms to avoid overfitting. Occam's razor is a logical principle that states that one should not increase, beyond what is necessary, the number of entities required to explain anything. Regularization techniques in machine learning provide means to limit the complexity of the final classifier. While boosting avoids overfitting when applied to datasets with a limited noise level, they clearly do not generalize well when the training sets contain many misclassified examples or examples that deviate substantially from their true classification. Meir and Rätsch (2003) lists three different regularized algorithms that mend the problem. First, AdaBoost$_{\text{reg}}$ solves the problem iteratively by assigning mistrust parameters to examples that prove very difficult to learn, and the effect of these examples on the final classifier is consequently marginalized (Rätsch et al. 2001). Second, BrownBoost uses the same principle, but uses a predetermined number of iterations, and eliminates the hardest examples completely when the algorithm approaches the final number of iterations (Freund 2001). Third, SmoothBoost is similar to AdaBoost, but defines a threshold that is the maximum weight any example can receive, and the algorithm consequently places a limit on the importance of difficult examples (Servedio 2003). The regularized version of GPboost is called GPboost$_{\text{reg}}$ and is based on AdaBoost$_{\text{reg}}$.

2.7 Regularized algorithms

Moving from simple patterns to regularized models that contain numerous weighted patterns has a significant impact on the computer capacity requirements. A boosting algorithm needs to obtain T weak classifiers from its base learner. Consequently, as we use identical population sizes, this requires T times the capacity. Moving from GPboost as used in Paper VI to the regularized version in Sætrom (2004), the demand for computing power increases with a factor $k \times y$, where y is the number of regularization parameters to be tested. The parameter k stems from our using k-fold cross validation to optimize the choice of each regularization parameter. Sætrom (2004) used $k = 10$ and $y = 7$, which means that the improved performance of GPboost$_{\text{reg}}$ came at the expense of a run time that was about seventy times that of the unregularized GPboost. Note that regularization is unnecessary in modestly noisy data, which is illustrated by the fact that we did not opt for regularization in Paper VI, as the marginally improved

performance seen in test runs could not justify the increased run time.

As we note in Paper II with reference to Hetland and Sætrom (2003a), a single PMC is three orders of magnitude faster for pattern matching purposes than comparable algorithms. Our cluster of machines, as described in Section 2.2, has about five hundred PMCs, which boosts the performance by another two orders of magnitude. Our current implementation of pattern evolution does not permit us to take full advantage of our PMC resources, as the overhead associated with fitness calculation, recombination, and mutation (cf. Section 2.5), in addition to the parsing and mapping of expressions from the internal representation to PMC configurations, is substantial. One way to avoid this is to use IO processors on the search cards, as was described as an alternative solution in Paper II. It should be noted, however, that performance has not been a limiting problem in any of the problems we have considered so far.

2.8 Statistical data mining

Machine learning is an active research field, and GP is only one of several algorithms. Our motivation for choosing GP is due to our supercomputing preferences, as we wanted to take advantage of the PMC's capacity. Other methods, such as genetic algorithms, decision trees, hidden markov models, and neural networks are also popular. Baldi and Brunak (2001) presents an excellent introduction to machine learning algorithms and their application in bioinformatics.

Statistical machine learning in general, and support vector machines (SVMs) in particular, has become enormously popular in recent years. For this reason, we have benchmarked our algorithm's performance against SVMs in Sætrom (2004). I will now outline the main ideas behind SVMs, but will limit the treatment to a two dimensional example in an effort to provide both informaticists and biologists with an informal introduction.

Imagine that you should construct a linear classifier $x_2 = cx_1 + d$ that separates between points in two dimensional space. Figure 2.3 (a) shows three of infinitely many lines that can be drawn between the groups of black and white circles. Intuitively, it makes sense to choose a line with some distance to all existing points. It seems safer with a line that has some margin on its classifications. An SVM classifier aims to construct the line with the largest possible classification margin, as illustrated in figure 2.3 (b). The infamous support vectors correspond to the circles that lie on the tangent of the supporting lines, which are also called classification boundaries. We label the *white* (1) and *black* (-1) circles, and express the

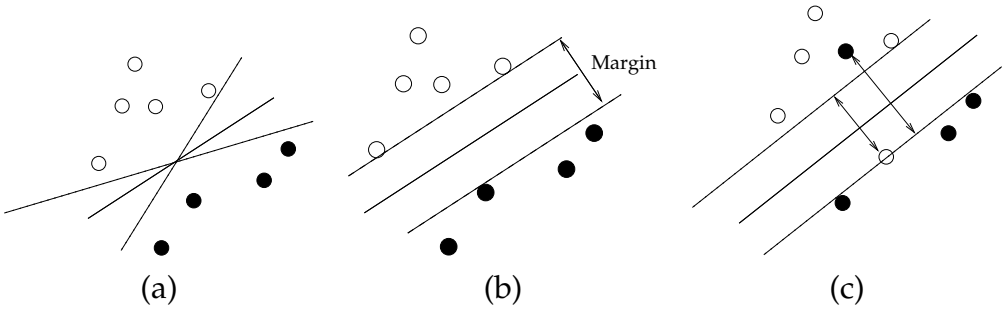

Figure 2.3: Support vector machine classification in two dimensions. As illustrated in (a), many classifiers may be optimal, but SVMs maximize the classification margin in the separable case (b), and, in addition, minimize the distance to any classification errors in the inseparable case (c).

supporting lines as $a_1x_1 + a_2x_2 + b = 1$ and $a_1x_1 + a_2x_2 + b = -1$, respectively, and the classifier becomes

$$\text{IF } a_1x_1 + a_2x_2 + b \begin{cases} \geq 1 & \text{THEN } \textit{white} \\ \leq -1 & \text{THEN } \textit{black} \end{cases} .$$

A problem occurs when the circles cannot be separated by a straight line, but this can be solved by choosing the line that maximizes the margin while minimizing the sum of the distances from the erroneously classified circles to their respective classification boundaries, as illustrated in Figure 2.3 (c). Our presentation lacks generality as we opted for coordinates in two dimensions to avoid a rigorous mathematical notation, but the ideas carry directly over to input vectors in n dimensions (Burges 1998, section 4).

We shall not dwell on the details here, but mention that finding the line—or hyperplane in the general case—can be done by numerically solving an optimization problem (Burges 1998). Statistical learning theory provides a mathematical foundation for SVMs, and bounds on the classification error can be computed (Vapnik 1998). It can be shown that non-linear SVMs can be obtained by performing a mapping from the input vectors to a space in higher dimension, and perform a linear classification there that is identical to a non-linear classification in the input space (Schölkopf 1997, section 2.1.4). Note that one such mapping results in an SVM that is identical to a neural network. A nice property of SVMs is that, in contrast with neural networks, an SVM always obtain the best classifier that exist

for a particular problem provided that its distinguishing characteristics are represented in the input vectors (Burges 1998).

In our benchmark experiments, we have used the ν-SVM algorithm of Schölkopf et al. (2000) that was modified to use the same k-fold cross validation procedure as we use with the boosted GP system (cf. Section 2.7). Note that there is a connection between boosting algorithms and SVMs (Müller et al. 2001). Boosting algorithms operate directly on the space spanned by all hypotheses defined by the weak learner, whereas SVMs use kernels to find optimal solutions in a high-dimensional space representation of the input space. Provided that the base learner of boosting algorithms explore the space of relevant hypotheses, these algorithms can be more efficient than SVM alternatives. When we obtain pattern-based classifiers that perform better than do SVM classifiers, this is most likely because our weak learner operates directly on the sequence with relevant pattern hypotheses, whereas the SVMs rely on potentially suboptimal vector representations of the sequences (see for instance Sætrom 2004).

Chapter 3

Non-coding RNA

Ribonucleic acid (RNA) seems to play an important role in many mechanisms that were unknown only a few years ago. It has been more than fifty years since Watson and Crick suggested the DNA structure that later turned out to be correct, but there are still work for new generations of biologists and informaticists. Biro (2004) recently proposed seven fundamental, but still unsolved questions in molecular biology, and understanding the role of non-protein-coding RNA (ncRNA)—a superclass of molecules that seems to be involved in transcription, splicing, translation, and more—will be important in that regard.

This chapter starts with an account about the factors that contribute to animal diversity, and proceeds with a discussion of ncRNA with emphasis on ncRNA gene discovery in *Escherichia coli*.

3.1 Central dogma of molecular biology

Deoxyribose nucleic acid (DNA) consists of two helical chains of nucleotides that are coiled around the same axis (Watson and Crick 1953). Each nucleotide along the chain consists of a deoxyribose sugar, a phosphate group, and a nitrogenous base. Deoxyribose is a pentose sugar, and the chain is constructed by linking the 5′ position of a given pentose ring to the 3′ position of the next pentose ring via a phosphate group. The sugar-phosphate backbones are linked together by the hydrogen bonds that form between the nitrogenous bases that in DNA are the purines adenine (A) and guanine (G), and the pyrimidines thymine (T) and cytosine (C). Adenine always binds to thymine, and are held together by the van der Waals forces that results from two hydrogen bonds, whereas guanine binds to cytosine with three hydrogen bonds; the ratios of A to T and G to C are

therefore both one in DNA. Both chains, or strands, are right-handed helices, but they run in opposite directions with respect to their phosphate linkages, and we therefore say that the strands are antiparallel. For simplicity, we often denote the nucleotides by the identity of their nitrogenous base. Ribonucleic acid (RNA) is different from DNA in that it is usually single-stranded, and contains ribose instead of deoxiribose, and uracil (U) instead of thymine. (See Lewin (2000a, chapter 1) for a detailed discussion on DNA and its properties.)

DNA's basepairing immediately provides a copy strategy for the cell: Separate the strands, and use one strand as a template to construct new complementary strands to obtain identical copies. Amazingly, this is exactly what happens when the cell divides and needs to duplicate its DNA (Alberts 2003). Enzymes that are called helicases unwind the double helix to provide single strands for another class of enzymes, namely the DNA polymerases that catalyze the formation of the complement nucleotides. The process of making an RNA copy of a stretch of DNA is called transcription and is performed by RNA polymerases. Transcription starts and ends at designated sites, and the product, which is called messenger RNA (mRNA), is an unstable intermediate that function as a template for protein factories called ribosomes that are located in the cytoplasm (Brenner et al. 1961). The ribosomes start at one end and continuously translate groups of three mRNA bases that each code for an amino acid that goes into an elongating polypeptide chain (Crick et al. 1961). In other words, DNA codes for RNA that codes for protein, which is the central dogma of genetic information transfer (Crick 1958).

3.2 Animal complexity

Perhaps surprisingly, animal diversity is not primarily due to a higher repertoire of genes in more complex species, but results from mechanisms for proteome expansion (Maniatis and Tasic 2002) and more elaborate gene regulation (Levine and Tjian 2003). When the drafts of the human genome were released, many were surprised to find that the public (Lander *et al.* 2001) and commercial (Venter et al. 2001) sequencing initiatives reported only about 22,000 and 26,000 genes, respectively. A subsequent publication of the finished sequence roughly confirms these numbers, and reports that there are probably between 20,000 and 25,000 protein-coding genes in the human genome (International Human Genome Sequencing Consortium 2004). In other words, our complexity over simpler species cannot be fully explained by a higher gene number. Not only do we have sur-

prisingly few protein-coding genes, but less than ten percent of our genes belong to gene families with other functions than those found in bacteria (Baltimore 2001).

There are at least three factors that contribute towards the increased complexity of higher organisms. First, gene rearrangements during lymphocyte differentiation, and subsequent somatic hypermutation, produce the diversity of mammalian immune systems (Nossal 2003). Second, alternative splicing enables production of different mRNA species from the same gene; for instance by skipping exons, retaining introns, or varying splice sites when the gene transcript is processed into an mRNA that is ready for protein translation (Ast 2004). Third, an elaborate regulation of gene expression must be an important factor, as between five and ten percent of protein-coding genes in metazoans are involved in transcription regulation (Levine and Tjian 2003). In addition to polymerases that catalyze transcription; protein complexes that bind specifically to DNA elements; and proteins that modify chromatin in order to regulate access to the template, there is also a large class of non-protein-coding genes that play key roles in generating diversity. These genes produce different kinds of ncRNA that be the subject of the remaining part of this thesis.

3.3 Many families of ncRNA

RNA that does not code for protein is not a recent discovery, and several species have been known for a long time. For example, ribosomal RNA (rRNA) is the fundamental structural element of the ribosomes, and transfer RNA (tRNA) mediates the growth of polypeptide chains guiding the right amino acids in place based on information from the mRNA (Lewin 2000b, chapter 6). Other ncRNAs were discovered already in the sixties (Storz 2002), but their importance has not been realized until recently. The discovery of catalytic RNA, or ribozymes, in the eighties (Guerrier-Takada et al. 1983) resulted in *RNA world* theories that hypothesize that life depended only on RNA initially (Gilbert 1986). An ancient RNA organism would require RNA to self-replicate and metabolize, and the theory also depends on RNA's ability to catalyze proteins to provide the transition into life as we know it. Bartel and Unrau (1999) notes that even though partial evidence exists, no ribozymes have been shown to have the capabilities that would be required to fulfill the RNA world hypothesis.

A plethora of ncRNAs have been discovered in recent years (Eddy 2001). An interesting observation is that the ratio of protein-coding RNA to the total transcriptional output decreases with an organism's develop-

Table 3.1: Non-coding RNAs belong to different sub-groups with diverse functions. Some different names have been and are still in use for the same molecules. For example, functional RNA (fRNA) is synonymous for ncRNA; small non-mRNA (snmRNA) denotes small ncRNAs; the stRNAs *lin-4* and *let-7* are the founding members of the miRNA class; and tmRNA was previously referred to as 10S RNA. The table was compiled from similar tables in the articles that are cited throughout this section.

Abbreviation and full name		Nucleotides	Major function
ncRNA	non-coding RNA		no protein encoding, but still functional
rRNA	ribosomal RNA	130 - 3000	part of ribosomes
tRNA	transfer RNA	70 - 80	protein translation
snRNA	small nuclear RNA	65 - 400	mRNA maturation, including slicing
snoRNA	small nucleolar RNA	60 - 550	rRNA modification
siRNA	short interfering RNA	21 - 28	mRNA depletion in RNAi
stRNA	small temporal RNA	21 - 22	larval development in *C. elegans*
miRNA	microRNA	19 - 29	mRNA depletion, translational suppression
tmRNA	transfer-messenger RNA	300 - 400	both tRNA and mRNA properties
gRNA	guide RNA	40 - 80	RNA editing

mental complexity (Mattick 2004). This may indicate that ncRNA has been instrumental in the evolutionary process towards increasingly complex organisms rather than being just ancient relics of ribo-organisms (Jeffares et al. 1998).

Morey and Avner (2004) divide ncRNAs into two groups based on their function. Housekeeping RNA includes small nuclear RNA (snRNA) and small nucleolar RNA (snoRNA) in addition to the more familiar rRNA and tRNA. Regulatory RNA is predominantly microRNA (miRNA), but also longer ncRNAs with roles in for instance the establishment of chromatin structures. Table 3.1 gives an overview of the different ncRNAs. Non-coding RNAs have diverse functions, and are in one way or the other involved in such different mechanisms as transcription; silencing; replication; RNA processing, modification, and stability; and protein translation, stability, and translocation (Storz 2002). A recent database lists 109 ncRNA classes distributed to 26 cellular processes (Liu et al. 2005). Other classes of ncRNAs exist, but the research field is still in its infancy, and we must therefore expect that this table will be somewhat dynamic, at least in the immediate future.

3.4 Computational challenges

As mentioned in Section 3.3, with the exception of rRNA and tRNA, ncRNAs have gone largely undeteceted until recently. Eddy (2001) explains this with gene discovery approaches's bias towards traditional genes that produces mRNA that codes for protein, and points out that ncRNAs are hard targets even for recessive mutational screens in classical genetics. Non-coding RNA genes often show only a modest conservation of primary structure, have no open reading frames, and are processed less systematically, which makes them more difficult to find than protein-coding genes (Morey and Avner 2004). To illustrate, ncRNAs are transcribed from several different promoters, and some are not even independently transcribed, but are processed from introns of protein coding genes (Aravin et al. 2003). Moreover, it has been shown that antisense transcripts, which are very common in the human genome (Yelin et al. 2003), have the potential to affect the expression level of overlapping and therefore complementary mRNA (Røsok and Sioud 2004). It is therefore natural to hypothesize that some ncRNAs may also be processed from antisense transcripts, or from the double-stranded structures that could be formed from overlapping sense and antisense transcripts that seem common from preliminary analysis of expressed sequence tags data (V. Blikstad, personal communi-

cation).

Eddy (2002) lists identification of transcription units without open reading frames; statistical sequence content analysis; and comparative genome analysis as methods for ncRNA identification. Algorithms based on these principles have had some success, but their accuracy is limited due to the ncRNA characteristics mentioned previously. What is more, the existing algorithms for transcription unit identification are far from optimal. Algorithms that have been used to predict ncRNAs in *E. coli* also fall into the aforementioned categories (Argaman et al. 2001; Carter et al. 2001; Chen et al. 2002; Rivas et al. 2001; Wassarman et al. 2001). Note that Argaman et al. (2001) combine promoter and terminator identification with analysis of sequence conservation to improve the predictive power, whereas Carter et al. (2001) use machine learning to combine sequence composition, known motifs, and secondary structure stability.

In Paper IV, we presented an alternative method that uses our genetic programming-based machine learning system to construct classifiers that distinguish between intergenic regions and confirmed ncRNAs in *E. coli*. We ran Northern blots and primer extension assays to confirm the correctness of twelve out of the sixteen predictions that were tested. Three of the validated ncRNAs confirmed the predictions of others, whereas the remaining nine genes were novel predictions. Zhang et al. (2004) estimate the number of ncRNAs in *E. coli* to be between 118 and 260, but speculates that this may be an overestimation. We extended the list of predictions in Hershberg et al. (2003) with about 150 novel predictions, and hypothesized that Zhang et al. (2004) may actually underestimate the number of ncRNAs in *E. coli*, as it is unlikely that none of our untested predictions will represent actual ncRNAs (Paper IV). Note that we only analyzed intergenic regions, which leaves out the possibility of finding independently transcribed ncRNAs, for instance from the antisense strand of protein-coding genes. This possibility further strengthens the hypothesis that there may be more ncRNAs in *E. coli* than previously estimated.

Chapter 4

RNA interference

Science Magazine chose RNA interference (RNAi) as Breakthrough of the Year in 2002 (Couzin 2002), and many have compared the technology's importance with that of recombinant DNA, monoclonal antibodies, and the polymerase chain reaction. Triggered by short double-stranded RNAs (dsRNAs) called short interfering RNA (siRNA), RNAi was incredibly effective compared with alternative technologies for sequence-specific knockdown of mRNA. Short interfering RNAs are very similar to mature members of an endogenous class of ncRNAs with regulating properties, namely the microRNAs (cf. Chapter 5). In hindsight, we might wonder why it took so long before researchers identified this apparently very important pathway for post-transcriptional gene regulation. But time is relative, and I would like to remark that even though DNA was isolated from white blood cells as early as 1869, its molecular basis and three dimensional structure were not discovered until 1929 and 1953. (See `http://www.dna50.org` for an excellent timeline of genetics and genomics that was published on occasion of the 50th anniversary of Watson and Crick's discovery of DNA's double helix structure.) Considering the large number of important discoveries that has been made since 1953, I think it is safe to say that science moves faster than ever, and that phenomena even more important than RNAi probably still awaits discovery.

Several excellent reviews have been written on RNAi. Older reviews, such as those by Zamore (2001), McManus and Sharp (2002), Hannon (2002), and Dykxhoorn et al. (2003) are still relevant. For more recent accounts, see detailed treatments on the history (Mello and Conte Jr. 2004), molecular basis (Meister and Tuschl 2004), efficiency (Mittal 2004), and applications (Dorsett and Tuschl 2004; Hannon and Rossi 2004) of RNAi.

This chapter will outline the principles of RNAi, address issues regarding siRNAs's efficacy and specificity, and discuss RNAi's role both as a

functional genomics tool and a potentially important therapeutic. Our own contributions are mentioned and related to the work of others where this is appropriate.

4.1 A natural process

RNA interference (RNAi) is a process for sequence-specific depletion of mRNA that was discovered following introduction of double-stranded RNA (dsRNA) in *C. elegans* (Fire et al. 1998). Figure 4.1 shows the steps of the process. First, the RNase-III-type enzyme Dicer processes the dsRNA into shorter duplexes of about 21 nucleotides with 5′ phosphates and 2-nucleotide 3′ overhangs (Bernstein et al. 2001; Elbashir et al. 2001b; Zamore et al. 2000). Second, a ribonucleoprotein complex called the RNA-induced silencing complex (RISC) unwinds the duplex, incorporates one strand, and cleaves cytoplasmic mRNA with near-perfect complementarity to the absorbed strand (Hammond et al. 2000; Martinez and Tuschl 2004; Zeng and Cullen 2002). The cleavage site is in the middle of the complementary region, ten nucleotides away from the nucleotide that is paired with the 5′ end of the siRNA (Elbashir et al. 2001b). Single-stranded RNA can also function as silencing agents, but dsRNA is a much more potent silencing trigger (Fire et al. 1998).

In nematodes, the silencing effect is passed on both to other tissues (Fire et al. 1998) and to progeny (Grishok et al. 2000), but neither of these features of RNAi are present in mammals. RNAi is clearly robust, and many ways of administering dsRNA to the worm have been successful, including, for instance, injection (Fire et al. 1998), soaking (Tabara et al. 1998), and feeding on transgenic bacteria that express dsRNA (Timmons and Fire 1998). Sequence-specific gene silencing occurs in many species. Post-transcriptional gene silencing in plants (Baulcombe 1999), quelling in fungi (Romano and Macino 1992), and RNAi in flies (Elbashir et al. 2001c), nematodes (Fire et al. 1998) and mammals (Elbashir et al. 2001a) shows that sequence-specific gene silencing has been an evolutionary advantage.

4.2 Short interfering RNAs

Many of the enzymes that are needed for RNAi are conserved in several species, which hinted towards a widespread existence of the silencing pathway (Hammond et al. 2001). Many doubted, however, that RNAi could be applied as a functional genomics tool in mammals because of

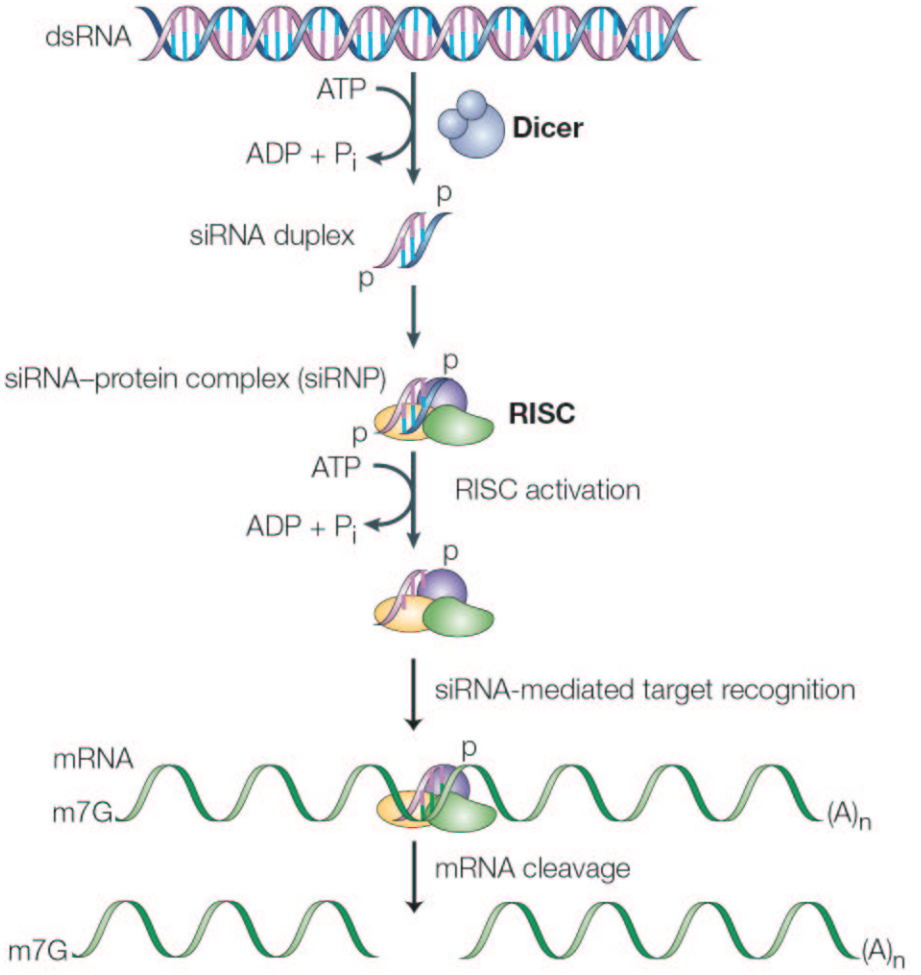

Figure 4.1: RNAi involves two main steps. Dicer cleaves dsRNA into shorter duplexes, and RISC incoporates one strand that provides the sequence-specificity of the subsequent target-recognition step that precedes mRNA cleavage. The figure is reprinted from Dykxhoorn et al. (2003) with permission from the authors and the journal.

an innate immune defense mechanism that protects the host from dsRNA threats such as viral infections (Manche et al. 1992). Initiated by dsRNA-dependent protein kinase (PKR) or 2′,5′-oligoadenylate synthetase, the defense mechanism induce non-specific cleavage of all RNA in addition to suppression of translation (Sledz and Williams 2004). Moreover, PKR also activates interferon, which in turn stimulates the production of several genes that make the cell sensitive to low levels of dsRNA and prone to apoptosis. Since dsRNA had been known to trigger this pathway for three decades, this was a considerable hurdle that had to be overcome before RNAi could be applied in humans (Hannon and Rossi 2004).

Following the realization that long dsRNAs are cut at 21 to 23 nucleotide intervals (Zamore et al. 2000), and that these short duplexes eventually mediate gene silencing (Elbashir et al. 2001b), it was shown that the interferon response could be circumvented by transfection of synthetic duplexes of approximately the same size in mammalian cells (Elbashir et al. 2001a). Short interfering RNAs are, however, still capable of activating the interferon pathway (Sledz et al. 2003), but the effect diminishes with lower siRNA concentrations (Persengiev et al. 2004).

Only a fraction of randomly selected siRNAs are effective at silencing their targets, and siRNAs whose targets are separated by only a few nucleotides may have very different silencing abilities (Holen et al. 2002). Both strands are capable of eliciting the effect, but RISC prefers to incorporate the strand with the lower thermodynamic stability at its 5′ end (Schwarz et al. 2003). Furthermore, there is some evidence that subsequent target cleavage may be more efficient if the binding between the absorbed strand and the target has a relatively low thermodynamic stability in the central region (Khvorova et al. 2003).

The next section discusses algorithms that improve the probability of obtaining siRNAs that have the potential of being effective silencing agents provided that they are efficiently transfected into the cell.

4.3 Rational design

Some siRNAs are remarkably effective silencing agents, whereas others are incapable of eliciting the effect (Elbashir et al. 2001c). Short interfering RNAs that are extremely effective can be used at lower concentrations (Reynolds et al. 2004), and are therefore less likely to introduce unwanted side-effects (cf. Section 4.5). In addition to the obvious concern about the quality of silencing experiments, there is also the cost issue, as siRNAs are relatively expensive reagents. It has therefore been important to develop

algorithms that can find the targets for which the most effective siRNAs can be constructed.

The first rules that emerged for siRNA design, the so called Tuschl rules, suggested that siRNAs should target sites with a balanced GC-content that were distal to the start codon Elbashir et al. (2002). Insights into how RISC prefers to incorporate the strand with the lower thermodynamic stability at its 5′ end (Schwarz et al. 2003), and how the thermodynamic stability profile may play a role for silencing efficacy (Khvorova et al. 2003), were welcomed by researchers developing siRNA design algorithms. Reynolds et al. (2004) were the first to propose an algorithm for rational siRNA design, and based their scheme on results that indicated significant positive and negative correlations between the siRNAs's efficacies and certain bases in specific positions. Moreover, they also suggested that siRNAs with a high potential for self-interaction—that is, complementary bases that may hybridize to form a hairpin—should be avoided. Other algorithms that are based on similar sequence feature observations include those of Amarzguioui and Prydz (2004); Hsieh et al. (2004); Takasaki et al. (2004); Ui-Tei et al. (2004); and Chalk et al. (2004). (See Paper VI for details on the algorithms.)

When RNAi emerged, molecular biologists had twenty years of experience with other technologies for sequence-specific knockdown of mRNA, such as antisense oligonucleotides and ribozymes (Scherer and Rossi 2003). Target accessibility contributed towards the efficacy of these technologies, and many have therefore speculated that the secondary structure of the mRNA may be important to siRNA efficacy as well. Luo and Chang (2004) proposed an algorithm based on this feature, but there are many reports that state that there is no dependency on target accessibility (Yoshinari et al. 2004). One explanation for the differing results may be that algorithms for secondary structure predictions are not yet optimal (Krol et al. 2004), and their output should therefore be treated with caution.

Pancoska et al. (2004) speculate that the duplex' melting temperature and the target segment's uniqueness compared with other transcript determine siRNA efficacy. Unfortunately, and as noted in Paper VI, we were not able to reproduce their algorithm, which is why we do not make any comparisons with it in this work. We have also yet to see other groups benchmark the results of Pancoska et al. (2004).

Our group was the first to use machine learning to predict the efficacy of siRNAs (Sætrom 2004). Boosted GP proved to be a better approach for pattern mining in short RNA sequences than was support vector machines that are considered state of the art in supervised learning. As we showed in Paper VI, our algorithm for siRNA efficacy prediction compared favorably

with other algorithms. In addition to our algorithm, only Amarzguioui and Prydz (2004); Ui-Tei et al. (2004); and Reynolds et al. (2004) had a stable and high performance across different datasets, which indicates that some algorithms have captured features of their training sets that are not generalizable to all siRNAs. Moreover, ensuring that the right strand gets into the RISC using difference in end stabilities (Schwarz et al. 2003) or duplex stability profiles (Khvorova et al. 2003) was not enough to match the performance of the best algorithms (Paper VI).

Algorithms for siRNA efficacy prediction capture the characteristics of effective siRNAs, but some have been trained on siRNA knockdowns obtained using different lab conditions, concentrations, and methods for relative mRNA knockdown measurement (see for instance the online supplementary material of Paper VI for details). As noted by Hannon and Rossi (2004), efficacy can be due to many factors, including transfection efficiency, siRNA concentration, and the individual siRNAs's efficacy. It is therefore natural to assume that some algorithms, including ours, have some potential for improved performance if presented with unbiased training sets with siRNAs evaluated under the same laboratory conditions and with the same protocols.

4.4 Short hairpin RNAs

Synthetic siRNAs are designed to mimic the duplexes that result from enzymatic processing of longer dsRNA by Dicer. Early observations of the microRNAs *let-7* and *lin-4*, whose mature structure resembles that of siRNAs, suggested that these were processed by Dicer from hairpin RNA precursors (Lee and Ambros 2001). Several groups used this observation to develop constructs that allow stable transfection of short hairpin RNAs (shRNAs) that give enduring silencing from siRNAs that result from Dicer processing. Paddison et al. (2002b) showed that endogenous transcription of long dsRNA of about 500 nucleotides resulted in sequence-specific silencing in cells with an inactive interferon response. As long dsRNA results in non-specific depletion of mRNA in cells with an intact response (Manche et al. 1992), this expression vector was of limited value. The problem was later resolved when Paddison et al. (2002a) used the U6 small nuclear RNA polymerase to express a shorter hairpin with a four nucleotide terminal loop and 3' overhang that more closely resembled the *let-7* microRNA. Short hairpin RNAs, like synthetic siRNAs, may induce the interferon response (Bridge et al. 2003), and the expression of hairpins from vectors must therefore be titrated to reduce the possibility for non-specific

silencing. Other vector expression systems have used alternative RNA polymerase III promoters or terminal loops that range from one through nine nucleotides (Brummelkamp et al. 2002; Sui et al. 2002; Yu et al. 2002). McManus et al. (2002) based their hairpin directly on the structure of a microRNA precursor, and showed that that the shRNA's structure was of major importance. Others have confirmed that the structure should probably resemble that of microRNA precursors for optimal efficacy (Boden et al. 2004; Miyagishi et al. 2004; Zeng et al. 2002). More details on siRNAs's similarities with mature microRNAs and shRNAs's similarities with microRNA precursors are given in chapter 5.

Kim et al. (2004) recently showed that the characteristic 5′ triphosphate of polymerase III transcripts is at least partly responsible for inducing the interferon response. Most small RNA genes are transcribed by polymerase III, but microRNAs differ and are mainly transcribed by polymerase II (Lee et al. 2004). Taken together, these observations may indicate that a polymerase II promoter may be a safer choice in shRNA expression systems. Denti et al. (2004) recently proposed an expression system that is based on the small nuclear U1 polymerase II promoter, but whether the system is able to shun the interferon response at higher concentrations than would be possible with a comparable polymerase III-based system is unknown. Note also that Zeng et al. (2002) used a polymerase II promoter to express natural miRNAs in human cells—a system that is reviewed in Zeng et al. (2005).

Stable *in vivo* expression is possible by using virus-mediated delivery and transcription. Indeed, both lentiviruses (Rubinson et al. 2003) and adenoviruses (Huang et al. 2004) have been used to express shRNAs. In principle, these delivery technologies make siRNAs viable in a gene therapy approach, but for reasons that will be discussed in Section 4.7, direct delivery of siRNAs is currently preferred in commercial RNAi therapeutics development.

Traditional shRNAs consist of three sequence segments corresponding to the reverse complement of the target, a spacer sequence, and the target plus two nucleotides that constitute the 3′ overhang. Kim et al. (2005) demonstrated that 27mer siRNAs are more effective at lower concentrations than conventional 21mers. They argue that these somewhat longer siRNAs are processed by Dicer, and thereby get the optimal properties required for efficient incorporation into RISC. Processing of a 27mer siRNA could potentially yield multiple cleavage products, but Kim et al. (2005) remarks that a 3′ overhang of two nucleotides added only to one of the strands guides Dicer to cleave avout 21 nucleotides, or about two helical turns, from this end. In light of these results, it is perhaps not surprising

that Siolas et al. (2004) report that shRNAs with a 29 basepair stem are more effective silencing agents than are shRNAs with a 19 basepair stem. Again, careful design of shRNAs based on common characteristics of microRNA precursors may contribute towards more effective shRNAs in the future.

4.5 Off-target risk

In addition to the problem with non-specific silencing by interferon stimulation, it is also possible to induce off-target silencing of transcripts with near-perfect complementarity to the siRNA. The first articles on siRNA reported excellent specificities, and alleles that differed only in a single nucleotide were specifically targeted (Elbashir et al. 2001c). Short interfering RNAs are indeed very specific, but there is no doubt that the initial reports on their performance in that regard were too optimistic.

There are several reports with conflicting results in the literature, but direct comparison is difficult due to differing cell lines, transfection procedures, concentrations, and measurement protocols. Table 4.1 shows results obtained in the same laboratory, but it is not possible to draw clear inferences about the mismatch tolerance from these observations. Other papers that have studied the position-specific mismatch tolerance of RNAi are also discordant (Jacque et al. 2002; Pusch et al. 2003; Vickers et al. 2003; Yu et al. 2002; Zeng and Cullen 2003). For example, Boutla et al. (2001) report that a central mismatch does not always abrogate silencing, as was proposed by Elbashir et al. (2001c). Whether or not a mismatch is tolerated also depends on the nature of the mutation. For example, guanosine to uracil mismatches—often referred to as G·U wobbles—are more readily tolerated than are other mismatches (Harborth et al. 2003; Saxena et al. 2003). A probable explanation for this is that G·U wobbles still possess a hydrogen bond, and are therefore anticipated to be more stable than regular mismatches according to thermodynamic criteria.

Several groups have used microarrays to check siRNAs's specificity by comparing the global gene expression before and after transfection. These reports do, however, disagree in their conclusions. Chi et al. (2003) and Semizarov et al. (2003) reported that siRNAs were highly specific, whereas Jackson et al. (2003) and Persengiev et al. (2004) saw expression profiles resulting from unspecific effects. In fact, Jackson et al. (2003) observed off-target effects when only a central core of eleven nucleotides were complementary to the target. Martinez and Tuschl (2004) recently confirmed that silencing is possible with only limited sequence complementarity, as

Table 4.1: Conflicting off-target observations in the same laboratory. Mismatch positions are counted from the 5′ end of the target, so that position nineteen equals the 5′-most position of the siRNA's antisense strand. Wildtype (WT) knockdown corresponds to the fraction of the target's mRNA that remained following transfection of an siRNA with perfect complementarity, and the negative integers represent the number of hydrogen bonds that are lost with the respective mismatches.

	WT	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
Holen et al. (2005)	20										60			20				20		
											-1									
Holen et al. (2005)	30	40			60							60		30	30			50	30	20
		-2			-2							-2		-2	-2			-2	-2	-2
Amarzguioui et al. (2003)	10	30	10	10	25			30			20	30		70	70					
		-3	-3	-3	-3			-3			-3	-3		-3	-3					
Holen et al. (2002)	20										30									
											-3									

thirteen basepairing nucleotides were enough to elicit silencing—four mismatches in positions 1 through 4 had only a marginal effect on silencing, whereas two mismatches in positions 17 and 19 resulted in about 3-fold reduction of cleavage (Martinez and Tuschl 2004). From these and other accounts (for example J. Canon's presentation at CHI's conference in Boston, MA on 9 November 2004), it seems likely that sequence-specific depletion of mRNA is possible even if there are some mismatches present, especially at the ends of the siRNAs.

The earliest siRNA design guidelines stated that researchers should use BLAST (Altschul et al. 1990) to check that their target did not have extensive sequence similarity with other transcripts (Elbashir et al. 2002). In Paper V, we showed that about 75 percent of commonly used siRNAs in the literature risked off-target activity if one accepts that three mismatches may carry a risk for silencing activity. Moreover, we also cautioned that BLAST is not appropriate when screening short oligomers for specificity, as the algorithm's sensitivity decreases with short oligomers and more mismatches. For example, BLAST will miss about six percent of all potential targets if three mismatches are allowed within a 21mer target (Paper V). Importantly, we argue that siRNAs should indeed be more specific, as experiments show that most transcripts will contain 21mers that are unique even if three mismatches are allowed.

It is widely accepted that RISC prefers to incorporate the strand with the lower 5′ end stability as reported by (Schwarz et al. 2003), but this should not be taken to imply that it is unnecessary to check if the other strand unintentionally targets other transcripts. Consider the possibility that strand selection by RISC may be more or less accurate, for instance when the difference in end stabilities is marginal. Some primary microRNA transcripts, including that of miR-30, give rise to two mature microRNAs (Lagos-Quintana et al. 2001; Mourelatos et al. 2002), which indicates that both strands may in some cases be active. Furthermore, and as noted previously, single-stranded RNA may also exploit the RNAi pathway (Holen et al. 2003), which means that there is a possibility that released strands can still become active silencing agents if they are not immediately degraded. It is important to take these uncertainties into consideration when analyzing the specificity of a given duplex, and we therefore think it is better to be on the safe side; consequently, we always check the specificity of both strands (Paper V; Paper VII).

An additional caveat to knockdown specificity is that siRNAs have been shown to function as microRNAs (Doench et al. 2003), and *vice versa* (Zeng et al. 2002). It is therefore possible that siRNAs can function as translational suppressors by targeting multiple partially complementary sites

in 3′ untranslated regions (see Section 5.4 for details on sequence requirements). Scacheri et al. (2004) recently provided an example of siRNAs that were assumed to suppress protein translation, as the siRNAs appeared to alter the protein expression of genes that were unrelated to the target even though the respective mRNA levels were unchanged. We recently developed an algorithm for microRNA target prediction (Paper VIII) that will be added to our siRNA design platform (Paper VII) to ensure specificity even if siRNAs function as microRNAs.

4.6 A tool in functional genomics

A quick search at PubMed Central shows that either of the medical subject heading terms for RNA interference, short interfering RNA, or RNA-induced silencing complex appeared in 14, 286, 1180, and 1543 articles that were published in 2001, 2002, 2003, and 2004, respectively. This shows not only that RNAi has been adopted as a standard functional genomics tool for sequence-specific silencing of genes, but that the research community has accepted and taken advantage of the new technology remarkably fast.

Hannon and Rossi (2004) suggest four rules for a successful RNAi experiment. First, researchers should pay attention to the design of their siRNAs. Here, evaluating an siRNA's efficacy and specificity is of equal importance (Paper VII). Second, the same phenotypic effect should be observed with siRNAs targeting different sites. If the observed phenotype is caused by sequence-dependent off-target effects, this will most probably be revealed using two independent siRNAs, as they are unlikely to accidentally knock down the same set of targets (Paper V). Third, one should work at the lowest possible concentrations, as sequence-independent off-target effects—that is, non-specific degradation of mRNA through the interferon pathway—depend on the siRNA concentration (Bridge et al. 2003; Persengiev et al. 2004; Sledz et al. 2003). In that regard, it is important to take advantage of good design algorithms to obtain the most effective siRNAs (Paper VI). Finally, if a phenotype can be reverted by expression of a modified target that cannot be recognized by the siRNA, this is considered best practice when it comes to proving that a target is really responsible for an altered phenotype. This can for instance be done by introducing a cDNA mutation at the target site so that the full complementarity between the mRNA and the siRNA is destroyed.

RNAi was relatively fast accepted as a more potent functional genomics tool than is antisense oligonucleotides and various oligonucleotides with cathalytic effects, such as ribozymes. Several companies sell siRNAs re-

agents on a market that has been estimated to reach about 200 million US dollars by 2008, but RNAi therapeutics target a market that is much bigger, and so this is where investors aim for the big returns (Howard 2003).

4.7 RNAi-based therapeutics

The ability of siRNAs to knock down specific targets with relatively high efficacy and specificity makes siRNAs viable as therapeutics, but much progress has to be made before an RNAi therapeutic becomes available. Ryther et al. (2005) identify efficacy, specificity, and delivery as the three important obstacles to effective siRNA therapeutics. Some siRNAs are very effective, and this may also reduce unwanted side-effects, which is the reason why RNAi has attracted a lot of attention in the last years. Progress has been fast in this field, and companies have moved several compounds into pre-clinical testing, but Food and Drug Administration approval can probably not be expected for another five or ten years (Howard 2003).

Several groups have administered siRNAs *in vivo*. For example, intravenous injection of siRNA into the tail of mice protected them from fulminant hepatitis due to knockdown of an apoptosis gene involved in liver failure (Song et al. 2003). Moreover, the excessive formation of red blood vessels in the eye was significantly reduced when injecting siRNA to the eyes of mice (Reich et al. 2003). Other attempted approaches using local, direct administration include intranasal delivery to the lungs and electroporation of postimplantation embryos (Ryther et al. 2005).

Soutschek et al. (2004) recently showed that chemically stabilized, cholesterol conjugated siRNAs have markedly improved pharmacokinetic properties. Both stability and biodistribution were significantly better than for unmodified siRNAs, as the half life in serum was more than fifteen times longer, and only modified siRNA were detectable in tissue samples. Therapeutic silencing was observed following intravenous injections in mice, but even though the dose of Soutschek et al. (2004) was about forty times lower than that used by Song et al. (2003), the dose and dose regiments still need optimization to be acceptable for clinical studies. Assuming that the mice weigh about 20 g, similar administration for a 70 kg person would be three injections of about 0.7 liters with 3.5 g of siRNA on three consecutive days. Intravenous injections is possible at the volumes described, but the cost of administering grams of siRNA would be enormous (Rossi 2004).

As outlined in Section 4.4, transfection of siRNA precursors results in stable knockdown and an RNAi gene therapy approach is therefore possible. Viral vectors use the infectious properties of the virus to get into the

cell where an shRNA is produced instead of the toxic virions. Thomas et al. (2003) list two classes of viral vectors. First, oncoretrovirus or lentivirus-based vectors integrate with the host's chromatin and can maintain continuous transcription. Second, vectors based on adeno-associated virus, adenovirus, or herpes virus reside in the nucleus without integrating into cellular genomes, which results in a transient effect. The future of gene therapy is, however, somewhat unclear after the death of one patient that were treated with an adenovirus vector in 1999, and two others that were believed cured with a retrovirus vector in 2002 and 2003 (Thomas et al. 2003). Presumably because of the hurdles of gene therapy, companies involved in commercial development of RNAi therapeutics currently focus exclusively on strategies for direct delivery of synthetic siRNA. As shown on their webpages, neither Alnylam Pharmaceuticals (Cambridge, MA), Sirna Therapeutics (Boulder, CO), CytRx Corporation (Los Angeles, CA), nor Intradigm Corporation (Rockville, MD) have published plans to deploy RNAi with a gene therapy-based strategy.

Chapter 5

MicroRNAs

One meaning of the word *dogma* is, according to Merriam-Webster's dictionary, a point of view or tenet put forth as authoritative widthout adequate grounds. Following the establishment of the central dogma of genetic information transfer in molecular biology (see Section 3.1) in the late 1950s, Francis H.C. Crick said that "biologists should not deceive themselves with the thought that some new class of biological molecules, of comparable importance to proteins, remains to be discovered". Scientists were therefore surprised when an entirely new class of functional ncRNAs with a potentially huge role in gene regulation was discovered in 2001. Hints towards the existence of a conserved class of endogenous RNA regulators came from studies of developmental biology in *C. elegans* where some ncRNAs were shown to suppress translation of proteins that are important for normal development of the worm in its larval stages. The members of the new family of endogenous regulators were called microRNAs (miRNAs) and these have since been shown to account for about one percent of the total transcriptional output in mammals, and have the potential to regulate thousands of genes. In addition to translational suppression, miRNAs may also mediate cleavage of mRNA, which means that there is a link between endogenous miRNAs and the synthetic siRNAs that mediate RNAi (cf. Chapter 4).

Several reviews on the functions and implications of miRNAs in plants (Baulcombe 2004), flies and worms (Ambros 2003), and humans (Ambros 2004) exist. Notable review articles that discuss miRNA genes and targets, as well as their biogenesis and mechanism of action have also been published relatively recently by Bartel and Chen (2004), He and Hannon (2004), and Bartel (2004).

In the following, we will describe some characteristics of miRNAs, including genes, biogenesis, and targets. Special attention will be given to

the algorithms that have been used to predict gene numbers and targets, and our own contributions will be discussed in relation to other methods.

5.1 MicroRNA genes

Studies of larval development timing in *C. elegans* revealed that a gene known to control larval development encoded an ncRNA with antisense complementarity to the mRNA of another gene (Lee et al. 1993). The ncRNA was called *lin-4* and its role in controlling the protein expression of the other gene was quickly confirmed (Wightman et al. 1993). It was later shown that *lin-4* promotes transition from the first to the second larval stage by blocking the synthesis of the LIN-14 (Olsen and Ambros 1999) and LIN-28 (Moss et al. 1997) proteins. Seven years later, another ncRNA, *let-7*, was shown to be required for larval to adult transition and to function by the same mechanisms as *lin-4* (Reinhart et al. 2000). Orthologues of *let-7* were found in many species including vertebrates (Pasquinelli et al. 2000), and this prompted screens for other endogenous ncRNAs with regulatory properties.

Indeed, numerous ncRNAs with similar characteristics as *lin-4* and *let-7* were identified by RNA cloning in *C. elegans*, *D. melanogaster*, and *H. sapiens* (Lagos-Quintana et al. 2001; Lau et al. 2001; Lee and Ambros 2001). Members of the new family of ncRNAs were called miRNAs. Lagos-Quintana et al. (2001) had previously developed a directional cloning method to isolate siRNAs in fly embryo lysate, and used this method to identify 16 miRNAs in flies and 21 miRNAs in humans. The same method was used by Lau et al. (2001) who identified 55 miRNAs in worms, and reported several orthologues in both flies and humans. Similarly, Lee and Ambros (2001) used cDNA cloning and informatics to identify 15 miRNAs in worms, of which three had homologue sequences in other species. Intensified cloning in the laboratories of Thomas Tuschl, Victor Ambros, David Bartel, and others then led to the identification of numerous miRNAs in various species (Bartel 2004). The most recent release of the miRNA registry now lists 1,420 entries in twelve plant, metazoan and virus species (Griffiths-Jones 2004, release 5.1). Since *lin-4* and *let-7* function in specific stages in the worm's larval development they were originally called small temporal RNA (stRNA). Nowadays, they are usually referred to as miRNAs because of their status as the class' founding.

The expression patterns of miRNAs have not been resolved in detail, but there have been some reports on this recently. MicroRNA genes are normally not polyadenylated, but are still predominantly transcribed by

RNA polymerase II (Lee et al. 2004)—a property they share with other RNA genes such as small nuclear RNA (Steinmetz et al. 2001). About a quarter of all miRNAs are, however, processed from the introns of other genes (Bartel 2004), and it has also been reported that some miRNAs are processed from normal mRNAs (Cai et al. 2004). Intronic miRNAs immediately suggest a mechanism for autonomous regulation, and if capped mRNAs can function as miRNAs as well, one may speculate that some alternative splice forms have been preserved because of their function as miRNA regulators.

5.2 Gene prediction

An important point is that current cloning technologies for experimental identification of miRNAs are biased towards abundant gene products. There is substantial evidence that the expression of many miRNAs are confined to particular tissues or organs (see for instance Lagos-Quintana et al. 2002), or restricted to certain developmental stages (see for instance Ambros 2000). Consequently, gene prediction algorithms are needed to find miRNAs that are not easily found in experimental screens.

Traditional gene finding algorithms cannot be used to identify miRNA genes, and the challenges of ncRNA gene prediction that was outlined in Section 3.4 apply to miRNA gene prediction as well. MicroRNA gene prediction usually involves finding evolutionary conserved hairpin structures with certain basepairing properties found in known miRNA genes (see Section 5.3), and three popular algorithms are MiRscan (Lim et al. 2003b), MiRseeker (Lai et al. 2003), and Srnaloop (Grad et al. 2003). To verify the computational predictions, Lim et al. (2003b) validated 16 candidates and Grad et al. (2003) 14 candidates in the worm. Based on extrapolation from the MiRscan's sensitivity on a human reference set, Lim et al. (2003a) estimated an upper limit of about 250 miRNA genes in humans. The development of an improved version of MiRscan was greatly aided by the identification of a conserved motif about 200 basepairs upstream of independently transcribed miRNA hairpins (Ohler et al. 2004). Based on statistical inferences, they concluded that as few as 20 miRNA genes remain to be found in the worm. It should be noted, however, that these estimates are based on the assumption that miRNAs can be accurately represented by the genes we know so far. It is entirely possible that miRNAs with slightly different characteristics exist, or that some miRNAs may be less conserved between species, in which case there are probably more miRNA genes than has been estimated.

Based on our experiments with ncRNA gene prediction methods in bacteria, we speculate that miRNA gene finding approaches may also have a great potential for improvements (see Section 3.4 and Paper IV). Indeed, others have suggested methods to improve miRNA gene prediction algorithms. For example, identification of orthologues based on profiles that exploit similarities in both primary and secondary structure (Lambert et al. 2004) may enable higher sensitivity than is provided by looking at the sequence homology alone (Legendre et al. 2004). Methods with higher sensitivity combined with additional examples of miRNAs with slightly different characteristics have the potential to discover new families of miRNAs.

5.3 Biogenesis in four steps

The biogenesis of miRNAs involves four important steps that corresponds to (i) enzymatic processing of primary transcripts; (ii) transport of precursors from the nucleus to the cytoplasm; (iii) enzymatic processing of precursors to mature double-stranded miRNAs; and (iv) assembly of the effector complex for mRNA cleavage and translational suppression. Figure 5.1 illustrates the various steps of the biogenesis. Note that steps one and two take place predominantly in the nucleus, whereas the remaining steps occur in the cytoplasm (Lee et al. 2002). In the following, we will describe these steps in more detail and outline some implications for shRNA and siRNA design.

(i) *Primary miRNA processing by Drosha.*

A ribonuclease (RNase) III known as Drosha cuts a relatively long primary miRNA (pri-miRNA) transcript into a miRNA precursor (pre-miRNA; Lee et al. 2003). Primary miRNA transcripts are characterized by a double-stranded stem and a hairpin loop, sometimes referred to as a fallback structure, that Drosha cuts about 22 nucleotides from the loop. Drosha cuts the stem in a staggered manner that is typical for RNase III enzymes, and leaves a precursor of about 70 nucleotides with a 3′ overhang of two nucleotides (Zeng et al. 2004). Drosha is part of a protein complex called the Microprocessor complex that comprise at least one polypeptide, DGCR8, that is necessary for pri-miRNA processing (Denli et al. 2004; Gregory et al. 2004). In the interest of clarity, we will identify the Microprocessor complex by the Drosha component in the remaining parts of this thesis, as this is still customary in the literature.

(ii) *Precursor transport from nucleus to cytoplasm.*

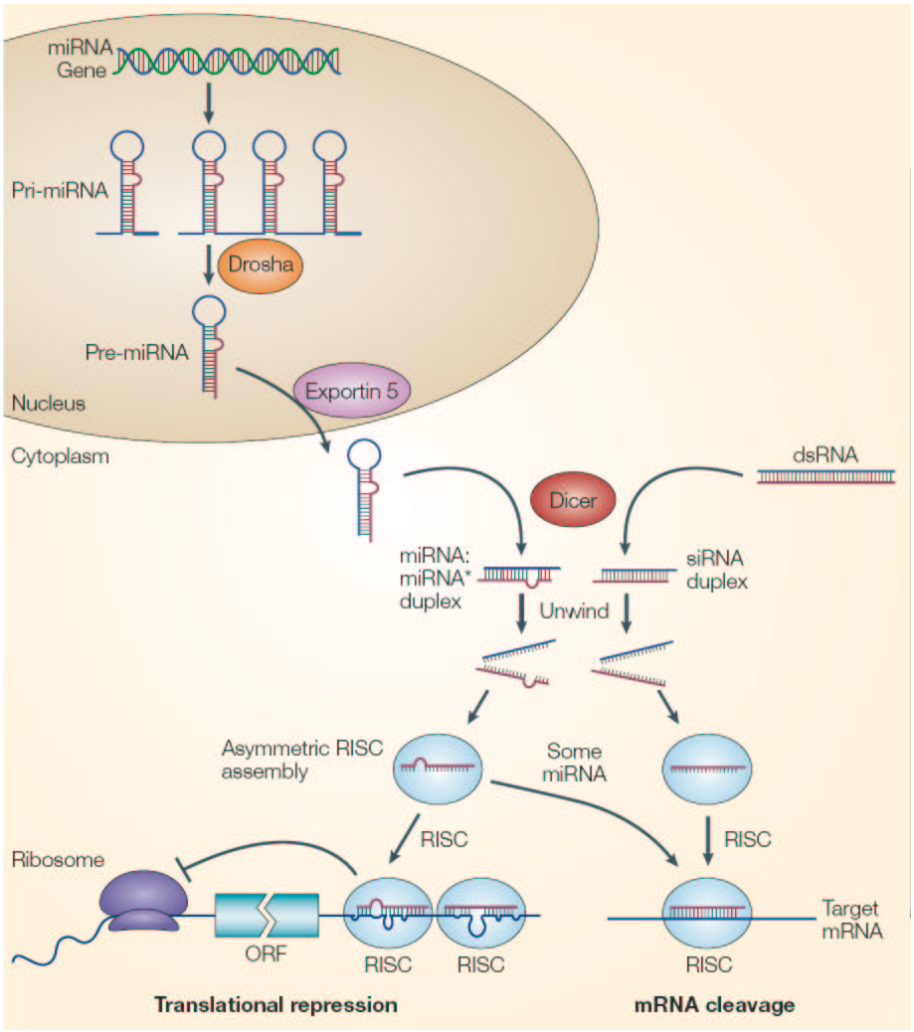

Figure 5.1: MicroRNAs mature in four main steps that include Drosha processing of pri-miRNAs into pre-miRNAs; Exportin-5-mediated transport of pre-miRNA to the cytoplasm; Dicer cutting of pre-miRNA into mature duplexes; and incorporation of one miRNA strand in the miRNP effector complex that most likely is identical to RISC. The figure is reprinted from He and Hannon (2004) with permission from the authors and the journal.

The precursors that results from Drosha processing are transported from the nucleus into the cytoplasm by the export receptor Exportin-5 (Yi et al. 2003). As was the case for Drosha, Exportin-5 is also part of a protein complex and depends on Ran guanosine triphosphate (Ran-GPT) to function properly (Bohnsack et al. 2004), as the pre-miRNAs are released from the complex following decreased expression of Ran-GPT in the cytoplasm (Lund et al. 2004). Zeng and Cullen (2004) have demonstrated that nuclear export of miR-30 precursors by Exportin-5 requires a stem of more than about 16 basepairs, and that a 3′ pre-miRNA overhang, as naturally resulting from Drosha processing, is clearly favored. They also argue that Exportin-5 not only mediates nuclear export, but also prevents pre-miRNAs from being degraded prior to cytoplasmic release.

(iii) *Precursor processing by Dicer.*

Another RNase III, Dicer, recognizes the pre-miRNA and cuts it about two helical turns from the end that results from Drosha processing (Hutvágner et al. 2001). Similar to the structure of siRNAs, the double-stranded miRNAs get characteristic 3′ overhangs of two nucleotides on both sides following the two-step processing by Drosha and Dicer (Bernstein et al. 2001; Ketting et al. 2001). In addition to its endonuclease activity, Dicer also has a helicase domain that may act together with other molecular factors in the subsequent step when the double-stranded miRNA is unwound and one of the strands incorporated into the effector complex (Murchison and Hannon 2004).

(iv) *Target recognition and downregulation by RISC.*

As previously mentioned, one of the strands of the miRNA is incorporated into the miRNA ribonucleoprotein complex (miRNP; Mourelatos et al. 2002), which is the active component of mRNA cleavage and translational repression. As is the case for siRNAs (cf. chapter 4), the strand with the lower 5′ end stability is selectively incorporated into miRNP, whereas the other strand is released and rapidly degraded (Schwarz et al. 2003). The miRNP may be identical to the RNA-induced silencing complex (RISC; Hutvágner and Zamore 2002), which is the active complex of RNAi by siRNAs (Nykänen et al. 2001). Again, as this is customary in the literature, we will identify the protein complex by the RISC acronym in the remaining parts of this thesis. RISC may mediate both mRNA cleavage and translational suppression, but it remains unclear whether this is due to the

degree of target complementarity only, or if this depends on molecular factors such as different versions of Dicer or RISC (Meister and Tuschl 2004).

It is possible, and even likely, that each of the four steps require molecular signatures made by the preceding step to function optimally. For example, Drosha may require certain characteristic primary or secondary structure features in the pri-miRNA transcript to provide efficient processing into pre-miRNAs. In turn, Exportin-5's efficiency may depend on pre-miRNA characteristics that may both be inherent in the primary transcript or resulting from Drosha processing, and so on. If we knew the features that are required for efficient biogenesis, this would likely aid the development of algorithms for rational shRNA and siRNA design, as these are similar to pre-miRNAs and mature double-stranded miRNAs, respectively.

Many groups have tried to determine which characteristics are important for biogenesis by performing mutagenesis of precursors and observing their relative expression in the cytoplasm. It seems that the pri-miRNA should be a basepaired extension of the pre-miRNA (Chen et al. 2004; Lund et al. 2004; Zeng and Cullen 2003), and the optimal size of the extension is likely between ten and twenty basepairs (Zeng et al. 2004). Analysis of the properties of miRNA genes confirmed that the nucleotides closest to the stem were highly conserved and that the degree of conservation decreased with the distance from the stem (Ohler et al. 2004). It also seems important that this double-stranded structure continues into the stem of the pre-miRNA (Lee et al. 2003), that the stem is more than about sixteen nucleotides long (Zeng and Cullen 2004), and that bulges and internal loops are relatively small (Lee et al. 2003; Zeng and Cullen 2003). Conversely, small hairpin loops seem to hinder efficient biogenesis (Zeng and Cullen 2003), and loops should therefore consist of more than ten nucleotides (Zeng et al. 2004). In that respect, it should be noted, however, that hairpin loops of three, five, and seven nucleotides functioned almost equally well in shRNA experiments targeting HIV-1 (Jacque et al. 2002), thus showing that results from just a few experiments are not necessarily definite. Furthermore, many of the inferences about efficient biogenesis depend on the correctness of several *in silico*-predicted secondary structures following mutagenesis. As noted by Zeng et al. (2004), some plasticity in the structure of miRNAs must be expected, as several possible folds with only a marginal difference in stability often results from miRNAs. Results from a recent study showed that eight of ten experimentally verified miRNA structures were different from their predicted counterparts (Krol et al. 2004), which further illustrates that secondary structure algorithms

are not guaranteed to be accurate. Further research is therefore necessary to reliably confirm the properties required for efficient biogenesis.

How the ribonucleases work and how they function in combination with other factors in larger complexes need to be resolved before we can properly understand the processes mediated by miRNAs and siRNAs. Drosha and Dicer are both RNase III enzymes, of which there are three functional classes (Carmell and Hannon 2004). All RNase IIIs contain a dsRNA substrate-binding domain (dsRBD), which give them their dsRNA specificity, in addition to at least one catalytic endonuclease domain with a conserved stretch of nine amino acids that is sometimes referred to as the RNase III signature motif. Bacterial RNase III enzymes are the simplest and belong to the first class that has been well characterized through studies in *E. coli* (Nicholson 1999). They contain a single dsRBD and one signature motif, as opposed to the second class that includes Drosha with its two RNase III signatures, one dsRBD, and a long N-terminal segment with unknown function (Filippov et al. 2000). The third class comprises Dicer which has two signatures, a dsRBD, a helicase domain, and at least two other domains (Bernstein et al. 2001). The dsRBDs are clearly important for RNase III enzymes's function in the miRNA pathway, but exactly how is not well known even though some theories exist. It is out of this thesis's scope to recapture even the current understanding of how the various protein complexes are composed and how they function, but the interested reader should confer recent reviews by Carmell and Hannon (2004), Meister and Tuschl (2004), and Murchison and Hannon (2004) to get a fairly updated overview of the biochemistry.

5.4 Target selection

Both *lin-4* and *let-7* target multiple sites in the 3′ UTR of their target mRNAs (Olsen and Ambros 1999). Indeed, multiple target sites on each 3′ UTR yield more potent translational suppression (Doench et al. 2003). In principle, however, it is possible that different miRNAs may induce a combinatorial effect by targeting multiple sites on the same 3′ UTR. Bartel and Chen (2004) suggested an electric circuit analogy to miRNA regulation: In exactly the same way as the total resistance of a rheostat with serially coupled resistors is calculated as the sum of resistances of the individual resistors, translational suppression may depend on the potency of a series of miRNAs that target multiple sites in the same 3′ UTR. It should be noted that Saxena et al. (2003) observed that neither a 3′ UTR location nor multiple target sites were necessary, but this seems to be the exception rather

than the rule.

The sequence requirements for blocking of protein synthesis are not known in detail, but it seems to be important that the first nine bases in the miRNA's 5′ end are largely complementary to the target region (Lewis et al. 2003), or at least that the binding energy of the miRNA·mRNA duplex is above a critical value (Doench and Sharp 2004). Brennecke et al. (2005) separate target sites in two classes: One of the classes comprise targets that have sufficiently strong binding between the target and the miRNA's 5′ end to not require additional complementarity in the miRNA's 3′ region. Conversely, miRNAs from the other class require binding between the target and the miRNA's 3′ region to compensate for lacking stability in the 5′ end.

Even though many miRNAs exist in humans, none of them have been shown to mediate translational repression; however, miRNAs may also function as siRNAs (Zeng et al. 2002) and silence the expression of complementary mRNAs by RNAi. Indeed, miR-196 was recently shown to direct cleavage of *Hoxb8* mRNA in mice embryos and cell culture (Yekta et al. 2004). Even though only a few targets have been accurately described, there are many reports on miRNAs with experimentally validated functions, such as for instance larval development in worms (Lee et al. 1993), cell proliferation in flies (Brennecke et al. 2003), and hematopoietic lineage differentiation in mice (Chen et al. 2004). Furthermore, the location of several miRNA genes to translocation breakpoints or deletions linked to leukemias (Calin et al. 2002) and the accumulation of miR-155 in B-cell lymphomas (Eis et al. 2005) may imply that miRNAs are involved in disease onset and development. Undoubtedly, the potential for miRNA regulation is large, as recent research has indicated that endogenously transcribed miRNAs may affect the protein expression of thousands of genes in humans (Lewis et al. 2005).

5.5 Target prediction

Even though miRNAs are able to mediate cleavage of mRNA with near-perfect complementarity to the guiding RNA strand, miRNA target prediction focuses on identification of potential targets for translational suppression. Several algorithms for target identification have emerged during the last years, but they are remarkably similar and usually use a seeding step to identify potential targets and several filtering steps to improve the predictions. Hence, the seeding step determines the algorithms's sensitivity while also affecting the specificity, whereas the remaining steps only

improve the specificity. There are two main types of seeding steps, namely one that uses sequence complementarity between the miRNA's 5′ region and the target, and another that demands that the same region has a thermodynamic stability above some critical value. Note that the two methods are indirectly the same, as duplex stabilities are calculated from the pairing of nucleotides in a double-stranded structure (see for instance Sugimoto et al. (1995) or Xia et al. (1998) for parameters). Furthermore, many of the algorithms use sequence complementarity in the seeding step and thermodynamic stability in a filtering step, and *vice versa*.

The TargetScan algorithm requires perfect Watson-Crick complementarity between 3′ UTR targets and nucleotides 2-8 in the miRNA's 5′ end (Lewis et al. 2003). The seed extensions are basepaired according to the optimal folding as determined by RNAfold (Hofacker 2003) and the UTRs are assigned a score depending on the number and stability of foldings. High-scoring candidates that are conserved between human, mouse, and rat become positive predictions. Stark et al. (2003) use the same principles as TargetScan, but seed matches between targets and nucleotides 1-8 of the miRNA allow G·U wobble basepairing, and favorable foldings are determined using mfold (Zuker 2003). As these algorithms use seed matches to obtain their candidates, miRNA targets that do not possess this property will be missed regardless of favorable duplex folding or sequence conservation. The miRanda algorithm is more flexible as it uses a position-specific scoring scheme that rewards 5′ complementarity and more hydrogen bonds, but the remaining duplex energy calculation and evolutionary conservation steps are similar to the other algorithms (Enright et al. 2003). The parameters of miRanda have been updated to reflect the binding properties of predicted human targets of virus-encoded miRNAs (Pfeffer et al. 2004) in addition to information about the influence of G·U wobbles (Doench and Sharp 2004), and was recently used to predict human miRNA targets (John et al. 2004). Rajewsky and Socci (2004) define a binding nucleus of consecutive basepairs, and calculate a weighted sum typically consisting of 6-8 addends favoring more hydrogen bonds. Note that the weights that was used differ only slightly from those of the miRanda algorithm (Enright et al. 2003). In a subsequent postprocessing step, Rajewsky and Socci (2004) use folding free energy as determined by mfold (Zuker 2003) to make the final predictions.

The RNAhybrid algorithm by Rehmsmeier et al. (2004) computes minimum free energy hybridization sites for miRNAs, while forcing perfect complementarity in nucleotides 2-7. Potential sites are normalized by the product of a miRNA and its potential target to avoid high-scoring but unlikely hybridizations to long target sequences. Extreme value statistics

similar to that used in sequence similarity searching is used to determine the likelihood of a candidate site being due to random hits in a large database (Rehmsmeier et al. 2004). Kiriakidou et al. (2004) combine computational and experimental methods and suggest that the nature of the complementarity is the most important characteristic of miRNA targets. Three types of interactions are outlined: a central bulge of 2-5 nucleotides on the mRNA; a central bulge of 6-9 nucleotides on the miRNA; and two opposing loops of 2-3 nucleotides on both sequences. In addition to thermodynamic duplex stability calculations and evolutionary conservation, the DIANA-microT algorithm uses the binding characteristics of Kiriakidou et al. (2004) to predict miRNA targets. It does not, however, require targets to consist of multiple adjacent binding sites.

In Paper VIII, we applied boosted GP to identify a model with several weighted patterns that identify potential target sequences with higher sensitivity than do comparable methods. Since additional filtering can be used to improve the specificity of any algorithm, we suggested that algorithms's performance should be compared before the application of such steps. Specifically, we compared our TargetBoost algorithm to various versions of Nucleus (Rajewsky and Socci 2004) and RNAhybrid (Rehmsmeier et al. 2004) whose seeding steps are based on thermodynamic stability and sequence complementarity, respectively. TargetBoost's performance compared favorably to both algorithms, and we therefore concluded that machine learning approaches such as GP have the potential to improve miRNA target prediction algorithms.

As previously mentioned, there are no known miRNA targets for translational repression in mammals. Algorithms for miRNA target prediction therefore depend on the assumption that mammalian targets can be accurately represented by worm and fruitfly targets. Smalheiser and Torvik (2004) compared the complementarity interactions between miRNAs and mRNA with those between miRNAs and scrambled controls. They find that the discriminative characteristics of putative targets are longer stretches of perfect complementarity; higher overall complementarity allowing for gaps, mismatches, and wobbles; and multiple proximal sites that are complementary to one or several miRNAs. Note that these results suggest that mammalian miRNA targets may possess other characteristics than do targets from *D. melanogaster*. Specifically, the stretches of perfect complementarity may be longer; targets in the protein coding region may be present; and the bias towards perfect complementarity in the miRNA's 5′ region may be weaker. This may indicate that targets from lower species do not accurately represent mammalian targets, and that we need to apply other techniques to find the first mammalian translational suppressors.

Chapter 6

Concluding remarks

Our initial purpose was to develop and document the performance of a special purpose search processor, namely the PMC. Following the development of screening and mining applications for this architecture, we realized that several scientifically important problems concerning the characteristics of ncRNA could be addressed using our approach. The present study was initiated as we realized that these problems could only be targeted if we scientifically validated all parts of our methodology.

This chapter will outline our contributions, and indicate some directions for future research.

6.1 New search processor in a cluster

One of our contributions is a MISD architecture—the PMC—for online pattern matching in data that cannot be efficiently processed by index-based search algorithms. The PMC represents one of the first MISD architectures ever to find practical applications. We provide a review of the architecture's features and performance compared with other solutions that have been published in the literature (Paper I). In addition to the technical solutions, an important part of our first paper is the mathematical formulation of hit functions that provides a framework to rigorously explain the chip's functionality. In Paper II, we explain how the pattern matching chip's MISD architecture was used to build a MIMD-type cluster that achieves near-linear scalability for chosen problems. The cluster's performance can be tailored to yield the appropriate ratio between query throughput and search speed depending on the problem's characteristics. Importantly, the cluster's versatility enable practical applications in many domains, but molecular biology has been our focus throughout this study.

6.2 High-performance screening

In Paper II, we identified some characteristics that are common for problems that can be appropriately addressed with our search cluster. First, the problem should be separable and independent, meaning that each subproblem can be solved in parallel with minimal overhead. Second, the problem should consist of a large number of queries that can be generated fast enough to avoid idle resources in the cluster. Third, the problem's data should be relatively static, as excessive dataset shuffling will significantly decrease the performance.

Interactive screening applications require immediate feedback to the user, but does not satisfy the second criteria. In Paper III, we demonstrated how interactive screening in nucleic acids or protein data could be realized, and the application provides an alternative to homology-based algorithms. Pattern-based applications such as the one in Paper III may not have a great future, but the design illustrates at least three principles that we think are important going forward. First, interactivity is crucial and enables researchers to improve their ideas based on their review of the results and their implications. We actually think that developers should consider the possibility of presenting the most important, though sometimes incomplete results before the full details are available. Second, users have to be able to put restrictions on their results so that they avoid excessive result reporting from uninteresting regions. Options to filter out some regions, such as repeats, are usually available with many search tools, but we think that the filter option holds great promise in order to improve current search tools. Third, direct comparison of results should be available in many applications, as illustrated by the histogram view in our application. A comparison of two inputs is often valuable, and we therefore think that research into alternative ways to present results is warranted.

In Paper V, we developed a screening application that satisfied all of the criteria that are characteristic for a problem that is ideally suited for our search cluster. Finding the most unique oligonucleotides of genes and transcriptomes are important for sequence-specific applications such as RNA interference. We found that most transcripts contain subsequences of about 20 nucleotides that can be uniquely targeted even if several mismatches are allowed. The paper presented an overview of the inherent uniqueness limitations for 19mers and 21mers, respectively. A minor but important contribution in Paper V was our demonstration of BLAST's limited sensitivity for short queries that consist of about 20 nucleotides and less. We think that molecular biologists and geneticists would benefit from learning the basic limitations of the algorithms they are using. In the same

manner as people are now *googling* the net, biologists are *blasting* the human genome, but one should be aware that neither of these verbs are synonymous to *searching*.

6.3 Complex pattern mining

In addition to the screening applications, we have developed a hardware-accelerated GP-based data mining platform. Complex patterns are randomly created in the first generation, and patterns are then bred in generation after generation with mutation, recombination, and selection until the top-performing patterns's ability to characterize groups of data is considered adequate. The system was first used for rule mining in time series data (cf. publications listed in Section 1.4). In this study, our contribution is the application of this system to various ncRNA sequences. The GP base learner has been adapted to use appropriate pattern architectures, boosting has been leveraged to increase the learning capacity, and regularization has been implemented to secure generalization.

Regularized boosting is computationally expensive, as described in Section 2.7. While others have opted for large-scale clusters of ordinary CPUs, we have used the relatively small PMC-based cluster of five accelerated workstations to achieve the desired performance. The small cluster is manageable and enables high-performance data mining on problems that contain relatively large volumes of data.

6.4 Motif-based ncRNA analysis

This work focuses on applications where classification of three types of ncRNAs has been involved. First, an ncRNA gene finding approach where false positive predictions of intergenic regions constitute predicted ncRNAs was described in Paper IV. Second, the efficacy of siRNAs in knockdown experiments was addressed in Paper VI, and an approach for rational design of both effective and specific siRNAs was proposed in Paper VII. Third, motif-based classification was used to capture the most important miRNA target binding features, and these were used to propose an improved seeding step for miRNA target prediction algorithms in Paper VIII.

Non-coding RNA genes differ from regular protein-coding genes, and traditional gene finding approaches do not perform well on ncRNAs. Several partly successful alternative approaches have been proposed, and our

main contribution in that regard is an alternative gene finding approach that performed well in validation experiments. In Paper VIII, we used hardware-accelerated boosted GP with regularization to separate between sequence windows of known ncRNA genes (positive examples) and intergenic regions (negative examples) in *E. coli*. Sixteen false positive predictions—that is, intergenic regions that were predicted to be ncRNAs—went into verification experiments in the laboratory. Twelve predictions were confirmed to be actual ncRNA genes, and nine of these were novel predictions. Based on many novel predictions and a relatively high confirmation rate in the experiments, we suggested that the number of ncRNAs in *E. coli* is actually higher than some previous studies had alluded.

Sætrom (2004) identified motif ensembles with a relatively high performance on predicting effective siRNAs for RNAi experiments. We subsequently showed that these classifiers generalized well to unseen data, and that they compared favorably to the other algorithms that were available (Paper VI). In addition, the results show that SVMs are not necessarily better than motif-based classification, and we attribute this to patterns's ability to operate directly on the solution, whereas SVMs depend on the sequences's vector representation. Our algorithm for rational design of siRNAs combines uniqueness screenings (Paper V) with efficacy predictions (Paper VI) to obtain siRNAs that are both effective and specific (Paper VII). Even the most unique siRNAs will carry some risk for off-target effects due to sequence similarity with mRNAs other than the intended target. With the complete off-target reports introduced in Paper VII, we provided researchers with the means to control that they chose siRNAs with the lowest risk of targeting genes that will affect the studied phenotype.

In Paper VIII, we modified the familiar boosted GP-approach to analyze the characteristics of known miRNA targets. Most algorithms use subsequent filtering based on thermodynamic features and sequence homology between species to increase the performance of their predictions. Importantly, however, the algorithms's sensitivities are determined by the initial seeding step, and we therefore compared all algorithms based on the initial output. These comparisons showed that the weighted motifs of the ensemble classifier performed better than did the other algorithms.

6.5 Alternatives for further work

This work contains articles on such different topics as for instance integrated circuit development and miRNA target analysis. There is no doubt that the study has been improved by the experience we have gained from disparate fields such as integrated circuits and molecular biology; however, we are certain that future research should be more focused. We believe that there are several interesting paths for future research, but realize that our small group cannot pursue all possibilities. There are two main directions that we may take; one is technology development with applications in other areas such as network surveillance and transactions monitoring, and the other is further analysis of characteristic properties of short regulatory RNAs. Either way, articles on the PMC and the GP-based data mining approach provide the foundation we need going forward. Moreover, work on applications in various fields have contributed several promising paths for future research. Both the screening and mining modules of our technology can be improved. Additions to the hardware—even completely new designs—are possible, and new functionality and application-specific adaptations to the GP system have also been discussed.

We have several ideas for new generations of the search chip that would have improved functionality and performance. In fact, Nedland (2000) proposed two new architectures that enable full regular expression functionality. We anticipate that research along these lines would be fruitful; however, we realize that the tremendous cost of ASICs will probably mean that future designs will be limited to FPGA implementations.

As for the existing PMC, the search card could easily be adapted to provide general-purpose functionality using ordinary CPUs instead of PMCs (Paper II). As noted in section 2.7, a more appropriate solution would in our case be to place a single IO processor on each card, as this would allow significantly improved cluster scalability when used in combination with GP-based data mining.

In our data mining efforts, we have strived for accuracy at the expense of intelligibility. This is adequate when we aim for the most appropriate predictions for subsequent laboratory validations. The main disadvantage of so-called black-box classifiers is that they provide limited knowledge about the mechanisms of action. For example, we cannot suggest candidate regulatory motifs that are typical of ncRNA genes even if we suspect that some contribute to the gene finding accuracy. We have some research underway that mends this problem and aim to decipher our ensemble models to obtain the most relevant motifs. Not only do we consider this improvement necessary for research that aim to address fundamen-

tal biology, but it may also make our existing solutions for gene finding, siRNA efficacy prediction, and miRNA target identification more appealing to biologists. Consequently, our collaborations with biologists may benefit greatly if we are able to produce readable expressions that provide some hints to which motifs that are functional.

Based on our experience with analysis of short RNA, we are eager to pursue many related problems. First, the ncRNA gene predictions in *E. coli* indicated some success for our approach, and we may therefore apply similar techniques to predict new miRNA genes in mammals. Second, we will do more research on effective siRNAs with the possible extension into effective shRNAs based on analysis of mature miRNAs and their precursors, respectively. Third, further development of the improved seeding step for miRNA target prediction into a complete alternative for target identification is underway. Fourth, identification of regulatory elements such as promoters, enhancers, and silencers is an opportunity that we consider due to the need for motif detection. Fifth, research on mechanisms for alternative splicing is also interesting for the same reasons. In any case, we expect that our method must be improved with several additional modules, including, but not limited to, thermodynamic analysis, structure conservation, and integration of information from various databases.

Glossary

This chapter provides readers with a small glossary that may be convenient considering that some specialized terms from the biologists's terminology may be unfamiliar to informaticists, and *vice versa*. The glossary does not aim to be complete, but should provide a short description of all abbreviations that are used throughout the thesis. Note that some of the entries have been adapted from dictionaries and glossaries from Merriam-Webster (Springfield, MA), the National Center for Biotechnology Information (Bethesda, MD), Microsoft (Redmond, WA), Invitrogen (Carlsbad, CA), the Technical University of Denmark (Kgs. Lyngby, Denmark), and Monster Isp (Mount Vernon, OH).

Application-specific integrated circuit (ASIC). As it is designed for a very specific purpose, ASICs contrast with more general-purpose devices such as memory chips or x86 processors that can be used in many different applications. ASICs are used in a number of specific applications, such as processors for controlling engines or chips on a motherboard chipsets. When produced in high volumes, ASICs have orders of magnitude higher cost-performance ratio than field-programmable gate arrays (FPGAs).

Basic local alignment tool (BLAST). A popular sequence comparison algorithm that is used to search for optimal local alignments between a sequence database and a pattern query. The BLAST algorithm is optimized for speed, and the initial seed search is done for a word of a specific length that scores at least some threshold when compared to the query using a substitution matrix. Word hits are then extended in either direction in an attempt to generate an alignment with optimal score. Note that when the text consists of nucleotides, practical implementations will require a perfect seed match between the word and the database.

Caenorhabditis elegans (C. elegans). A small hermaphroditic nematode that was developed as a model organism in the 1960s by Sydney

Brenner and colleagues. One important reason for using the worm in that it is possible to trace the cell lineage of every one of its approximately 1,000 constituent cells. C. elegans is used primarily to study the genetics of development and neurobiology.

Central processing unit (CPU). A general-purpose processing unit that performs the digital operations in a computer. The CPU is designed to run a group of instructions, or instruction set. CPU instructions can consist of adding and subtracting numbers, fetching information from memory, and other simple functions.

Complementary deoxyribonucleic acid (cDNA). A strand of DNA that is complementary to a given messenger ribonucleic acid (mRNA) and that serves as a template for production of the mRNA in the presence of an enzyme called reverse transcriptase.

Deoxyribonucleic acid (DNA). Any of various nucleic acids that are usually the molecular basis of heredity and localized especially in the cell's nucleus. DNA consists of two chains of alternate links between deoxyribose and phosphate that are held together by hydrogen bonds in a double helix configuration.

Double-stranded ribonucleic acid (dsRNA). Ribonucleic acid (RNA) duplex that is held together by hydrogen bonds in the same way as deoxyribonucleic acid (DNA). RNA is usually single-stranded, but may also have a double-stranded structure even though this configuration is much less stable than that of DNA.

Double-stranded ribonucleic acid substrate-binding domain (dsRBD). A protein domain with an affinity to double-stranded ribonucleic acid (dsRNA). Consequently, proteins with dsRBDs are prone to bind to dsRNAs.

Double-stranded ribonucleic acid-dependent protein kinase (PKR). This enzyme is involved in the interferon pathway, which is a mammalian host defence mechanism that results in non-specific destruction of messenger ribonucleic acid (mRNA) following introduction of double-stranded ribonucleic acid (dsRNA) of more than about 30 basepairs.

Drosophila melanogaster (D. melanogaster). Sometimes called fruit fly or just fly, it has been used as a genetic system since early in the 20th century because it lends itself easily to breeding experiments. In the 1980s, researchers began characterizing the genes that corresponded

with mutant phenotypes and discovered functional genes that have subsequently been identified in other species, including vertebrates.

Escherichia coli (E. coli). A widely used bacterium, and the simplest model organism. Studies in *E. coli* culminated in the 1950s with the discovery of deoxyribonucleic acid (DNA) as the genetic material and continued with the elucidation of the chemical details of replication and transcription. The genome of the widely used *E. coli* lab strain K12 was completely sequenced in September 1997.

Field-programmable gate array (FPGA). A microchip that may contain thousands of programmable logic gates. Good features of FPGAs include short development times, and FPGAs are often used for prototype or custom designs, including for example logic emulation. Applications that require high-volume production usually use application-specific integrated circuits (ASICs) instead.

Genetic programming (GP). A problem-solving algorithm that uses mutation and recombination to breed generations of computer programs that is intended to solve a certain problem. Compared with genetic algorithms that operate directly on bit strings, GP operates on computer programs with a some predefined architecture. In theory, the best computer programs improve with each generation, and the final solution approach the optimal solution.

Gigabytes (GB). A byte is a group of eight binary digits called bits, and a gigabyte is by definition 2^{30} or 1,073,741,824 bytes, as this is the power of 2 that is closest to one billion.

Input/output (IO). The complementary tasks of gathering and distributing data. Input is data that is aquired from a device or entered by the user through a device. Output is data that is sent to a device.

Interagon query language (IQL). A simple expression language that defines how queries are constructed using characters, strings, and string set operators. The IQL is the preferred query language for an application accessing the pattern matching chip (PMC), as the language's expressiveness corresponds closely to the available functionality of the chip architecture.

Megabytes (MB). A byte is a group of eight binary digits called bits, and a megabyte is by definition 2^{20} or 1,048,576 bytes, as this is the power of 2 that is closest to one million.

Messenger ribonucleic acid (mRNA). A ribonucleic acid (RNA) that carries the code for a particular protein from the nuclear deoxyribonucleic acid (DNA) to a ribosome in the cytoplasm and acts as a template for the formation of that protein.

Micro-ribonucleic acid (miRNA). Short endogenous double-stranded ribonucleic acids (dsRNA) of about 22 nucleotides with characteristic 3′ overhangs of two nucleotides. The overhangs result from processing of miRNA precursors by protein complexes containing the enzymes Drosha and Dicer. Double-stranded miRNA is unwound and one strand is incorporated into the micro-ribonucleic acid ribonucleoprotein complex (miRNP), which is the effector complex of translational suppression and cleavage of messenger ribonucleic acid (mRNA).

Micro-ribonucleic acid ribonucleoprotein complex (miRNP). The effector complex in translational suppression and messenger ribonucleic acid (mRNA) cleavage that may be mediated by micro-ribonucleic acids (miRNAs). The miRNP may be identical to the ribonucleic acid-induced silencing complex (RISC) that is the effector complex of ribonucleic acid interference (RNAi).

Multiple instruction stream - multiple data stream (MIMD). Is one of four categories in Flynn's taxonomy for classification of architectures along two axes, namely the number of instruction streams executing concurrently, and the number of data sets to which those instructions are being applied. A MIMD architecture is one where many instructions are concurrently applied to multiple data sets.

Multiple instruction stream - single data stream (MISD). Is one of four categories of Flynn's taxonomy for classification of architectures along two axes, namely the number of instruction streams executing concurrently, and the number of data sets to which those instructions are being applied. A MISD architecture is one where many instructions are concurrently applied to a single data set.

Non-coding RNA (ncRNA). Functional ribonucleic acid (RNA) that does not code for a protein. Several classes of ncRNA is described in section 3.3 of this thesis.

Pattern matching chip (PMC). A special-purpose search processor that matches complex regular expression-like queries using a multiple instruction stream - single data stream (MISD) hardware architecture.

Designed by researchers at the Norwegian University of Science and Technology (NTNU) in the nineties, and subsequently developed commercially by Fast Search & Transfer (FAST) and Interagon. The PMC's prototype was developed on a field-programmable gate array (FPGA), whereas the final version is an application-specific integrated circuit (ASIC).

Peripheral component interconnect (PCI). A specification for high-performance, 32-bit or 64-bit input/output (IO) buses. A PCI bus can be configured dynamically and is designed to be used by devices with high-bandwidth requirements.

Precursor of mature micro-ribonucleic acid (pre-miRNA). Micro-ribonucleic acid (miRNA) precursors of about 70 nucleotides with a characteristic 3' overhang of two nucleotides and a double-stranded stem whose strands are connected by nucleotides that form a hairpin loop. The pre-miRNA is processed from a primary miRNA by a protein complex that contains a ribonuclease called Drosha.

Primary micro-ribonucleic acid transcript (pri-miRNA). A relatively long ribonucleic acid (RNA) transcript with a characteristic fallback structure that is processed from the introns of a messenger ribonucleic acid (mRNA) or produced from an independent transcription unit. A pri-miRNA is the initial transcript in the biogenesis of a mature micro-ribonucleic acid (miRNA).

Ran guanosine triphosphate (Ran-GPT). A protein that has been shown to work in conjunction with the export receptor Exportin-5 to transport micro-ribonucleic acid (miRNA) precusors from the nucleus to the cytoplasm.

Receiver operating characteristic (ROC). A plot of a classifier's sensitivity for the range of specificties, or *vice versa*. A ROC score corresponds to the area under the curve, and is usually taken to represent a classifier's overall performance.

Ribonuclease (RNase). An enzyme that catalyzes the hydrolysis of ribonucleic acid (RNA), which is essentially the same as breaking it down.

Ribonucleic acid (RNA). Any of various nucleic acids that contain ribose and uracil as structural components, and are associated with the control of cellular chemical activities

Ribonucleic acid interference (RNAi). A natural process for sequence-specific depletion of messenger ribonucleic acid (mRNA). Double-stranded ribonucleic acid (dsRNA) is cut into short dsRNA with 3′ overhangs of two nucleotides that is subsequently unwound and incorporated into an endonucleolytic protein complex (see RISC) that degrade ribonucleic acid (RNA) with complementarity to its RNA component.

Ribonucleic acid-induced silencing complex (RISC). The multi-ribonucleoprotein complex that is the effector complex of ribonucleic acid interference (RNAi). RISC is proposed to bring the antisense strand of the short interfering ribonucleic acid (siRNA) and the cellular messenger ribonucleic acid (mRNA) together, and an endonucleolytic activity associated with the RISC cleaves the mRNA that is subsequently released and degraded.

Short hairpin ribonucleic acid (shRNA). Consists of sense and antisense sequneces from a target gene connected by a loop, and is expressed in mammalian cells from a vector by some promoter. The shRNA is then transported from the nucleus into the cytoplasm where an enzyme called Dicer processes it and leaves a short interfering ribonucleic acid-like (siRNA) molecule.

Short interfering ribonucleic acid (siRNA). A short double-stranded ribonucleic acid of about 21 nucleotides with 3′ overhangs of two nucleotides that mediates the ribonucleic acid interference (RNAi) response in mammalian cells.

Support vector machine (SVM). A generalized linear classifier that corresponds to the optimal classifier as defined by some maximum-margin criterion. The maximum-margin criterion provides regularization that helps the classifier to generalize better to unseen samples.

Untranslated region (UTR). A region of the messenger ribonucleic acid (mRNA) that is not translated into protein. There are UTRs on both ends of mRNAs, and these are commonly referred to as 5′ UTRs and 3′ UTRs.

Very-large-scale integration (VLSI). VLSI originally referred to chips with many tens of thousands transistors, as a natural successor to large-scale integration (LSI) chips that contain more than thousand transistors. There have been efforts to name various levels of integrations

above VLSI, but these are no longer in widespread use. Note that all microprocessors are VLSI or better.

Bibliography

Bruce Alberts. DNA replication and recombination. *Nature*, 421(6921): 431–435, 2003.

Stephen F. Altschul, Warren Gish, Webb Miller, Eugene W. Myers, and David J. Lipman. Basic local alignment search tool. *J. Mol. Biol.*, 215(3): 403–410, 1990.

Mohammed Amarzguioui, Torgeir Holen, Eshrat Babaie, and Hans Prydz. Tolerance for mutations and chemical modifications in a siRNA. *Nucleic Acids Res.*, 31(2):589–595, 2003.

Mohammed Amarzguioui and Hans Prydz. An algorithm for selection of functional siRNA sequences. *Biochem. Biophys. Res. Commun.*, 316(4): 1050–1058, 2004.

Victor Ambros. Control of developmental timing in Caenorhabditis elegans. *Curr. Opin. Genet. Dev.*, 10(4):428–433, 2000.

Victor Ambros. MicroRNA pathways in flies and worms: growth, death, fat, stress, and timing. *Cell*, 113(6):673–676, 2003.

Victor Ambros. The functions of animal microRNAs. *Nature*, 431(7006): 350–355, 2004.

Alexei A. Aravin, Mariana Lagos-Quintana, Abdullah Yalcin, Mihaela Zavolan, Debora Marks, Ben Snyder, Terry Gaasterland, Jutta Meyer, and Thomas Tuschl. The small RNA profile during Drosophila melanogaster development. *Dev. Cell*, 5(2):337–350, 2003.

Liron Argaman, Ruth Hershberg, Jörg Vogel, Gill Bejerano, E. Gerhart H. Wagner, Hanah Margalit, and Shoshy Altuvia. Novel small RNA-encoding genes in the intergenic regions of *Escherichia coli*. *Curr. Biol.*, 11 (12):941–950, 2001.

Gil Ast. How did alternative splicing evolve. *Nat. Rev. Genet.*, 5(10):773–782, 2004.

Pierre Baldi and Søren Brunak. *Bioinformatics: the machine learning approach*. The MIT Press, Cambridge, Massachusetts, 2nd edition, 2001.

Pierre Baldi, Søren Brunak, Yves Chauvin, Claus A.F. Andersen, and Henrik Nielsen. Assessing the accuracy of prediction algorithms for classification: an overview. *Bioinformatics*, 16(5):412–424, 2000.

David Baltimore. Our genome unveiled. *Nature*, 409(6822):814–816, 2001.

David P. Bartel. MicroRNAs: genomics, biogenesis, mechanism, and function. *Cell*, 116(2):281–297, 2004.

David P. Bartel and Chang-Zheng Chen. Micromanagers of gene expression: the potentially widespread influence of metazoan microRNAs. *Nat. Rev. Genet.*, 5(5):396–400, 2004.

David P. Bartel and Peter J. Unrau. Constructing an RNA world. *Trends Cell Biol.*, 9(12):M9–M13, 1999.

David Baulcombe. RNA silencing in plants. *Nature*, 431(7006):356–363, 2004.

David C. Baulcombe. Fast forward genetics based on virus-induced gene silencing. *Curr. Opin. Plant Biol.*, 2(2):109–113, 1999.

Emily Bernstein, Amy A. Caudy, Scott M. Hammond, and Gregory J. Hannon. Role for a bidentate ribonuclease in the initiation step of RNA interference. *Nature*, 409(6818):295–296, 2001.

Doron Betel and Christopher W.V. Hogue. Kangaroo - a pattern-matching program for biological sequences. *BMC Bioinformatics*, 3(1):20–22, 2002.

Olaf René Birkeland and Ola Snøve Jr. The pattern matching chip, 2002. Technical note, available upon request.

Olaf René Birkeland, Ola Snøve Jr., Arne Halaas, Ståle H. Fjeldstad, Magnar Nedland, Håkon Humberset, and Pål Sætrom. A MISD architecture in a pattern-mining supercomputing cluster. *IEEE Trans. on Comp.*, 2005. Submitted.

Jan Charles Biro. Seven fundamental, unsolved questions in molecular biology. cooperative storage and bi-directional transfer of biological information by nucleic acids and proteins: an alternative to the "central dogma". *Med. Hypotheses*, 63(6):951–962, 2004.

Daniel Boden, Oliver Pusch, Rebecca Silbermann, Fred Lee, Lynne Tucker, and Bharat Ramratnam. Enhanced gene silencing of HIV-1 specific siRNA using microRNA designed hairpins. *Nucleic Acids Res.*, 32(3): 1154–1158, 2004.

Markus T. Bohnsack, Kevin Czaplinski, and Dirk Görlich. Exportin 5 is a RanGTP-dependent dsRNA-binding protein that mediates nuclear export of pre-miRNAs. *RNA*, 10(2):185–191, 2004.

Alexandra Boutla, Christos Delidakis, Ioannis Livadaras, Mina Tsagris, and Martin Tabler. Short 5'-phosphorylated double-stranded RNAs induce RNA interference in drosophila. *Curr. Biol.*, 11(22):1776–1780, 2001.

Leo Breiman. Bagging predictors. *Mach. Learn.*, 24(2):123–140, 1996.

Julius Brennecke, David R. Hipfner, Alexander Stark, Robert B. Russell, and Stephen M. Cohen. *bantam* encodes a developmentally regulated miRNA that controls cell proliferation and regulates the proapoptotic gene *hid* in *Drosophila*. *Cell*, 113(1):25–36, 2003.

Julius Brennecke, Alexander Stark, Robert B. Russell, and Stephen M. Cohen. Principles of microRNA-target recognition. *PLoS Biology*, 3(3):e85, 2005.

Sydney Brenner, François Jacob, and Matthew Meselson. An unstable intermediate carrying information from genes to ribosomes for protein synthesis. *Nature*, 190:576–581, 1961.

Alan J. Bridge, Stephanie Pebernard, Annick Ducraux, Anne-Laure Nicoulaz, and Richard Iggo. Induction of an interferon response by RNAi vectors in mammalian cells. *Nat. Genet.*, 34(3):263–264, 2003.

Thijn R. Brummelkamp, René Bernards, and Reuven Agami. A system for stable expression of short interfering RNAs in mammalian cells. *Science*, 296(5567):550–553, 2002.

Christopher J.C. Burges. A tutorial on support vector machines for pattern recognition. *Knowledge Discovery and Data Mining*, 2(2):121–167, 1998.

Xuezhong Cai, Curt H. Hagedorn, and Bryan R. Cullen. Human microRNAs are processed from capped, polyadenylated transcripts that can also function as mRNAs. *RNA*, 10(12):1957–1966, 2004.

George Adrian Calin, Calin Dan Dumitry, Masayoshi Shimizu, Roberta Bichi, Simona Zupu, Evan Noch, Hansjuerg Aldler, Sashi Rattan, Michael Keating, Kanti Rai, Laura Rassenti, Thomas Kipps, Massimo Negrini, Florencia Bullrich, and Carlo M. Croce. Frequent deletions and down-regulations of micro-RNA genes miR15 and miR16 at 13q14 in chronic lymphotic leukemia. *Proc. Natl. Acad. Sci. U.S.A.*, 99(24):15524–15529, 2002.

Michelle A. Carmell and Gregory J. Hannon. RNase III enzymes and the initiation of gene silencing. *Nat. Struct. and Mol. Biol.*, 11(3):214–218, 2004.

Richard J. Carter, Inna Dubchak, and Stephen R. Holbrook. A computational approach to identify genes for functional RNAs in genomic sequences. *Nucleic Acids Res.*, 29(19):3928–3938, 2001.

Alistair M. Chalk, Claes Wahlestedt, and Erik L.L. Sonnhammer. Improved and automated prediction of effective siRNA. *Biochem. Biophys. Res. Commun.*, 319(1):264–274, 2004.

Chang-Zheng Chen, Ling Li, Harvey F. Lodish, and David P. Bartel. MicroRNAs modulate hematopoietic lineage differentiation. *Science*, 303 (5654):83–86, 2004.

Shuo Chen, Elena A. Lesnik, Thomas A. Hall, Rangarajan Sampath, Richard H. Griffey, Dave J. Ecker, and Lawrence B. Blyn. A bioinformatics based approach to discover small RNA genes in the *Escherichia coli* genome. *Biosystems*, 65(2-3):157–177, 2002.

Jen-Tsan Chi, Howard Y. Chang, Nancy N. Wang, Dustin S. Chang, Nina Dunphy, and Patrick O. Brown. Genomewide view of gene silencing by small interfering RNAs. *Proc. Natl. Acad. Sci. U.S.A.*, 100(11):6343–6346, 2003.

Jennifer Couzin. Breakthrough of the year. Small RNAs make a big splash. *Science*, 298(5602):2296–2297, 2002. News.

Francis H. C. Crick. The biological replication of macromolecules. *Symp. Soc. Exp. Biol.*, 12:138–163, 1958.

Francis H.C. Crick, Leslie Barnett, Sydney Brenner, and Richard J. Watts-Tobin. General nature of the genetic code for proteins. *Nature*, 192:1227–1232, 1961.

Ahmet M. Denli, Bastiaan B.J. Tops, Ronald H.A. Plasterk, René F. Ketting, and Gregory J. Hannon. Processing of primary microRNAs by the Microprocessor complex. *Nature*, 432(7014):231–235, 2004.

Michela Alessandra Denti, Alessandro Rosa, Olga Sthandier, Fernanda Gabriella De Angelis, and Irene Bozzoni. A new vector, based on the PolII promoter of the U1 snRNA gene, for the expression of sirnas in mammalian cells. *Mol. Ther.*, 10(1):191–199, 2004.

John G. Doench, Christian P. Petersen, and Phillip A. Sharp. siRNAs can function as miRNAs. *Genes Dev.*, 17(4):438–442, 2003.

John G. Doench and Phillip A. Sharp. Specificity of microRNA target selection in translational repression. *Genes Dev.*, 18(5):504–511, 2004.

Yair Dorsett and Thomas Tuschl. siRNAs: applications in functional genomics and potential as therapeutics. *Nat. Rev. Drug Discov.*, 3(4):318–329, 2004.

Derek M. Dykxhoorn, Carl D. Novina, and Phillip A. Sharp. Killing the messenger: short RNAs that silence gene expression. *Nat. Rev. Mol. Cell Biol.*, 4(6):457–467, 2003.

Sean R. Eddy. Non-coding RNA genes and the modern RNA world. *Nat. Rev. Genet.*, 2(12):919–929, Dec 2001.

Sean R. Eddy. Computational genomics of noncoding rna genes. *Cell*, 109 (2):137–140, 2002.

Peggy S. Eis, Wayne Tam, Liping Sun, Amy Chadburn, Zongdong Li, Mario F. Gomez, Elsebet Lund, and James E. Dahlberg. Accumulation of miR-155 and BIC RNA in human B cell lymphomas. *Proc. Natl. Acad. Sci. U.S.A.*, 102(10):3627–3632, 2005.

Sayda M. Elbashir, Jens Harborth, Winfried Lendeckel, Abdullah Yalcin, Klaus Weber, and Thomas Tuschl. Duplexes of 21-nucleotide RNAs mediate RNA interference in cultured mammalian cells. *Nature*, 411(6836): 494–498, 2001a.

Sayda M. Elbashir, Jens Harborth, Klaus Weber, and Thomas Tuschl. Analysis of gene function in somatic mammalian cells using small interfering RNAs. *Methods*, 26(2):199–213, 2002.

Sayda M. Elbashir, W. Lendeckel, and Thomas Tuschl. RNA interference is mediated by 21- and 22-nucleotide RNAs. *Genes Dev.*, 15(2):188–200, 2001b.

Sayda M. Elbashir, Javier Martinez, Agnieszka Patkaniowska, Winfried Lendeckel, and Thomas Tuschl. Functional anatomy of siRNAs for mediating efficient RNAi in drosophila melanogaster embryo lysates. *EMBO J.*, 20(23):6877–6888, 2001c.

Anton J. Enright, Bino John, Ulrike Gaul, Thomas Tuschl, Chris Sander, and Debora S. Marks. MicroRNA targets in *Drosophila*. *Genome Biol.*, 5(1): R1, 2003.

Fast Search & Transfer ASA. Digital processing device, 2000a. International publication number WO 00/22545.

Fast Search & Transfer ASA. A processing circuit and a search processor circuit, 2000b. International publication number WO 00/29981.

Valery Filippov, Victor Solovyev, Maria Fillipova, and Sarjeet S. Gill. A novel type of RNase III family proteins in eukaryotes. *Gene*, 245(1):213–221, 2000.

Andrew Fire, SiQun Xu, Mary K. Montgommery, Steven A. Kostas, Samuel E. Driver, and Craig C. Mello. Potent and specific genetic interference by double-stranded RNA in *Caenorhabditis elegans*. *Nature*, 391 (6593):806–811, 1998.

Yoav Freund. An adaptive version of the boost by majority algorithm. *Mach. Learn.*, 43(3):293–318, 2001.

Yoav Freund and Robert E. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. *J. Comput. System Sci.*, 55(1):119–139, Aug 1997.

Jeffrey E.F. Friedl. *Mastering Regular Expressions*. O'Reilly, Cambridge, MA, 2nd edition, 2002.

Walter Gilbert. The RNA world. *Nature*, 319(6055):618, 1986.

Yonatan Grad, John Aach, Gabriel D. Hayes, Brenda J. Reinhart, George M. Church, Gary Ruvkun, and John Kim. Computational and experimental identification of C. elegans microRNAs. *Mol. Cell.*, 11(5): 1253–1263, 2003.

Richard I. Gregory, Kai ping Yan, Govindasamy Amuthan, Rhimmaiah Chendrimada, Behzad Doratotaj, Neil Cooch, and Ramin Chiekhattar. The Microprocessor complex mediates the genesis of microRNAs. *Nature*, 432(7014):235–240, 2004.

Sam Griffiths-Jones. The microRNA registry. *Nucleic Acids Res.*, 32(90001): D109–111, 2004.

Alla Grishok, Hiroaki Tabara, and Craic C. Mello. Genetic requirements for inheritance of RNAi in C. elegans. *Science*, 287(5462):2494–2497, 2000.

Cecilia Guerrier-Takada, Katheleen Gardiner, Terry Marsh, Norman Pace, and Sidney Altman. The RNA moiety of ribonuclease P is the catalytic subunit of the enzyme. *Cell*, 35(3 Pt 2):849–857, 1983.

Dan Gusfield. *Algorithms on Strings, Trees, and Sequences : Computer science and computational biology*. Cambridge University Press, Cambridge, UK, 1997.

Arne Halaas, Børge Svingen, Magnar Nedland, Pål Sætrom, Ola Snøve Jr., and Olaf Renè Birkeland. A recursive MISD architecture for pattern matching. *IEEE Trans. on VLSI Syst.*, 12(7):727–734, 2004.

Scott M. Hammond, Emily Bernstein, David Beach, and Gregory J. Hannon. An RNA-directed nuclease mediates post-transcriptional gene silencing in drosophila cells. *Nature*, 404(6775):293–296, 2000.

Scott M. Hammond, Sabrina Boettcher, Amy A. Caudy, Ryuji Kobayashi, and Gregory J. Hannon. Argonaute2, a link between genetic and biochemical analyses of RNAi. *Science*, 293(5532):1146–1150, 2001.

Gregory J. Hannon. RNA interference. *Nature*, 418(6894):244–251, 2002.

Gregory J. Hannon and John J. Rossi. Unlocking the potential of the human genome with RNA interference. *Nature*, 431(7006):371–378, 2004.

Jens Harborth, Sayda M. Elbashir, Kim Vandenburgh, Heiko Manninga, Stephen A. Scaringe, Klaus Weber, and Thomas Tuschl. Sequence, chemical, and structural variation of small interfering RNAs and short hairpin

RNAs and the effect on mammalian gene silencing. *Antisense Nucleic Acid Drug Dev.*, 13:83–106, 2003.

Lin He and Gregory J. Hannon. MicroRNAs: small RNAs with a big role in gene regulation. *Nat. Rev. Genet.*, 5(7):522–531, 2004.

Ruth Hershberg, Shoshy Altuvia, and Hanah Margalit. A survey of small RNA-encoding genes in *Escherichia coli*. *Nucleic Acids Res.*, 31(7):1813–1820, 2003.

Magnus Lie Hetland and Pål Sætrom. Temporal rule discovery using genetic programming and specialized hardware. In *Proc. of the 4th Int. Conf. on Recent Advances in Soft Computing*, 2002.

Magnus Lie Hetland and Pål Sætrom. A comparison of hardware and software in sequence rule evolution. In *Eight Scandinavian Conference on Artificial Intelligence*, 2003a.

Magnus Lie Hetland and Pål Sætrom. The role of discretization parameters in sequence rule evolution. In *Proc. 7th Int. Conf. on Knowledge-Based Intelligent Information & Engineering Systems, KES*, 2003b.

Ivo L. Hofacker. Vienna RNA secondary structure server. *Nucleic Acids Res.*, 31(13):3429–3431, 2003.

Torgeir Holen, Mohammed Amarzguioui, Merete T. Wiiger, Eshrat Babaie, and Hans Prydz. Positional effects of short interfering RNAs targeting the human coagulation trigger Tissue Factor. *Nucleic Acids Res.*, 30 (8):1757–1766, 2002.

Torgeir Holen, Mohammed Amarzguioui Eshrat Babaie, and Hans Prydz. Similar behaviour of single-strand and double-strand siRNAs suggests they act through a common RNAi pathway. *Nucleic Acids Res.*, 31(9):2401–2407, 2003.

Torgeir Holen, Svein Erik Moe, Jan Gunnar Sørbø, Ole Petter Ottersen, and Arne Klungland. Tolerated wobble mutations in siRNAs decrease specificity, but can enhance activity in vivo. 2005. Manuscript.

Ken Howard. Unlocking the money-making potential of RNAi. *Nat. Biotechnol.*, 21(12):1441–1446, 2003.

Andrew C. Hsieh, Ronghai Bo, Judith Manola, Francisca Vazquez, Olivia Bare, Anastasia Khvorova, Stephen Scaringe, and William R. Sellers. A

library of siRNA duplexes targeting the phosphoinositide 3-kinase pathway: determinants of gene silencing for use in cell-based screens. *Nucleic Acids Res.*, 32(3):893–901, 2004.

Alan Huang, Yan Chen, Xinzhong Wang, Shanchuan Zhao, Nancy Su, and David W. White. Functional silencing of hepatic microsomal glucose-6-phosphatase gene expression in vivo by adenovirus-mediated delivery of short hairpin RNA. *FEBS Lett.*, 558(1–3):69–73, 2004.

Györgi Hutvágner and Phillip D. Zamore. A microRNA in a multiple-turnover RNAi enzyme complex. *Science*, 297(5589):2056–2060, 2002.

György Hutvágner, Juanita McLachlan, Amy E. Pasquinelli, Éva Bálint, Thomas Tuschl, and Phillip D. Zamore. A cellular function for the RNA-interference enzyme Dicer in the maturation of the let-7 small temporal RNA. *Science*, 293(5531):834–838, 2001.

Interagon AS. The Interagon query language : a reference guide. `http://www.interagon.com/pub/whitepapers/IQL.reference-latest.pdf`, sep 2002.

International Human Genome Sequencing Consortium. Finishing the euchromatic sequence of the human genome. *Nature*, 431(7011):931–945, 2004.

Aimee L. Jackson, Steven R. Bartz, Janell Schelter, Sumire V. Kobayashi, Julja Burchard, Mao Mao, Bin Li, Guy Cavet, and Peter S. Linsley. Expression profiling reveals off-target gene regulation by RNAi. *Nat. Biotechnol.*, 21(6):635–637, 2003.

Jean-Marc Jacque, Karine Triques, and Mario Stevenson. Modulation of HIV-1 replication by RNA interference. *Nature*, 418(6896):435–438, 2002.

Daniel C. Jeffares, Anthony M. Poole, and David Penny. Relics from the RNA world. *J. Mol. Evol.*, 46(1):18–36, 1998.

Bino John, Anton J. Enright, Alexei A. Aravin, Thomas Tuschl, Chris Sander, and Debora S. Marks. Human microRNA targets. *PLoS Biology*, 2 (11):e363, 2004.

Michael Kearns and Leslie Valiant. Cryptographic limitations on learning boolean formulae and finite automata. *J. ACM*, 41(1):67–95, 1994.

René F. Ketting, Sylvia E.J. Fischer, Emily Bernstein, Titia Sijen, Gregory J. Hannon, and Ronald H.A. Plasterk. Dicer functions in RNA interference and in synthesis of small RNA involved in developmental timing in C. elegans. *Genes Dev.*, 15(20):2654–2659, 2001.

Anastasia Khvorova, Angela Reynolds, and Sumedha D. Jayasena. Functional siRNAs and miRNAs exhibit strand bias. *Cell*, 115:209–216, 2003.

Dong-Ho Kim, Mark A. Behlke, Scott D. Rose, Mi-Sook Chang, Sangdun Choi, and John J. Rossi. Synthetic dsRNA Dicer substrates enhance RNAi potency and efficacy. *Nat. Biotechnol.*, page Epub ahead of print, 2005.

Dong-Ho Kim, Michael Longo, Young Han, Patric Lundberg, Edouard Cantin, and John J. Rossi. Interferon induction by siRNA and ssRNA synthesized by phage polymerase. *Nat. Biotechnol.*, 22(3):321–325, 2004.

Marianthi Kiriakidou, Peter T. Nelson, Andrei Kouranov, Perko Fitziev, Costas Bouyioukos, Zissimos Mourelatos, and Artemis Hatzigeorgiou. A combined computational-experimental approach predicts human microRNA targets. *Genes Dev.*, 18(10):1165–1178, 2004.

John R. Koza. *Genetic Programming: On the Programming of Computers by Natural Selection*. MIT Press, Cambridge Massachusetts, Dec 1992.

Jacek Krol, Krzysztof Sobczak, Urszula Wilczynska, Maria Drath, Anna Jasinska, Danuta Kaxzynska, and Wlodzimierz J. Krzyzosiak. Structural features of microRNA precursors and their relevance to miRNA biogenesis and siRNA/shRNA design. *J. Biol. Chem.*, 2004.

Mariana Lagos-Quintana, Reinhard Rauhut, Winfried Lendeckel, and Thomas Tuschl. Identification of novel genes coding for small expressed RNAs. *Science*, 294(5543):853–858, 2001.

Mariana Lagos-Quintana, Reinhard Rauhut, Abdullah Yalcin, Jutta Meyer, Winfried Lendeckel, and Thomas Tuschl. Identification of tissue-specific microRNAs from mouse. *Curr. Biol.*, 12(9):735–739, 2002.

Eric C. Lai, Pavel Tomancak, Robert W. Williams, and Gerald M. Rubin. Computational identification of *Drosophila* microRNA genes. *Genome Biol.*, 4(7):R42, 2003.

André Lambert, Jean-Fred Fontaine, Matthieu Legendre, Fabrice Leclerc, Emmanuelle Permal, Francois Major, Harald Putzer, Olivier Delfour, Bernard Michot, and Daniel Gautheret. The ERPIN server: an interface to

profile-based RNA motif identification. *Nucleic Acids Res.*, 32(Web Server issue):W160–165, 2004.

Eric S. Lander *et al.* Initial sequencing and analysis of the human genome. *Nature*, 409(6822):860–921, 2001.

Nelson C. Lau, Lee P. Lim, Earl G. Weinstein, and David P. Bartel. An abundant class of tiny RNAs with probable regulatory roles in *Caenorhabditis elegans*. *Science*, 294(5543):858–862, 2001.

Rosalind C. Lee and Victor Ambros. An extensive class of small RNAs in *Caenorhabditis elegans*. *Science*, 294(5543):862–864, 2001.

Rosalind C. Lee, R.L. Feinbaum, and Victor Ambros. The *C. elegans* heterochronic gene *lin-4* encodes small RNAs with antisense complementarity to *lin-14*. *Cell*, 75(5):843–854, 1993.

Yoontae Lee, Chiyoung Ahn, Jinju Han, Hyounjeong Choi, Jaekwang Kim, Jeongbin Yim, Junho Lee, Patrick Provost, Olof Rådmark, Sunyoung Kim, and V. Narry Kim. The nuclear RNase III Drosha initiates microRNA processing. *Nature*, 425(6956):415–419, 2003.

Yoontae Lee, Kipyoung Jeon, Jun-Tae Lee, Sunyoung Kim, and V. Narry Kim. MicroRNA maturation: stepwise processing and subcellular localization. *EMBO J.*, 21(17):4663–4670, 2002.

Yoontae Lee, Minju Kim, Jinju Han, Kyu-Hyun Yeom, Sanghyuk Lee, Sung Hee Baek, and V Narry Kim. MicroRNA genes are transcribed by RNA polymerase II. *EMBO J.*, 2004.

Matthieu Legendre, André Lambert, and Daniel Gautheret. Profile-based detection of microRNA precursors in animal genomes. *Bioinformatics*, 21 (7):841–845, 2004.

Michael Levine and Robert Tjian. Transcription regulation and animal diversity. *Nature*, 424(6945):147–151, 2003.

Benjamin Lewin. *Genes VII*. Oxford University Press, Oxford, UK, 2000a.

Benjamin Lewin. *Genes VII*. Oxford University Press, Oxford, UK, 2000b.

Benjamin P. Lewis, Christopher B. Burge, and David P. Bartel. Conserved seed pairing, often flanked by adenosines, indicates that thousands of human genes are microRNA targets. *Cell*, 120(1):15–20, 2005.

Benjamin P. Lewis, I hung Shih, Matthew W. Jones-Rhoades, David P. Bartel, and Christopher B. Burge. Prediction of mammalian microRNA targets. *Cell*, 115(7):787–798, 2003.

Lee P. Lim, Margaret E. Glasner, Soraya Yekta, Christopher B. Burge, and David P. Bartel. Vertebrate microRNA genes. *Science*, 299(5612):1540, 2003a.

Lee P. Lim, Nelson C. Lau, Earl G. Weinstein, Aliaa Abdelhakim, Soraya Yekta, Matthew W. Rhoades, Christopher B. Burge, and David P. Bartel. The microRNAs of *Caenorhabditis elegans*. *Genes Dev.*, 17(8):991–1008, 2003b.

Changning Liu, Baoyan Bai, Geir Skogerbø, Lun Cai, Wei Deng, Yong Zhang, Dongbo By, Yi Zhao, and Runsheng Chen. NONCODE: an integrated knowledge database of non-coding RNAs. *Nucleic Acids Res.*, 33 (Database Issue):D112–D115, 2005.

Elsebet Lund, Stephan Güttinger, Angelo Calado, James E. Dahlberg, and Ulrike Kutay. Nuclear export of microRNA precursors. *Science*, 303(5654): 95–98, 2004.

Kathy Q. Luo and Donald C. Chang. The gene-silencing efficiency of siRNA is strongly dependent on the local structure of mRNA at the targeted region. *Biochem. Biophys. Res. Commun.*, 318(1):303–310, 2004.

Lisa Manche, Simon R. Green, Christian Schmedt, and Michael B. Matthews. Interactions between double-stranded RNA regulators and the protein kinase DAI. *Mol. Cell.*, 12(11):5238–5248, 1992.

Tom Maniatis and Bosiljka Tasic. Alternative pre-mRNA splicing and proteome expansion in metazoans. *Nature*, 418(6894):236–243, 2002.

J. Kent Martin and Daniel S. Hirschberg. Small sample statistics for classification error rates I: Error rate measurements. Technical Report 96-21, ICS Dept., UC Irvine, 1996.

Javier Martinez and Thomas Tuschl. RISC is a 5′ phosphomonoester-producing rna endonuclease. *Genes Dev.*, 18(9):975–980, 2004.

John S. Mattick. RNA regulation: a new genetics? *Nat. Rev. Genet.*, 5(4): 316–323, 2004.

Michael T. McManus, Christian P. Petersen, Brian B. Haines, Jianzhu Chen, and Phillip A. Sharp. Gene silencing using micro-RNA designed hairpins. *RNA*, 8(6):842–850, 2002.

Michael T. McManus and Phillip A. Sharp. Gene silencing in mammals by small interfering RNAs. *Nat. Rev. Genet.*, 3(10):737–747, 2002.

Ron Meir and Gunnar Rätsch. An introduction to boosting and leveraging. In S. Mendelson and A. Smola, editors, *Advanced Lectures on Machine Learning*, volume 2600, pages 118–183. Springer-Verlag, 2003.

Gunter Meister and Thomas Tuschl. Mechanisms of gene silencing by double-stranded RNA. *Nature*, 431(7006):343–349, 2004.

Craig C. Mello and Darryl Conte Jr. Revealing the world of RNA interference. *Nature*, 431(7006):338–342, 2004.

Panagiotis D. Michailidis and Konstantinos G. Margaritis. On-line approximate string searching algorithms: Survey and experimental results. *International Journal of Computer Mathematics*, 79(8):867–888, 2002.

Vivek Mittal. Improving the efficiency of RNA interference in mammals. *Nat. Rev. Genet.*, 5(5):355–365, 2004.

Makoto Miyagishi, Hidetoshi Sumimoto, Hiroyuki Miyoshi, Yutaka Kawakami, and Kazunari Taira. Optimization of an siRNA-expression system with an improved hairpin and its significant suppressive effects in mammalian cells. *J. Gene Med.*, 6(7):715–723, 2004.

Céline Morey and Philip Avner. Employment opportunities for non-coding RNAs. *FEBS Lett.*, 567(1):27–34, 2004.

Eric G. Moss, Rosalind C. Lee, and Victor Ambros. The cold shock domain protein LIN-28 controls developmental timing in C. elegans and is regulated by the *lin-4* RNA. *Cell*, 88(5):637–646, 1997.

Zissimos Mourelatos, Josée Dostie, Sergey Paushkin, Anup Sharma, Bernard Charroux, Linda Abel, Juri Rappsilber, Matthias Mann, and Gideon Dreyfuss. miRNPs: a novel class of ribonucleoproteins containing numerous microRNAs. *Genes Dev.*, 16(6):720–728, 2002.

Klaus-Robert Müller, Sebastian Mika, Gunnar Rätsch, and Koji Tsuda. An introduction to kernel-based learning algorithms. *IEEE Trans. Neural Networks*, 12(2):181–201, 2001.

Elizabeth P. Murchison and Gregory J. Hannon. miRNAs on the move: miRNA biogenesis and the RNAi machinery. *Curr. Opin. Cell Biol.*, 16(3): 223–229, 2004.

Gonzalo Navarro. A guided tour to approximate string matching. *ACM Comput. Surv.*, 33(1):31–88, 2001.

Gonzalo Navarro and Mathieu Raffinot. *Flexible pattern matching in strings: practical on-line search algorithms for texts and biological sequences*. Cambridge University Press, Cambridge, UK, 2002.

Magnar Nedland. Design of a hardware regular expression matcher, 2000. Master thesis.

Allen W. Nicholson. Function, mechanism and regulation of bacterial ribonucleases. *FEMS Microbiol. Rev.*, 23(3):371–390, 1999.

Gustav J.V. Nossal. The double helix and immunology. *Nature*, 421(6921): 440–444, 2003.

Antti Nykänen, Benjamin Haley, and Phillip D. Zamore. ATP requirements and small interfering RNA structure in the RNA interference pathway. *Cell*, 107(3):309–321, 2001.

Uwe Ohler, Soraya Yekta, Lee P. Lim, David P. Bartel, and Christopher B. Burge. Patterns of flanking sequence conservation and a characteristic upstream motif for microRNA gene identification. *RNA*, 10(9):1309–1322, 2004.

Philip H. Olsen and Victor Ambros. The *lin-4* regulatory RNA controls developmental timing in *Caenorhabditis elegans* by blocking LIN-14 protein synthesis after the initiation of translation. *Dev. Biol.*, 216(2):671–680, 1999.

Patrick J. Paddison, Amy A. Caudy, Emily Bernstein, Gregory J. Hannon, and Douglas S. Conklin. Short hairpin RNAs (shRNAs) induce sequence-specific silencing in mammalian cells. *Genes Dev.*, 16(8):948–958, 2002a.

Patrick J. Paddison, Amy A. Caudy, and Gregory J. Hannon. Stable suppression of gene expression by RNAi in mammalian cells. *Proc. Natl. Acad. Sci. U.S.A.*, 99(3):1443–1448, 2002b.

Petr Pancoska, Zdenek Moravek, and Ute M. Moll. Efficient RNA interference depends on global context of the target sequence: quantitative analysis of silencing efficiency using Eulerian graph representation of siRNA. *Nucleic Acids Res.*, 32(4):1469–1479, 2004.

Amy E. Pasquinelli, Brenda J. Reinhart, Frank Slack, Mark Q. Martindale, Mitzi I. Kuroda, Betsy Maller, David C. Hayward, Eldon W. Ball, Bernard Degnan, Peter Müller, Jürg Spring, Ashok Srinivasan, Mark Fishman, John Finnerty, Joseph Corbo, Michael Levine, Patrick Leahy, Eric Davidson, and Gary Ruvkun. Conservation of the sequence and temporal expression of *let-7* heterochronic regulatory RNA. *Nature*, 408(6808): 86–89, 2000.

Stephan P. Persengiev, Xiaochun Zhu, and Michael R. Green. Nonspecific, concentration-dependent stimulation and repression of mammalian gene expression by small interfering RNAs. *RNA*, 10(1):12–18, 2004.

Sebastien Pfeffer, Mihaela Zavolan, Friedrich A. Grässer, Minchen Chien, James J. Russo, Jingyue Ju, Bino John, Anton J. Enright, Debora Marks, Chris Sander, and Thomas Tuschl. Identification of virus-encoded microRNAs. *Science*, 304(5671):734–736, 2004.

Oliver Pusch, Daniel Boden, Rebecca Silbermann, Fred Lee, Lynne Tucker, and Bharat Ramratnam. Nucleotide sequence homology requirements of HIV-1-specific short hairpin rna. *Nucleic Acids Res.*, 31(22):6444–6449, 2003.

Nikolaus Rajewsky and Nicholas D. Socci. Computational identification of microRNA targets. *Dev. Biol.*, 267(2):529–535, 2004.

Gunnar Rätsch, Takashi Onoda, and Klaus-Robert Müller. Soft margins for AdaBoost. *Mach. Learn.*, 42(3):287–320, Mar 2001.

Marc Rehmsmeier, Peter Steffen, Matthias Höchsmann, and Robert Giegerich. Fast and effective prediction of microRNA/target duplexes. *RNA*, 10(10):1507–1517, 2004.

Samuel J. Reich, Joshua Fosnot, Akiko Kuroki, Waizing Tang, Xiangyang Yang, Albert M. Maguire, Jean Bennett, and Michael J. Tolentino. Small interfering RNA targeting VEGF effectively inhibits neovascularization in a mouse model. *Mol. Vis.*, (9):210–216, 2003.

Brenda J. Reinhart, Frank J. Slack, Michael Basson, Amy E. Pasquinelli, Jill C. Bettinger, Ann E. Rougvie, H. Robert Horvitz, and Gary Ruvkun. The 21-nucleotide *let-7* RNA regulates developmental timing in *Caenorhabditis elegans*. *Nature*, 403(6772):901–906, 2000.

Angela Reynolds, Devin Leake, Queta Boese, Stephen Scaringe, William S. Marshall, and Anastasia Khvorova. Rational siRNA design for RNA interference. *Nat. Biotechnol.*, 22(3):326–330, 2004.

Elena Rivas, Robert J. Klein, Thomas A. Jones, and Sean R. Eddy. Computational identification of noncoding RNAs in *E. coli* by comparative genomics. *Curr. Biol.*, 11(17):1369–1373, 2001.

Nicoletta Romano and Giuseppe Macino. Quelling: transient inactivation of gene expression in neurospora crassa by transformation with homologous sequences. *Mol. Microbiol.*, 6(22):3343–3353, 1992.

Øystein Røsok and Mouldy Sioud. Systematic identification of sense-antisense transcripts in mammalian cells. *Nat. Biotechnol.*, 22(1):104–108, 2004.

John J. Rossi. Medicine: a cholesterol connection in RNAi, 2004. Comment.

Douglas A. Rubinson, Christopher P. Dillon, Adam V. Kwiatkowski, Claudia Sievers, Lili Yang, Johnny Kopinja, Mingdi Zhang, Michael T. McManus, Frank B. Gertler, Martin L. Scott, and Luk Van Parijs. A lentivirus-based system to functionally silence genes in primary mammalian cells, stem cells and transgenic mice by RNA interference. *Nat. Genet.*, 33(3):401–406, 2003.

Robin C.C. Ryther, Alex S. Flynt, John A. Phillips III, and James G. Patton. siRNA therapeutics: big potential from small RNAs. *Gene Ther.*, 12(1):5–11, 2005.

Ola Sætrom, Ola Snøve Jr., and Pål Sætrom. Weighted sequence motifs as an improved seeding step in microRNA target prediction algorithms. *RNA*, 11(7):995–1003, 2005a.

Pål Sætrom. Predicting the efficacy of short oligonucleotides in antisense and RNAi experiments with boosted genetic programming. *Bioinformatics*, 20(17):3055–3063, 2004.

Pål Sætrom and Magnus Lie Hetland. Multiobjective evolution of temporal rules. In *Eight Scandinavian Conference on Artificial Intelligence*, 2003a.

Pål Sætrom and Magnus Lie Hetland. Unsupervised temporal rule mining with genetic programming and specialized hardware. In *Proceedings of the International Conference on Machine Learning and Applications (ICMLA'03)*, pages 145–151, 2003b.

Pål Sætrom, Ragnhild Sneve, Knut I. Kristiansen, Ola Snøve Jr., Thomas Grünfeld, Torbjørn Rognes, and Erling Seeberg. Predicting non-coding RNA genes in *Escherichia coli* with boosted genetic programming. *Nucleic Acids Res.*, 33(10):3263–3270, 2005b.

Pål Sætrom and Ola Snøve Jr. A comparison of siRNA efficacy predictors. *Biochem. Biophys. Res. Commun.*, 321(1):247–253, 2004.

Sandeep Saxena, Zophonías O. Jónsson, and Anindya Dutta. Implications for off-target acitivity of small inhibitory RNA in mammalian cells. *J. Biol. Chem.*, 278(45):44312–44319, 2003.

Peter C. Scacheri, Orit Rozenblatt-Rosen, Natasha J. Caplen, Tyra G. Wolfsberg, Lowell Umayam, Jeffrey C. Lee, Christina M. Hughes, Kalai Selvi Shanmugam, Arindam Bhattacharjee, Matthew Meyerson, and Francis S. Collins. Short interfering RNAs can induce unexpected and divergent changes in the levels of untargeted proteins in mammalian cells. *Proc. Natl. Acad. Sci. U.S.A.*, 101(7):1892–1897, 2004.

Lisa J. Scherer and John J. Rossi. Approaches for the sequence-specific knock-down of mRNA. *Nat. Biotechnol.*, 21(12):1457–1465, 2003.

Bernard Schölkopf, Alexander J. Smola, Robert Williamson, and Peter L. Bartlett. New support vector algorithms. *Neural Comput.*, 12(5):1207–1245, 2000.

Bernhard Schölkopf. *Statistical Vector Learning*. Oldenbourg Verlag, Munich, 1997.

Dianne S. Schwarz, György Hutvágner, Tingting Du, Zuoshang Xu, Neil Aronin, and Phillip D. Zamore. Asymmetry in the assembly of the RNAi enzyme complex. *Cell*, 115:199–208, 2003.

Dimitri Semizarov, Leigh Frost, Aparna Sarthy, Paul Kroeger, Donald N. Halbert, and Stephen W. Fesik. Specificity of short interfering RNA determined through gene expression signatures. *Proc. Natl. Acad. Sci. U.S.A.*, 100(11):6347–6352, 2003.

Rocco A. Servedio. Smooth boosting and learning with malicious noise. *Journal of Machine Learning Research*, 4(4):633–648, 2003.

Despina Siolas, Cara Lerner, Julja Burchard, Wei Ge, Peter S. Linsley, Patrick J. Paddison, Gregory J. Hannon, and Michele A. Cleary. Synthetic shRNAs as potent RNAi triggers. *Nat. Biotechnol.*, 2004. Epub ahead of print.

Carol A. Sledz, Michelle Holko, Michael J. de Veer, Robert H. Silverman, and Bryan R.G. Williams. Activation of the interferon system by short-interfering RNAs. *Nat. Cell Biol.*, 5(9):834–839, 2003.

Carol A. Sledz and Bryan R.G. Williams. RNA interference and double-stranded-rna-activated pathways. *Biochem. Soc. Trans.*, 32(Pt 6):952–956, 2004.

Neil R. Smalheiser and Vetle I. Torvik. A population-based statistical approach identifies parameters characteristic of human microRNA-mRNA interactions. *BMC Bioinformatics*, 5(1):139, 2004.

Ola Snøve Jr. and Torgeir Holen. Many commonly used siRNAs risk off-target activity. *Biochem. Biophys. Res. Commun.*, 319(1):256–263, 2004.

Ola Snøve Jr., Håkon Humberset, Olaf René Birkeland, and Pål Sætrom. Sequence Explorer: interactive exploration of genomic sequence data, 2005. Manuscript.

Ola Snøve Jr., Magnar Nedland, Ståle H. Fjeldstad, Håkon Humberset, Olaf R. Birkeland, Thomas Grünfeld, and Pål Sætrom. Designing effective siRNAs with off-target control. *Biochem. Biophys. Res. Commun.*, 325(3): 769–773, 2004.

Erwei Song, Sang-Kyung Lee, Lie Wang, Nedim Ince, Nengtai Ouyang, Jun Min, Jisheng Chen, Premlata Shankar, and Judy Lieberman. RNA interference targeting Fas protects mice from fulminant hepatitis. *Nat. Med.*, 9(3):347–351, 2003.

Jürgen Soutschek, Akin Akinc, Birgit Bramlage, Klaus Carisse, Rainer Constien, Mary Donoghue, Sayda Elbashir, Anke Geick, Philipp Hadwiger, Jens Harborth, Matthias John, Venkitasamy Kesavan, Gary Lavine, Rajendra K. Pandey, Timothy Racie, Kallanthottathil G. Rajeev, Ingo Röhl, Ivanka Toudjarska, Gang Wang, Silvio Wuschko, David Bumcrot, Victor

Koteliansky, Stefan Limmer, Muthiah Manoharan, and Hans-Peter Vornlocher. Therapeutic silencing of an endogenous gene by systemic administration of modified siRNAs. *Nature*, 432(7014):173–178, 2004.

Alexander Stark, Julius Brennecke, Robert B. Russell, and Stephen M. Cohen. Identification of *Drosophila* microRNA targets. *PLoS Biology*, 1(3): E60, 2003.

Eric J. Steinmetz, Nicholas K. Conrad, David A. Brow, and Jeffry L. Corden. RNA-binding protein Nrd1 directs poly(A)-independent 3'-end formation of RNA polymerase II transcripts. *Nature*, 413(6853):327–331, 2001.

Mervyn Stone. Cross-validatory choice and assessment of statistical predictions. *Journal of the Royal Statistical Society. Series B (Methodological)*, 36 (2):111–147, 1974.

Gisela Storz. An expanding universe of noncoding RNAs. *Science*, 296 (5571):1260–1263, 2002.

Naoki Sugimoto, Shu ichi Nakano, Misa Katoh, Akiko Matsumura, Hiroyuki Nakamuta, Tatsuo Ohmichi, Mari Yoneyama, , and Muneo Sasaki. Thermodynamic parameters to predict stability of RNA/DNA hybrid duplexes. *Biochemistry*, 34(35):11211–11216, 1995.

Guangchao Sui, El Bachir Affar, Frederique Gay, Yujiang Shi, William C. Forrester, and Yang Shi. A DNA vector-based RNAi technology to suppress gene expression in mammalian cells. *Proc. Natl. Acad. Sci. U.S.A.*, 99 (8):5515–5520, 2002.

Hiroaki Tabara, Alla Grishok, , and Craig C. Mello. RNAi in C. elegans: soaking in the genome sequence. *Science*, 282(5388):430–431, 1998.

Shigeru Takasaki, Shuji Kotani, and Akihiko Konagaya. An effective method for selecting siRNA target sequences in mammalian cells. *Cell Cycle*, 3(6):790–795, 2004.

Clare E. Thomas, Anja Ehrhardt, and Mark A. Kay. Progress and problems with the use of viral vectors for gene therapy. *Nat. Rev. Genet.*, 4(5): 346–358, 2003.

Lisa Timmons and Andrew Fire. Specific interference by ingested dsRNA. *Nature*, 395(6705):854, 1998.

Kumiko Ui-Tei, Yuki Naito, Funitaka Takahashi, Takeshi Haraguchi, Hiroko Ohki-Hamazaki, Aya Juni, Ryu Ueda, and Kaoru Saigo. Guidelines for the selection of highly effective siRNA sequences for mammalian and chick RNA interference. *Nucleic Acids Res.*, 32(3):936–948, 2004.

Vladimir N. Vapnik. *Statistical Learning Theory*. Wiley-Interscience, New York, NY, USA, 1998.

J. Craig Venter et al. The sequence of the human genome. *Science*, 291 (5507):1304–1351, 2001.

Timothy A. Vickers, Seongjoon Koo, C. Frank Bennett, Stanley T. Crooke, Nicholas M. Dean, and Brenda F. Baker. Efficient reduction of target RNAs by small interfering RNA and RNase H-dependent antisense agents. A comparative analysis. *J. Biol. Chem.*, 278(9):7108–7118, Feb 2003.

Karen M. Wassarman, Francis Repoila, Carsten Rosenow, Gisela Storz, and Susan Gottesman. Identification of novel small RNAs using comparative genomics and microarrays. *Genes Dev.*, 15(13):1637–1651, 2001.

James D. Watson and Francis H. C. Crick. Molecular structure of nucleic acids. *Nature*, 171(4356):737–738, 1953.

Bruce Wightman, Ilho Ha, and Gary Ruvkun. Posttranscriptional regulation of the heterochronic gene *lin-14* by *lin-4* mediates temporal pattern formation in *C. elegans*. *Cell*, 75(5):855–862, 1993.

Tianbing Xia, John SantaLucia Jr, Mark E. Burkard, Ryszard Kierzek, Susan J. Schroeder, Xiaoqi Jiao, Christopher Cox, and Douglas H. Turner. Thermodynamic parameters for an expanded nearest-neighbor model for formation of RNA duplexes with Watson-Crick base pairs. *Biochemistry*, 37:14719–14735, 1998.

Soraya Yekta, I-hung Shih, and David P. Bartel. MicroRNA-directed cleavage of *HOXB8* mRNA. *Science*, 304(5670):594–596, 2004.

Rodrigo Yelin, Dvir Dahary, Rotem Sorek, Erez Y. Levanon, Orly Goldstein, Avi Shoshan, Alex Diber, Sharon Biton, Yael Tamir, Rami Khosravi, Sergey Nemzer, Elhanan Pinner, Shira Walach, Jeanne Bernstein, Kinneret Savitsky, and Galit Rotman. Widespread occurrence of antisense transcription in the human genome. *Nat. Biotechnol.*, 21(4):379–386, 2003.

Rui Yi, Yi Qin, Ian G. Macara, and Bryan R. Cullen. Exportin-5 mediates the nuclear export of pre-microRNAs and short hairpin RNAs. *Genes Dev.*, 17(24):3011–3016, 2003.

Koichi Yoshinari, Makoto Miyagishi, and Kazunari Taira. Effects on RNAi of the tight structure, sequence and position of the targeted region. *Nucleic Acids Res.*, 32(2):691–699, 2004.

Jenn-Yah Yu, Stacy L. DeRuiter, and David L. Turner. RNA interference by expression of short-interfering RNAs and hairpin RNAs in mammalian cells. *Proc. Natl. Acad. Sci. U.S.A.*, 99(9):6047–6052, 2002.

Phillip D. Zamore. RNA interference: listening to the sound of silence. *Nat. Struct. and Mol. Biol.*, 8(9):746–750, 2001.

Phillip D. Zamore, Thomas Tuschl, Phillip A. Sharp, and David P. Bartel. RNAi: double-stranded RNA directs the ATP-dependent cleavage of mRNA at 21 to 23 nucleotide intervals. *Cell*, 101(1):25–33, 2000.

Yan Zeng, Xuezhong Cai, and Bryan R. Cullen. Use of RNA polymerase II to transcribe artificial microRNAs. *Methods Enzymol.*, 392:371–380, 2005.

Yan Zeng and Bryan R. Cullen. RNA interference in human cells is restricted to the cytoplasm. *RNA*, 8(7):855–860, 2002.

Yan Zeng and Bryan R. Cullen. Sequence requirements for microRNA processing and function in human cells. *RNA*, 9(1):112–123, 2003.

Yan Zeng and Bryan R. Cullen. Structural requirements for pre-microRNA binding and nuclear export by Exportin 5. *Nucleic Acids Res.*, 32(16):4776–4785, 2004.

Yan Zeng, Eric J. Wagner, and Bryan R. Cullen. Both natural and designed micro RNAs can inhibit the expression of cognate mRNA when expressed in human cells. *Mol. Cell.*, 9(6):1327–1333, 2002.

Yan Zeng, Rui Yi, and Bryan R. Cullen. Recognition and cleavage of primary microRNA precursors by the nuclear processing enzyme Drosha. *EMBO J.*, 2004. Epub ahead of print.

Yong Zhang, Zhihua Zhang, Lunjiang Ling, Baochen Shi, and Runsheng Chen. Conservation analysis of small RNA genes in *Escherichia coli*. *Bioinformatics*, 20(5):599–603, 2004.

Michael Zuker. Mfold web server for nucleic acid folding and hybridization prediction. *Nucleic Acids Res.*, 31(13):3406–3415, 2003.

Papers

Paper I

A recursive MISD architecture for pattern matching

Paper I is not included due to copyright restrictions.

Paper II

A MISD architecture in a pattern-mining supercomputing cluster

A MISD architecture in a pattern-mining supercomputing cluster

Olaf René Birkeland [a] Ola Snøve Jr. [a] Arne Halaas [b]
Ståle H. Fjeldstad [a] Magnar Nedland [a] Håkon Humberset [a]
Pål Sætrom [a,*],

[a] *Interagon AS, Medisinsk teknisk senter, NO-7489 Trondheim, Norway*

[b] *Department of Computer and Information Science, Norwegian University of Science and Technology, Trondheim, Norway*

Abstract

Multiple instruction stream-single data stream (MISD) architectures have not found many practical applications in supercomputing. We present a multiple instruction stream-multiple data stream (MIMD) cluster implementation that uses MISD search processors with extreme pattern mining performance. For regular expressions, a single search processor is three orders of magnitude faster than a modern CPU running `nr-grep`. We use PCI cards that hold sixteen search processors with local memory to build a relatively small cluster of five PCs with six PCI cards each, and this cluster can handle anything between 64 independent queries at 48 GB per second or 30,720 independent queries at 100 MB per second. The cluster's performance characteristics are such that we can easily scale the system to obtain higher performance with containable overhead. Because this may be the first commercially used MISD implementation we discuss several applications in molecular biology, seismic data processing, network surveillance, and financial transaction analysis.

Key words: B.7.1.i VLSI, C.1.1.a MISD processors, C.5.6 Multiprocessor Systems, H.2.8.d Data mining, I.2.6.g Machine learning

* Corresponding author. Fax: +47 455 94 458
Email addresses: `olaf.rene.birkeland@interagon.com` (Olaf René Birkeland), `ola.snove@interagon.com` (Ola Snøve Jr.), `arne.halaas@idi.ntnu.no` (Arne Halaas), `staale.fjeldstad@interagon.com` (Ståle H. Fjeldstad), `magnar.nedland@interagon.com` (Magnar Nedland), `haakon.humberset@interagon.com` (Håkon Humberset), `paal.saetrom@interagon.com` (Pål Sætrom).

1 Introduction

Supercomputers nowadays are either single machines with sophisticated architectures or many relatively simple machines in a cluster configuration. The trend is towards clusters as illustrated by the popularity of Linux clusters. Furthermore, IBM's BlueGene/L cluster machine is now number one on the November 2004 list of the world's top supercomputers on `http://www.top500.org`. Clusters are relatively cheap alternatives to integrated machines, and have traditionally enabled people to work with problems that have been too demanding for standard workstations, but that do not require the use of every clock cycle of a vector machine (capacity versus capability computing) [48].

Still, cluster supercomputers are not cheap, and cluster solutions based on standard components carries several costs in addition to acquiring the hardware. For example, the 65,536 node BlueGene/L's estimated cost is less than \$800,000,000 [1] but still requires 9 GWh of electricity anually for the machine itself [2]. The cooling, maintenance, and hardware replacement costs of Google's 15,000 node search cluster are considerable [5].

Our aim was to build a small cluster of machines with special-purpose search processors to enable pattern mining with performance comparable to modern supercomputers at a fraction of the cost. The cluster consists of several PCs that are equipped with PCI boards that contain multiple instruction stream-single data stream (MISD) search processors with local memory. With the possible exception of systolic arrays [11], MISD architectures have been considered impractical [23], and our search processor may be one of the first MISD architectures to find practical applications (cf. [48]). Figure 1 shows the various building blocks of our pattern mining cluster.

Our MISD architecture consists of a data distribution tree whose leaf nodes are processing elements, and a result processing tree that uses the output from these elements to match regular expression-like queries. The 0.20 μm CMOS VLSI implementation of the architecture does 1.024×10^{11} character comparisons per second at 100 MHz, and may, depending on the queries's length, match up to 64 independent queries or 127 partly dependent queries in parallel [15, 16, 18]. We integrated sixteen processors on a PCI 2.2 compliant search card where each processor can access 128 MB of local memory. With five machines holding six cards each, the resulting cluster's performance can be tailored to fit the given application. Some applications require many queries screened against relatively small data volumes. Conversely, other applications require fewer queries screened against larger data volumes at extreme

[1] See `http://www.llnl.gov/pao/news/news_releases/2004/NR-04-09-15.html`

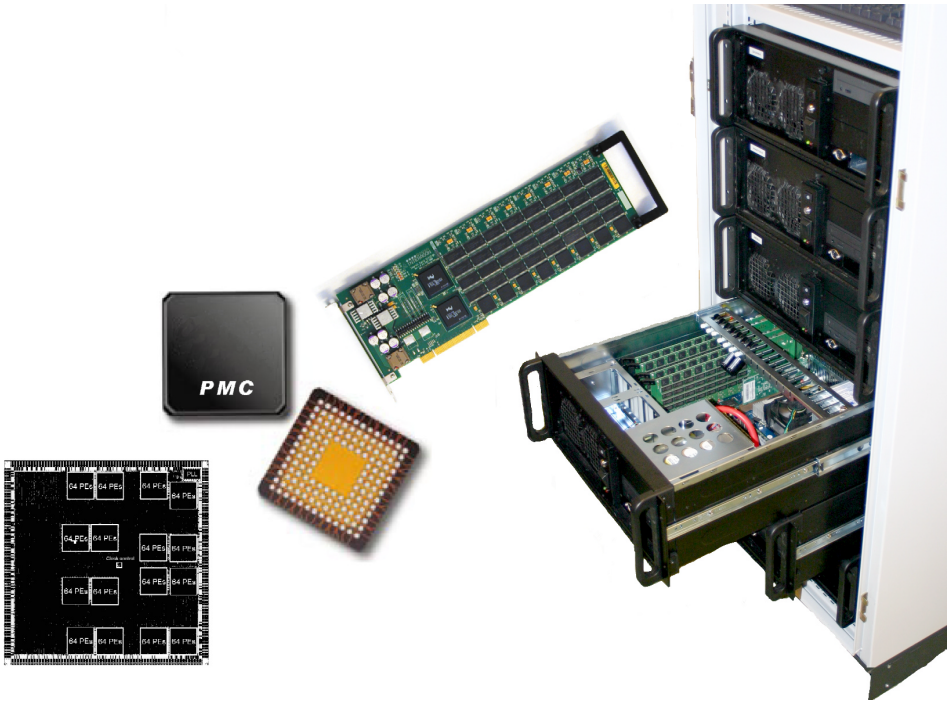

Fig. 1. Multiple instruction stream-single data stream VLSI architectures on PCI search cards have been used to design a high-performance pattern mining cluster using PCs with otherwise standard specifications.

search speed. The peak theoretical performances of our cluster is 64 independent queries at 48 GB per second and 30,720 independent queries at 100 MB per second, but we generally tailor the performance to obtain application-optimized ratios of query throughput to data volume. For example, when finding all fixed-length (25 character) subsequences with hamming distance above a given threshold to all other subsequences in a large sequence database (see Section 2), we obtain about 90 percent of theoretical performance with our current implementation, and demonstrate linear scalability. That is, the system's overhead from adding more devices is negligible.

Several architectures for approximate string matching have been proposed over the years, but a fair comparison of performance is difficult due to significant discrepancies between the algorithms [18]. Many groups have developed special purpose hardware for sequence analysis in computational biology (see for instance [21, 31]), and Hughey has published a comparison of parallel hardware for sequence comparison and alignment [22]. Our architecture implementation does almost 1,000 times more character comparisons per chip than the closest competitor [18], and the outlined cluster contains 480 search processors with a corresponding increase in performance. Commercial alternatives for regular expression matching are available from Integrated Device Technologies (Santa Clara, CA), Safenet (Belcamp, MD), TippingPoint (Austin, TX), and Tarari (San Diego, CA), but benchmarking is difficult as their designs have never been published in peer-reviewed journals. While other architectures have been pro-

posed for solving an increasing discrepancy in CPU and memory bandwidth in SIMD/MIMD processing [36], our MISD construction have reduced that problem on an architectual level.

We will motivate our design by introducing a simplified version of an important problem from computational biology, and our cluster's architecture, implementation, and performance will be described in the context of this problem. Machine learning systems can take advantage of the cluster's search capacity by continuously screening candidate pattern solutions against large datasets. To illustrate this approach, we describe applications from such different domains as molecular biology, seismic data analysis, financial transaction monitoring, and network surveillance, where advanced pattern mining is greatly aided by the high performance of our search cluster. Finally, we discuss alternative designs that could be implemented depending on the application's problem characteristics and performance requirements.

2 A motivating example from modern genetics

Many methods in genetics use short DNA or RNA molecules, so called oligonucleotide (oligo) probes, that bind to longer target sequences via base complementarity. Examples include techniques for sequence-specific knockdown of mRNA such as antisense oligos (ODNs), catalytic RNAs (ribozymes), and short interfering RNAs (siRNAs) [42]; oligo microarrays for relative measurement of gene expression [33]; and variants of the polymerase chain reaction for amplification of RNA or DNA [39]. These tools rely on short stretches of nucleotides that bind preferentially to complementary nucleotides.

Formally, you have an alphabet of four characters that can be divided into two different pairs that have a strong preference for each other. A major determinant for success is the degree of similarity between the probes and the target sequence as measured by the Hamming distance as similarity to sequences other than the target sequences results in poor performance in the aforementioned methods.

We use the notation of [34], and let $\delta(\mathbf{x}, \mathbf{y})$ denote the Hamming distance between two equally long strings $\mathbf{x}$ and $\mathbf{y}$. We define the k-neighborhood of $\mathbf{x}$ as all strings $\mathbf{y}$ that satisfy $\delta(\mathbf{x}, \mathbf{y}) \leq k$. Note that multiple probe matches within some region is usually only counted once. For example, the specificity of a microarray probe is not compromised due to multiple binding sites within the same mRNA transcript. Therefore, we let $\mathbf{T} = (\mathbf{t}_1, \ldots, \mathbf{t}_m)$ denote a target database with m documents that may correspond to genes, transcripts, exons, or other biological entities.

A library of specific oligos can be built for any entity and in general consists of patterns that are unique to that particular entity with some degree of fuzziness—that is, the pattern is found only in the target entity even if a matching region is allowed to differ somewhat from the exact pattern. A special case is the k-neighborhood library, which for the entity $\mathbf{t}_i$ consists of all oligonucleotides, $\mathbf{x} \in \mathbf{t}_i$, where $\delta(\mathbf{x}, \mathbf{y}) > k$ for all $\mathbf{y} \in \mathbf{T} - \mathbf{t}_i$; that is, all probes mismatch in at least k positions with all other equally long oligos from other transcripts. The k-neighborhood library is useful when selecting ODNs and ribozymes, as well as probes for microarray experiments, when disregarding chemical and thermodynamic properties that are important for the methods's sensitivities. siRNAs are double-stranded RNAs where both strands, at least in principle, may be active. Therefore, both the probe and its reverse complement must be equally specific to its target transcript.

The above problem can be solved by measuring the similarity between each candidate subsequence in the target and each subsequence in the rest of the target database. That is, we repeatedly search the target database with a large set of query sequences, and in the case where we want to design oligo probes against each target in the database, the number of searches approach the database size. Thus, the search problem consists of a high number of readily available queries that is screened against a static document collection. What is more, all queries can be screened against the database in parallel with very little overhead.

3 Cluster implementation

In the following, we will describe the design and implementation of a high-performance search cluster, intended to solve problems similar to the k-neighborhood library problem. More specifically, the cluster is designed to solve search problems that share the important characteristics of (i) being dividable into independent subproblems that can be solved in parallel with minimal overhead; (ii) consisting of a large number of queries that are available with limited latency; and (iii) having a relatively static dataset. Note in particular that our search architecture's functionality permits far more advanced queries than will be used in the motivating example, but in the interest of simplicity, we will describe the architecture in the context of the k-neighborhood problem that use simple mismatch similarities.

The design will be described bottom up, starting with our special purpose search architecture [18], the PCI card implementation, and how these are the cornerstones in our cluster of PCs with otherwise ordinary specifications.

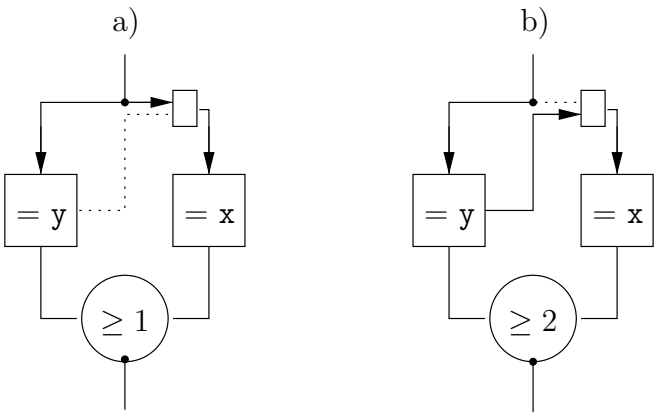

Fig. 2. Data flow, character matching, and result processing for queries a) `x|y` and b) `xy`.

3.1 A MISD search architecture

We developed an application specific integrated circuit, the Pattern Matching Chip (PMC), designed to obtain extreme performance on search problems involving complex patterns [15, 16, 18]. Our patented implementation does 1.024×10^{11} character comparisons per second and permits searching up to 64 independent patterns at 100 MB per second.

To understand the main principle of our design, visualize the overall operation of the PMC as a stream of data flowing through the chip from left to right. The data is distributed to 1,024 processing elements (PEs) via a binary data distribution tree. The results from the PEs's comparisons are then used to obtain the final output in a result processing tree. We illustrate the chip's basic function with two simple queries, namely `x|y` and `xy`. That is, either an `x` or a `y` in the former, and an `x` directly followed by a `y` in the latter. Figure 2 shows how the chip is configured to get the desired results. Matching either of two characters is done by configuring the two PEs to match the respective characters, have them receive the data stream in parallel, and make the result processing tree perform a boolean `OR` operation by reporting a match if the sum of its two children's results is greater than or equal to one. Note that the data flow is illustrated by solid lines in the data distribution tree. Similarly, the expression `xy` is matched if the data flow becomes sequential—that is, the rightmost PE receives the data flow from its left neighbor—and the result processing node's operation is changed to the `AND` operator.

The processing elements and the result processing nodes can perform several more advanced operations. The functionality is sufficient to implement limited regular expression matching [17] excluding nested Kleene closures with constant response time in arbitrary data. A description of the architecture along with advanced configuration examples have been published elsewhere [18], and

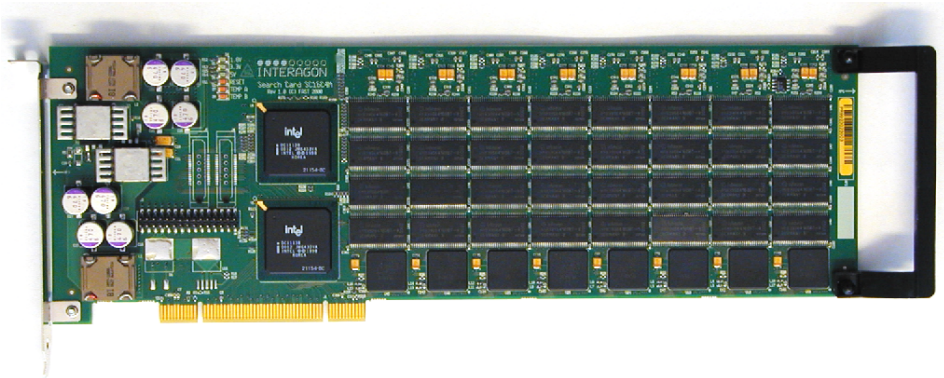

Fig. 3. Search card with 16 PMC chips. Eight chips are mounted along the bottom edge of the card, another eight along the top edge on the reverse side. Local memory is located between the PMCs on both sides of the card, and the left hand side of the card is used for system interface and local power supply.

a detailed technical note is available from the authors upon request.

3.2 A PCI search card for PC integration

We designed the PMC chip with an integrated PCI interface as we intended to place PMCs on accelerator cards. Fig. 3 shows our current PCI accelerator card. In this implementation, the host system has full control over each individual PMC through transparent PCI-PCI bridges.

Each card holds 16 PMC chips along with local memory, typically 2 GB per card, which gives 128 MB of dedicated memory per PMC. With 1,024 PEs per chip, the card carries an accumulated 16,384 PEs within a single full length PCI card. The card is powered through the PCI slot and has a peak power consumption less than 25W. Note that the PMCs use a lower core voltage (1.8V) than is available in the PCI bus, and the card therefore contains a separate DC/DC converter. The distribution of processing across several chips results in that there is no local hot spot that requires a fan on the card. Consequently, there are no moving parts, and that results in less power consumption, less noise, and increased reliability. These favorable features are highly important when scaling into a larger system.

Using exclusively low-profile surface mount components, the cards can be stacked side by side in adjacent PCI slots without restricting the system's airflow. In a typical system, this allows six PCI cards to be inserted into each server, or a total of 98,304 PEs per machine, and at 100 MHz each, this accumulates to about 10^{13} operations per second. Any server grade power supply easily handles this added system load of 150W.

3.3 Resource scheduling for optimal scalability

The search problems we are considering scale in two independent dimensions, namely (i) query volume and (ii) data size. Risvik describes a general framework for designing clusters to handle this kind of search applications [37]. The main principles in this framework are partitioning of data to handle larger data sizes and duplication of data to handle larger query volumes. These design principles have also been used in the Google search engine [5].

We use the above partitioning principles in our cluster implementation. First, because of the 128 MB of memory dedicated to each PMC, we partition a given dataset of size d MB on $p = \lceil d/128 \rceil$ PMCs. To minimize communications between nodes in the cluster when joining the search results, we generally divide the data on PMCs located on the same cluster node. Nevertheless, our cluster implementation can also handle searches in larger datasets that must be partitioned on several nodes. For example, in a cluster where each node has 6 search cards, giving a maximum size per node of $6 \times 16 \times 128$ MB = 12 GB, we partition a 20 GB dataset on two nodes. In the following, we will, however, only discuss search problems where the dataset can fit on one node.

Second, we duplicate the dataset on the remaining PMCs in the cluster. Thus, in a cluster with n nodes, each equipped with m search cards, we would at most search $\left\lfloor \frac{n \cdot m \cdot 16}{p} \right\rfloor$ instances of the dataset. Requiring that each dataset is located on the same node reduces the number of instances to $n \cdot \left\lfloor \frac{m \cdot 16}{p} \right\rfloor$. In the worst case, this latter strategy will leave nearly half the PMCs in the cluster idle (when the dataset requires $8m + 1$ PMCs). Even so, the loss is negligible when p is much smaller than $16m$; for example, a recent compilation of human transcripts is about 360 MB (`http://www.ncbi.nlm.nih.gov/IEB/Research/Acembly`) and therefore requires three PMCs.

Given some highly parallel search problem, having a large number of queries that can be parallelized with little overhead, our current partitioning and scheduling algorithm works as follows:

(1) *Cluster distribution.* The total query space is divided evenly on the nodes in the cluster.
(2) *Node distribution.* Each node process duplicates the dataset to be searched on its available PMCs, and runs a multi-threaded parallel search on these PMCs.

To illustrate this scheduling process, consider the k-neighborhood library problem described in Section 2. First, we evenly distribute the $|T|$ entities against which the library should be designed to the nodes in the cluster. Second, a separate thread at each node parses the entities, generates queries for each of

Table 1
Parameters for different system sizes using the current PCI card. Memory per PMC can be configured between one 64 Mbit SDRAM chip to four 1Gbit devices. 128 MB per PMC is used in this table.

System	PEs	Comparisons/s	Memory	Bandwidth	Power
One chip	1,024	10^{11}	128 MB	100 MB/s	1 W
PCI card	16,384	$1.6{\cdot}10^{12}$	2 GB	1.6 GB/s	25 W
One server	98,304	10^{13}	12 GB	9.6 GB/s	350 W
5 nodes	491,520	$5{\cdot}10^{13}$	60 GB	48 GB/s	2 kW
100 nodes	9,830,400	10^{15}	1.2 TB	1 TB/s	35 kW

the subsequences to be evaluated, and pushes the queries on a synchronized search queue. The PMC search threads read the queue, run the searches, and write the results to a common result pool for post processing. In Section 4.4, we compare the performance of this solution to the theoretical maximum performance of our cluster.

3.4 A cluster with great flexibility

Our current search cluster consists of five rack-mounted PCs, each being single Pentium4® CPU systems with 1 GB RAM running disk-based Debian Linux as is freely available from `http://www.debian.org`. The CPU speed ranges from 2.4 GHz to 2.8 GHz. Each node has six search cards for a total of $5 \cdot 10^{13}$ comparisons per second. We currently connect the nodes through a 100 Mbps Ethernet switch, but as each node has a gigabit Ethernet interface, we can easily upgrade the cluster network to 1 Gbps if needed.

As Table 1 indicates, we reach 1 peta comparisons per second with 100 cluster nodes. To get this size, we can extend our current fully meshed design by adding additional routers and switches. Alternatively, we can use the tree-based design of [37]. The mesh design is more flexible, but the tree design will reduce network communications, as this can allow dedicated nodes for post-processing if required. We would therefore prefer the tree design when, for instance, searching datasets that must be distributed over more than one cluster node.

4 Results

We now present the characteristics of our cluster solution. That is, we present the requirements for maximal search throughput in the cluster, the memory bandwith characteristics, and the cluster's power consumption. Finally, we compare the cluster's observed performance to its theoretical maximum performance on the k-neighborhood screening problem outlined in Section 2.

4.1 Theoretical search throughput and scalability

As outlined in Section 3.3, the two main scalability factors to consider are query volume and data size. The following calculations assume that the entire dataset can be held in the PMC's local memory. During any search, the CPU will upload the configuration for the next search into local memory. The configuration upload time for a single pass in each server is

$$t_c = \frac{M \cdot C}{S} \tag{1}$$

where M is the number of PMCs in each server, C is the configuration image size for each chip, and S is the effective PCI bandwidth. In one machine with ninety-six PMCs, t_c becomes approximately 30 ms given an effective PCI bandwith of 50 MB per second.

The search time for a single pass of the data is given as

$$t_s(n) = \frac{n}{\theta_{max}}, \tag{2}$$

where n is the amount of data distributed to each chip, and θ_{max} is the search speed of a single PMC.

Hence, the effective search throughput for each chip is given as

$$\theta_s(n) = \begin{cases} \theta_{max} & \text{if } t_c < t_s(n) \\ \frac{t_s(n)}{t_c}\ \theta_{max} & \text{otherwise} \end{cases} \tag{3}$$

We may combine equations 1 through 3 to obtain the minimum dataset n that must be used for a system of M PMCs to run at peak bandwidth. A single PMC must search more than three megabytes per pass, and a six card machine with ninety-six PMCs consequently must be configured with almost 300 megabytes or more. As the search time is linearly dependent on the data

size, the search throughput will not be altered if data is added beyond the minimum requirement.

Adding more queries can be handled either by running queries in parallel on more PMCs, alternatively evaluating groups of queries serially. Either way, more queries or data only requires a linear increase in run time or compute resources.

One final limitation stems from the fact that we use the PCI bus for reporting results. The results can be directed to any PCI device, even the local memory of each PMC. In the lack of postprocessing resources locally on the card as described in section 6.1), results are commonly routed to system memory located at the root of the PCI bus complex. The bandwidth of this resource is very system dependent, but limits the use of the current system implementation to applications with relatively low hit rates. A standard desktop system typically delivers 50 MB/s bandwidth. At four bytes per hit this is equivalent to 12.5 million hits per second.

4.2 Memory bandwidth

In a conventional SIMD processor architecture, the memory bandwith is usually a major performance bottleneck. Memory latency reductions come at the expense of increased bandwith requirements [10]. Traditional processors compensates the lack of bandwith with SIMD vector processing to minimize instruction stream bandwidth [29]. Still, there is a high load on the memory subsystem. System testing shows a range from 0.2 bytes of data traffic per instruction for computationally intensive programs, to 7 bytes per instruction for memory intensive benchmarks [36].

The MISD architecture of the PMC implies that each memory access is used more efficiently, i.e. by being processed simultaneously by several PEs. Even a petacomp cluster consisting of 100 servers (see Table 1) will achieve full performance with $100 \cdot 6 \cdot 16 \cdot 100$ MB/s = 1 TB/s bandwidth. Ordinary low-cost SDRAM can provide this. Note that the PMC architecture does not have a separate instruction stream requiring bandwidth during searches as the instructions are stored in configuration registers inside each chip.

One limitation in the current implementation is the loading of data into local memory. Each server in the system can load any data in parallel to the others. Within each node, the loading of data is done by the CPU, typically at 50–80 MB per second. If all the memory within one node should be loaded with unique data, it will take 3–4 minutes to load all 12 GB. If data are to be duplicated across PMCs, broadcast can reduce this by an factor of eight. This is still a considerable amount of time compared to the time required for a

single search. Thus the current system implementation leans towards solutions where the data are relatively static. Note that broadcasting is not specified in the PCI standard, but it can nevertheless be implemented by non-compliant software configuration of the local PCI bus segments of each accelerator card.

4.3 Power consumption

With an energy-efficient MISD architecture, the PMC system requires very little power to operate. In a server configuration, the PCI cards consume 35 pJ per character comparison including system overhead (cf. table 1). As a comparison, a CPU requires about 150 times more energy per character comparison assuming 200W system power and the optimistic theoretical performance described in Section 6.3.

4.4 Application performance

As outlined in Section 2, the k-neighborhood screening problem can be solved by measuring the similarity between each candidate subsequence in the target and each subsequence in the rest of the target database. In the following, we present our k-neighborhood screening solution.

To do a k-neighborhood screening, we use the PMC's Hamming distance functionality [18]. The binary tree structure of the PMC's data distribution and result gathering tree results in that a k-neighborhood screening of a string of length n will use

$$\pi(n) = 2^{\lceil \log_2 n \rceil} \tag{4}$$

PEs. As a single PMC has 1,024 PEs, it can handle $1024/\pi(n)$ k-neighborhood screenings in parallel. So, for example, one PMC can screen 32 25mers at once (a 25mer is a subsequence of length 25). Note that the number of parallel screenings is independent of the size of the neighborhood.

A single PMC can screen up to 128 MB at a rate of 100 MB per second. Thus, if we only consider sequences that are shorter than 128 MB, a single PMC can theoretically get a throughput (short oligonucleotide queries per second) of

$$\theta(d, n) = \frac{1024}{\pi(n)} \cdot \frac{100}{d}, \tag{5}$$

where d is the size of the sequence to be screened (in MBs) and $\pi(n)$ is defined in (4). By considering more than one PMC, we can extend this result to an arbitrary data size:

$$\theta(d, n, p) = \theta(d, n) \cdot p, \tag{6}$$

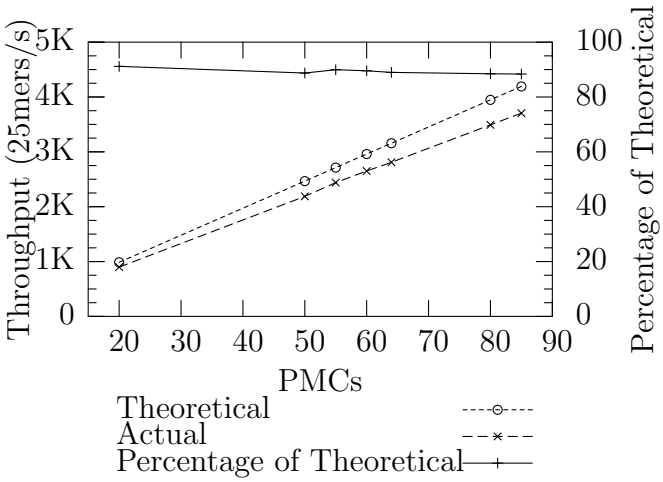

Fig. 4. Performance statistics for 25mer screenings of the human transcriptome. The theoretical performance is plotted with the actual performance, as well as with a line showing the ratio between the two numbers.

where p is the number of PMCs used and $\theta(d, n)$ is defined in (5). This means that four PMCs screening 25mers in a database of 200 MB will have a throughput of 64 25mers per second. A node in our search cluster, having six PCI cards, achieves a theoretical throughput of 1,536 25mers per second on the same database.

To test the system, we built libraries that contain the most specific 25mers from any genomic transcript in the latest Ensembl release of Human cDNA [7]. This dataset is 65 MB, which means that it fits on a single PMC and that the PMCs can run at full search speed (see Equation (3)).

Figure 4 shows the true performance plotted against the theoretical performance when using more PMCs in the oligonucleotide screening application. Note that the performance is nearly linearly scalable, hence increasing the number of PMCs will increase the performance accordingly.

In general, the standard similarity search algorithms from computational biology cannot be used as these either lack in performance as is the case with Smith-Waterman [43] or in sensitivity as is the case with BLAST [3] (cf. [44]). To put our performance figures in perspective, we compare them to the results reported by [34], who used a dynamic programming algorithm to create a complete k-neighborhood library for a small dataset of 10^6 nucleotides. They screened 10^4 25mers in approximately one hour on a single CPU of a Compaq GS80 server. This gives a throughput of approximately three 25mers per second on this dataset. Because the search time of their algorithm scales linearly with the size of the database, they would have a throughput of about $4 \cdot 10^{-2}$ 25mers per second (or about three 25mers per minute) when screening the human transcriptome. This means that they would need about 10^3 CPUs

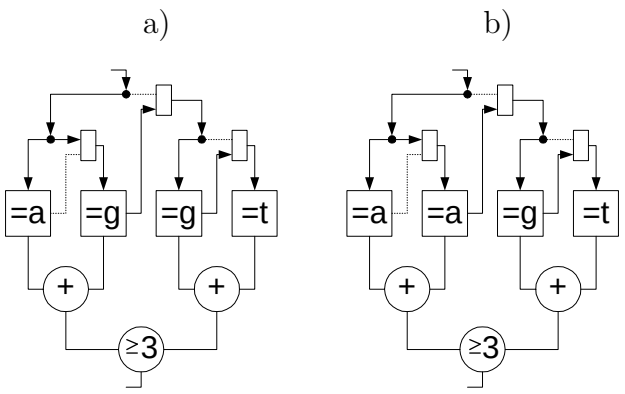

Fig. 5. Implementing G/U-wobbles (a) and differential weights (b) for the trimer `tga`. The configuration in (a) matches `tga` and `tgg`. The configuration in (b) matches `.ga` and `t.a`, where `.` denotes any character.

to reach the throughput of a single PMC and nearly 10^5 CPUs to reach the throughput of a single PMC server.

Recently, Yamada and Morishita [50] reported an index-based solution for k-neighborhood screenings of 19mers, with $k \leq 4$, to be used in RNAi experiments. Using a database of human transcripts (its exact size not listed), and $k = 3$ and $k = 4$, they report a throughput of $1.4 \cdot 10^2$ and 37 19mers per second on a Dell Precision 650 with a 3.2 GHz Xeon CPU and 2 GB main memory. Although this throughput is comparable to that of the PMC (for $k = 3$ and $k = 4$ one of their CPUs is equivalent to 3.2 and 0.82 PMCs), their solution do not share the PMCs flexibility.

As Figure 5 shows, we can easily allow for G/U-wobble base-pairings (Panel a), which seem to be tolerated by the RNAi machinery in some cases [40]. What is more, we can give different weights to different positions in the 19mer so that mismatches at specific positions are considered more or less important (Figure 5, Panel b). Both possibilities are important, as others have reported that mismatches at the ends of the siRNAs are well tolerated (Dr. Torgeir Holen, private communication), but that mismatches in the middle of the siRNAs abolish their silencing effect [14]. As neither the dynamic programming solution of [34] nor the index solution of [50] can easily support such positional weighting, our solution has an advantage.

5 Pattern mining applications

The k-neighborhood screening problem referred to throughout this paper, is an example of a search problem where you want to find instances in the dataset that satisfy some known properties. The opposite problem occurs

when you have a dataset with some known properties, and you want to create a model that characterizes parts of—or even the complete—dataset. In the former problem, you know the query and want to find the data that the query matches; in the latter, you want to find the queries that characterize the data. The latter problem is also known as data mining.

We have previously described a boosted, hardware accelerated, genetic programming algorithm [38], which creates models that characterize sequences belonging to some conceptual class. In [38], we used this algorithm to create models that predicted whether short oligonucleotides were effective when used in antisense and RNAi experiments.

Cluster-based solutions are common when using machine learning to solve data mining problems (for example [12, 27]), and there are two main reasons for this. First, using machine learning requires several independent experiments to establish good method accuracy (for example bootstrap [13] and cross-validation [8, 46]—compared in [26]). These independent experiments require no coordination, and are consequently easy to run in parallel [12]. Second, many data mining problems require large CPU resources to be solved, hence parallelizing the algorithm may be the only way to get results [27, 28].

We use the above approaches to parallelize our data mining algorithm on the search cluster. First, we do several independent runs on each cluster node—the number of parallel runs depending both on the number of PMCs available and the total CPU load. Second, we partition large datasets on several PMCs, not only to handle datasets larger than 128 MB, but also to speed the search process on smaller datasets (as per Equation 3).

In the following, we will outline several unpublished application case studies that fit our search cluster and machine learning algorithm. These include mining in biological sequences in the form of microRNA target prediction, seismic data processing, financial knowledge mining, and network surveillance. Note that we have also used the search cluster to analyze time series [20].

5.1 MicroRNA target prediction

MicroRNAs (miRNAs) are short RNAs that regulate genes by binding to the gene's mRNA [6]. MicroRNAs can either cause the mRNA to be degraded or to prevent protein synthesis—the outcome depends on how well the miRNA binds to its target site. More specifically, near perfect binding to the target site causes degradation; a more imperfect match blocks translation. There is some knowledge of how a miRNA binds its target sites, and several algorithms have, with some success, used this knowledge to predict potential target sites (see [30] for an overview). Few human miRNA target sites have been verified,

however, and better algorithms for predicting target sites is still needed [30].

As an alternative to existing miRNA target prediction algorithms, we have used our boosted genetic programming algorithm to develop a miRNA target predictor (O. Sætrom et al., manuscript in preparation). To develop this predictor, we used verified target sites and random 3' UTR sequences as positive and negative training sets. Given a miRNA sequence, our predictor generates several search queries, which it evaluates on the PMC. The predictor then combines the search results into a prioritized list of candidate miRNA target sites.

Not only did the search cluster speed the training process so that we effectively could run the necessary cross-validation experiments to establish that the predictor was both valid and accurate. The cluster will also be invaluable when using the predictor, both when running large scale predictions of miRNA target sites, and when determining probable miRNA off-target effects in siRNA experiments [41].

5.2 Seismic data processing

One of the major challenges in modern oil exploration is to extend the lifetime of existing oil fields. This can be done by analyzing seismic data to predict optimal drill paths for future production wells. Seismic data is a representation of the subsurface structure that is generated by recording the earth's response to energy pulses in the form of sound waves. Seismic data are typically recorded in cubes with a reflection amplitude for each voxel in the volume. Local patterns within the seismic cube may give information about composition, fluid content, property and geometry of rocks in the subsurface [9]. Our machine learning platform trains a program to recognize specific patterns such as those related to the porosity of the sand. The idea of using machine learning to classify seismic data is not new, and has proven successful in several seismic applications [9, 32].

Our high-performance pattern mining system is well suited for this application because of the huge data volumes, and the inherent complexity and heterogeneity of the earth's crest. A seismic survey of 250 square kilometers generates a cube that contains about 4 billion voxels. In addition to the reflection amplitude itself, several derived attributes are also included in the training set [4, 47]. Some subsurface properties, as for instance gas-filled sand, have obvious patterns visible to the human eye. Other properties are extremely difficult to detect due to minor differences in their geophysical properties, and due to acoustic noise and interference with other reflections. Our pattern mining system has been used on two different seismic applications; classifying

sand types based on their degree of porosity, and predicting the depth of the oil water contact within the reservoir.

A representative training set is necessary, and is generated from logging test wells, or from synthetic models representative for the properties to be classified. The test wells are positioned within the seismic cube, which results in a mapping between observed properties and seismic reflection in each voxel. The learning process identifies patterns that are correlated with the well logs, and these are later used to predict properties in the overall seismic volume. The final prediction indicates optimal drill paths for production wells.

5.3 Financial knowledge mining

It is of significant interest to be able to classify financial information. Applications include credit rating, insurance risk assessment, and transaction monitoring. E.g, in credit card transaction monitoring, valid and fraudulent transactions should be separated prior to processing [24, 49]. With hardware accelerated machine learning, our high-performance system could find transaction classifiers based upon training on past records, and applying the predicted classifiers to new transactions. By optimizing sensitivity and specificity according to the cost of false positives (loss of revenue when incorrectly blocking a credit card) and false negatives (covering losses due to fraud), a cost optimal balance can be found.

5.4 Network surveillance

Network content monitoring is used in applications such as virus scanning, intrusion detection, and surveillance [45]. The PMC is useful in content surveillance when looking for individual or combinations of complex pattern classifiers. If unknown, these classifiers can be found with machine learning. Even with well known classifiers, the PMCs unprecedented performance for screening real time data streams could be required. With an integrated network connection as outlined in section 6.2, the rapid response time, combined with no index preprocessing steps, enables novel search approaches that would otherwise become too expensive.

6 Potential system enhancements

Depending on the application, it may be preferable with slightly different designs. In this section, we list some alternatives that can easily be implemented

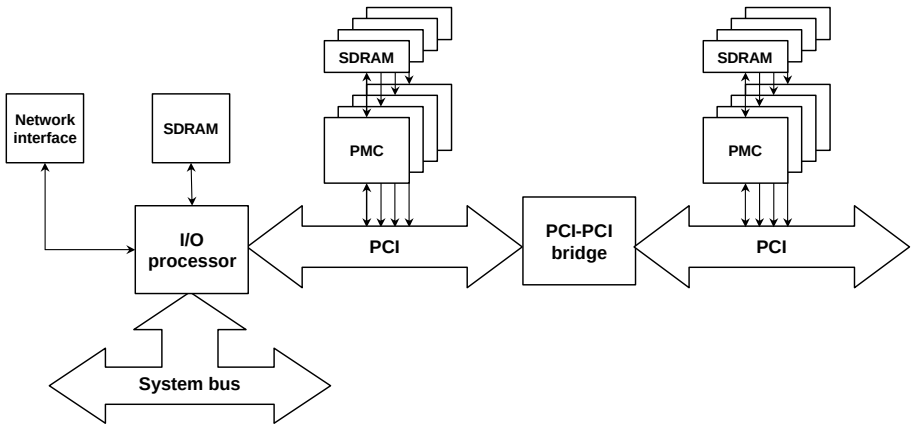

Fig. 6. Block diagram of accelerator card with built in I/O processor and optional network interface. The I/O processor might also connect directly to other peripherals like a storage subsystem.

on our existing search card if required by a specific application.

6.1 I/O processor on card

Adding an I/O processor to each accelerator card is the obvious next step for improving the system performance. These improvements come from one or more of the following factors:

- *Reduced configuration time.* Maintaining 16 PMCs on each card, these are now configured from the local I/O processor instead of the CPU. This alters the configuration time in Equation (1) by reducing M to a fixed value of 16. Correspondingly, optimal query throughput can be achieved with as little as 0.5 MB of data per PMC for each query.
- *Faster data loading.* Loading of data can be handled locally without CPU intervention. This could be from a shared system resource like the disk drive, or with direct connection to a storage subsystem. Combining broadcasted writes with a sufficient data storage, thoughputs of 800 MB per second are achievable.
- *Reduced main CPU load.* The I/O processor can parse and map high level queries to PMC register configurations. This offloads the CPU computationally and in bandwidth as only compact high level queries need to be sent to each card. In practice, each I/O processor could run the cluster node's applications.

The I/O processor would require a software system of its own, including a small operating system. Due to the added complexity, this option was not chosen for the initial system design. It would also add a device dependent

power consumption increase in the range of 5–10 W. This would affect the potential density of PMC chips within each server.

An alternative to implementing an I/O processor on the card would be to reduce the host server to a minimum. This could for example be a blade server managing a limited number of PMCs. Most commercial blade servers include a proprietary expansion slot, thus such a design would have to be customized for a specific server brand. The blade server solution would be more costly and larger than using an I/O processor, but with the benefit of a single processor architecture for the software.

6.2 Network interface on card

In the network surveillance application described in section 5.4, the monitored stream is fed to the PMCs through the PCI system bus. By adding a network interface to the card itself, this bottleneck can be removed. This would most likely be combined with an I/O processor as described in section 6.1, which will handle the network stack as well as all of the PMC resources.

6.3 CPUs instead of PMCs

The high production volumes of CPUs allow extensive design and manufacturing efforts. Consequently one should expect a CPU to achieve higher clock rates than a standard cell ASIC like the PMC. With an increasing number of parallel pipelines in the CPU, the CPU's number of operations per second will approach the PMC's. For example, a 3.8 GHz Pentium4® processor can in one clock cycle execute eight byte comparisons [2] with the streaming single instruction multiple data (SIMD) unit, potentially in parallel with two ALU-operations [1]. Theoretically, this accumulates to 38 billion comparisons per second, approaching the 100 billion comparisons for a single PMC chip. Despite this narrow gap in performance, the PMC architecture has several advantages that makes it more feasible for high-performance pattern mining clusters than a standard CPU.

- *The PMC operates closer to peak performance.* The SIMD architecture of the CPU allows parallel comparisons, but not parallel branching dependent of individual results. For anything but long fixed keyword comparisons, which are easily handled by indexing rather than brute force comparisons,

[2] There is also a 16 byte comparison instruction available, but this can only be issued every second clock cycle, rendering the same throughput.

the CPU will not reach its peak performance. The MISD architecture of the PMC chip, or the MIMD architecture of our cluster, is more appropriate.
- *The PMC executes patterns directly, without overhead.* Evaluating regular expressions is much more than the low level comparisons. While the PMC has separate units for handling these functions, a CPU must use the same processing core as for the comparisons themselves. Fine grained pattern matching results in a large number of data dependent branching, which effectively kills the performance of super-scalar specular out-of-order execution.
- *The PMC's design has a much higher potential.* In this comparison, the CPU has a technology advantage being fabricated on a 90 nm process. With a similar process the PMC would integrate five times more processing elements, in addition to a potential increase in operating speed.
- *The PMC has much lower power requirements.* The thermal design power of the CPU used is 115 W, excluding required support circuitry and memory. For the PMC, this number is 1 W for the chip alone, 1.5 W including peripherals and memory.

Taking these factors into account, the performance advantage of the PMC increases, especially when building a petacomp cluster (see Table 1). Using `nrgrep` [35] on a 1 GHz Pentium3® as a benchmark for evaluating regular expressions, a *single* PMC demonstrated a three orders of magnitude increase in speed [19]. The PMC system also scaled better with increasing data volumes.

6.4 Integration of PMC and memory

Even denser systems can be built by integrating memory and processing on the same die. This approach is limited to relatively small memory arrays, e.g. 8 MB per chip, as embedded memory can not be packed as dense as in a separate memory chip. As an intermediate alternative, a multi chip module (MCM) can be constructed.

Any such integration eliminates the memory sizing flexibility with the current configuration, as well as the cost advantage of using standard memory components. The integration also implies suboptimal implementation of both the compute and memory function [25, 36]. It would thus only be viable for applications with moderate memory demands.

7 Summary of this work

We have presented a supercomputing cluster for pattern mining purposes. The MIMD cluster is based on search processors with MISD architectures, and it seems that this is one of the first MISD implementations ever to find practical applications. The search processor's architecture is patented [15, 16] and details on its functionality and performance in a single chip configuration has been published elsewhere [18].

We have demonstrated that the performance of our cluster is orders of magnitude higher for pattern mining purposes than can be obtained with similar-sized clusters of machines with ordinary CPUs. Complete parallelization is needed if an application is to take full advantage of the cluster's performance. We described how our boosted genetic programming-based machine learning system does that, and how it can be used in numerous pattern mining applications in such diverse sectors as biotechnology, seismics, networks, and finance.

We are now working on commercial aspects of the outlined applications where we benchmark the cluster against other systems.

Acknowledgment

The work was supported by the Norwegian Research Council, grant 151899/150, and the bioinformatics platform at the Norwegian University of Science and Technology, Trondheim, Norway.

References

[1] IA-32 Intel® architecture optimization reference manual. Technical report, Intel Corporation, 2004. This manual is available from `http://developer.intel.com/design/Pentium4/documentation.htm`.

[2] N. R. Adiga, G. Almasi, G. S. Almasi, Y. Aridor, R. Barik, D. Beece, R. Bellofatto, G. Bhanot, R. Bickford, M. Blumrich, A. A. Bright, J. Brunheroto, C. Cacaval, J. Castaños, W. Chan, L. Ceze, P. Coteus, S. Chatterjee, D. Chen, G. Chiu, T. M. Cipolla, P. Crumley, K. M. Desai, A. Deutsch, T. Domany, M. B. Dombrowa, W. Donath, M. Eleftheriou, C. Erway, J. Esch, B. Fitch, J. Gagliano, A. Gara, R. Garg, R. Germain, M. E. Giampapa, B. Gopalsamy, J. Gunnels, M. Gupta, F. Gustavson, S. Hall, R. A. Haring, D. Heidel, P. Heidelberger, L. M. Herger, D. Hoenicke, R. D. Jackson, T. Jamal-Eddine, G. V. Kopcsay,

E. Krevat, M. P. Kurhekar, A. P. Lanzetta, D. Lieber, L. K. Liu, M. Lu, M. Mendell, A. Misra, Y. Moatti, L. Mok, J. E. Moreira, B. J. Nathanson, M. Newton, M. Ohmacht, A. Oliner, V. Pandit, R. B. Pudota, R. Rand, R. Regan, B. Rubin, A. Ruehli, S. Rus, R. K. Sahoo, A. Sanomiya, E. Schenfeld, M. Sharma, E. Shmueli, S. Singh, P. Song, V. Srinivasan, B. D. Steinmacher-Burow, K. Strauss, C. Surovic, R. Swetz, T. Takken, R. B. Tremaine, M. Tsao, A. R. Umamaheshwaran, P. Verma, P. Vranas, T. J. C. Ward, M. Wazlowski, W. Barrett, C. Engel, B. Drehmel, B. Hilgart, D. Hill, F. Kasemkhani, D. Krolak, C. T. Li, T. Liebsch, J. Marcella, A. Muff, A. Okomo, M. Rouse, A. Schram, M. Tubbs, G. Ulsh, C. Wait, J. Wittrup, M. Bae, K. Dockser, L. Kissel, M. K. Seager, J. S. Vetter, and K. Yates. An overview of the BlueGene/L supercomputer. In IEEE, editor, *SC2002: From Terabytes to Insight. Proceedings of the IEEE ACM SC 2002 Conference*, 2002. ISBN 0-7695-1524-X. URL `http://www.sc-2002.org/paperpdfs/pap.pap207.pdf`.

[3] Stephen F. Altschul, Warren Gish, Webb Miller, Eugene W. Myers, and David J. Lipman. Basic local alignment search tool. *J. Mol. Biol.*, 215 (3):403–410, 1990.

[4] Arthur E. Barnes. Seismic attributes in your facies. *CSEG Recorder*, pages 41–47, September 2001.

[5] Luiz André Barroso, Jeffrey Dean, and Urs Hölzle. Web search for a planet: The google cluster architecture. *IEEE Micro*, 23(2):22–28, 2003.

[6] David P. Bartel. MicroRNAs: genomics, biogenesis, mechanism, and function. *Cell*, 116(2):281–297, 2004.

[7] E. Birney, D. Andrews, P. Bevan, M. Caccamo, G. Cameron, Y. Chen, L. Clarke, G. Coates, T. Cox, J. Cuff, V. Curwen, T. Cutts, T. Down, R. Durbin, E. Eyras, X.M. Fernandez-Suarez, P. Gane, B. Gibbins, J. Gilbert, M. Hammond, H. Hotz, V. Iyer, A. Kahari, K. Jekosch, A. Kasprzyk, D. Keefe, S. Keenan, H. Lehvaslaiho, G. McVicker, C. Melsopp, P. Meidl, E. Mongin, R. Pettett, S. Potter, G. Proctor, M. Rae, S. Searle, G. Slater, D. Smedley, J. Smith, W. Spooner, A. Stabenau, J. Stalker, D. Storey, A. Ureta-Vidal, C. Woodwark, M. Clamp, and T. Hubbard. Ensembl 2004. *Nucleic Acids Res.*, 32(1):468–470, 2004.

[8] L. Breiman, J. H. Friedman, R. A. Olshen, and C. J. Stone. *Classification and Regression Trees*. Wadsworth, Belmont, CA, 1984.

[9] Alistair R. Brown. *Interpretation of three-dimensional seismic data*. The American Association of Petroleum Geologists and the Society of Exploration Geophysicists, 5th edition, 1999.

[10] Doug Burger, James R. Goodman, and Alain Kägi. Memory bandwidth limitations of future microprocessors. In *23rd International Symposium on Computer Architecture (ISCA)*, pages 78–89. IEEE Computer Society, May 1996.

[11] Alan Chalmers and Jonathan Tidmus. *Practical Parallel Processing*. International Thompson Computer Press, 1996.

[12] I. Dutra, D. Page, V. Santos Costa, J. Shavlik, and M. Waddell. Toward

automatic management of embarrassingly parallel applications. In *Euro-Par 2003 Parallel Processing*, volume 2790 of *Lecture Notes in Computer Science*, pages 509–516, 2003.

[13] Bradley Efron and Robert J. Tibshirani. *An Introduction to the Bootstrap*. Chapman & Hall, New York, 1993.

[14] Sayda M. Elbashir, Javier Martinez, Agnieszka Patkaniowska, Winfried Lendeckel, and Thomas Tuschl. Functional anatomy of siRNAs for mediating efficient RNAi in drosophila melanogaster embryo lysates. *EMBO J.*, 20(23):6877–6888, 2001.

[15] Fast Search & Transfer ASA. Digital processing device, 2000. International publication number WO 00/22545.

[16] Fast Search & Transfer ASA. A processing circuit and a search processor circuit, 2000. International publication number WO 00/29981.

[17] Jeffrey E.F. Friedl. *Mastering Regular Expressions*. O'Reilly, Cambridge, MA, 2nd edition, 2002.

[18] Arne Halaas, Børge Svingen, Magnar Nedland, Pål Sætrom, Ola Snøve Jr., and Olaf Renè Birkeland. A recursive MISD architecture for pattern matching. *IEEE Trans. on VLSI Syst.*, 12(7):727–734, 2004.

[19] Magnus Lie Hetland and Pål Sætrom. A comparison of hardware and software in sequence rule evolution. In *Eight Scandinavian Conference on Artificial Intelligence*, 2003.

[20] Magnus Lie Hetland and Pål Sætrom. Evolutionary rule mining in time series databases. *Mach. Learn.*, 58(2–3):107–125, 2005.

[21] Jeffrey D. Hirschberg, David M. Dahle, Kevin Karplus, Don Speck, and Richard Hughey. Kestrel: A programmable array for sequence analysis. *Journal of VLSI Signal Processing Systems for Signal Image and Video Technology*, 19(2):115–126, 1998.

[22] Richard Hughey. Parallel hardware for sequence comparison and alignment. *CABIOS*, 12(6):473–479, 1996.

[23] Kai Hwang and Fayé A. Briggs. *Computer Architecture and Parallel Processing*. McGraw-Hill Book Company, 1985. page 32–35.

[24] International Monetary Fund. Financial system abuse, financial crime and money laundering — background paper. `http://www.imf.org/external/np/ml/2001/eng/021201.pdf`, February 2001.

[25] Graham Kirsch. Active memory: Micron's Yukon. In *17th International Parallel and Distributed Processing Symposium (IPDPS)*, page 89. IEEE Computer Society, 2003.

[26] R. Kohavi. A study of cross-validation and bootstrap for accuracy estimation and model selection. In *Proc. of the 14th IJCAI*, pages 1137–1143, 1995.

[27] John R. Koza, David Andre, Forrest H Bennett III, and Martin Keane. *Genetic Programming 3: Darwinian Invention and Problem Solving*. Morgan Kaufmann Publishers, San Fransisco, CA, Apr 1999.

[28] John R. Koza, Martin A. Keane, Matthew J. Streeter, William Mydlowec, Jessen Yu, and Guido Lanza. *Genetic Programming IV: Routine Human-*

Competitive Machine Intelligence. Kluwer Academic Publishers, 2003. ISBN 1-4020-7446-8.

[29] Christos Kozyrakis and David Patterson. Overcoming the limitations of conventional vector processors. In *30th International Symposium on Computer Architecture (ISCA)*, pages 399–409. IEEE Computer Society, Jun 2003. ISBN 0-7695-1945-8.

[30] Eric C Lai. Predicting and validating microRNA targets. *Genome Biol.*, 5(9):115, 2004.

[31] Jim W. Lindelien. The value of accelerated computing in bioinformatics. See `http://www.timelogic.com/whitepapers/decypher_benefits_e.pdf`, 2002.

[32] Yexin Liu and Mauricio D. Sacchi. Propagation of borehole derived properties via a support vector machine. *CSEG recorder*, pages 54–58, December 2003.

[33] R. Mei, E. Hubbell, S. Bekiranov, M. Mittmann, F.C. Christians, M.-M. Shen, G. Lu, J. Fang, W.-M. Liu, T. Ryder, P. Kaplan, D. Kulp, and T.A. Webster. Probe selection for high-density oligonucleotide arrays. *Proc. Natl. Acad. Sci. U.S.A.*, 100(20):11237–11242, 2003.

[34] O.M. Melko and A.R. Mushegian. Distribution of words with a predefined range of mismatches to a DNA probe in bacterial genomes. *Bioinformatics*, 20(1):67–74, 2004.

[35] Gonzalo Navarro. NR-grep: a fast and flexible pattern matching tool. *Software Practice and Experience (SPE)*, 31:1265–1312, 2001.

[36] David Patterson, Thomas Anderson, Neal Cardwell, Richard Fromm, Kimberly Keeton, Christoforos Kozyrakis, Randi Thomas, and Katherine Yelick. A case for intelligent RAM: IRAM. *IEEE Micro*, 17(2):34–44, Apr 1997.

[37] Knut Magne Risvik. *Scaling Internet Search Engines: Methods and Analysis*. PhD thesis, NTNU, Trondheim, Norway, May 2004.

[38] Pål Sætrom. Predicting the efficacy of short oligonucleotides in antisense and RNAi experiments with boosted genetic programming. *Bioinformatics*, 20(17):3055–3063, 2004.

[39] R.K. Saiki, S. Scharf, F. Faloona, K.B. Mullis, G.T. Torn, H.A. Erlich, and N. Arnheim. Enzymatic amplification of beta-globin genomic sequences and restriction site analysis for diagnosis of sickle cell anemia. *Science*, 230(4732):1350–1354, 1985.

[40] Sandeep Saxena, Zophonías O. Jónsson, and Anindya Dutta. Implications for off-target acitivity of small inhibitory RNA in mammalian cells. *J. Biol. Chem.*, 278(45):44312–44319, 2003.

[41] Peter C. Scacheri, Orit Rozenblatt-Rosen, Natasha J. Caplen, Tyra G. Wolfsberg, Lowell Umayam, Jeffrey C. Lee, Christina M. Hughes, Kalai Selvi Shanmugam, Arindam Bhattacharjee, Matthew Meyerson, and Francis S. Collins. Short interfering RNAs can induce unexpected and divergent changes in the levels of untargeted proteins in mammalian cells. *Proc. Natl. Acad. Sci. U.S.A.*, 101(7):1892–1897, 2004.

[42] Lisa J. Scherer and John J. Rossi. Approaches for the sequence-specific knock-down of mRNA. *Nat. Biotechnol.*, 21(12):1457–1465, 2003.

[43] Temple F. Smith and Michael S. Waterman. Identification of common molecular subsequences. *J. Mol. Biol.*, 147(1):403–410, 1981.

[44] Ola Snøve Jr. and Torgeir Holen. Many commonly used siRNAs risk off-target activity. *Biochem. Biophys. Res. Commun.*, 319(1):256–263, 2004.

[45] Stuart Staniford, Vern Paxson, and Nicholas Weaver. How to 0wn the Internet in Your Spare Time. In *11th USENIX Security Symposium*, August 2002.

[46] Mervyn Stone. Cross-validatory choice and assessment of statistical predictions. *Journal of the Royal Statistical Society. Series B (Methodological)*, 36(2):111–147, 1974.

[47] M. Turhan Taner. Seismic attributes. *CSEG Recorder*, pages 48–56, September 2001.

[48] Aad J. van der Steen and Jack J. Dongarra. Overview of recent supercomputers. Technical report, EuroBen, October 2004. This 14th issue of the annual report is available from `http://www.top500.org/ORSC/2004/`.

[49] Wesley K. Wilhelm. Payment card fraud in a chip card world. `http://www.fairisaac.com/NR/rdonlyres/7CE35A4B-96B0-43A0-9503-78BDAC483%510/0/PaymentCardFraudWP.pdf`, March 2003.

[50] Tomoyuki Yamada and Shinichi Morishita. Accelerated off-target search algorithm for siRNA. *Bioinformatics*, page bti155, 2004.

Paper III

Sequence Explorer: interactive exploration of genomic sequence data

Sequence Explorer: interactive exploration of genomic sequence data

Ola Snøve Jr. [a] Håkon Humberset [a] Olaf René Birkeland [a]
Pål Sætrom [a,*],

[a] *Interagon AS, Medisinsk teknisk senter, NO-7489 Trondheim, Norway*

Abstract

Current solutions for complex motif searching in DNA and protein sequences are not interactive as users usually wait tens of seconds before the results can be viewed. We propose a hardware-accelerated client-server solution that is fast enough to retain the interactive feeling even when screening whole genomes.

We structured our framework for interactive sequence analysis around query, dataset, filter, and result presentation modules. The query and dataset specification enable simultaneous, interactive screening of multiple complex queries against several datasets. The filters impose restrictions such as only allowing hits to be reported if they occur in coding regions, and the different result presentations include histograms and hit lists.

Our results show that interactive searching is possible even though response times vary significantly depending on filter, network bandwidth and hit frequencies. With a relatively small server, we obtain response times of about one and a half second on gigabytes of data when queries are sufficiently specific to avoid network bottlenecks due to high hit frequencies.

Key words: siRNA, RNA interference, efficacy prediction

Searching RNA, DNA, and protein sequence data usually means looking for similarities between a query sequence and some database. The Smith-Waterman algorithm [12] is the most sensitive algorithm, but it is very CPU intensive, which is why several heuristic approaches have emerged with great success.

* Corresponding author. Fax: +47 455 94 458
Email addresses: ola.snove@interagon.com (Ola Snøve Jr.),
haakon.humberset@interagon.com (Håkon Humberset),
olaf.rene.birkeland@interagon.com (Olaf René Birkeland),
paal.saetrom@interagon.com (Pål Sætrom).

Notable heuristics include FASTA [8], BLAST [1], ParAlign [9], and PatternHunter [6]. Similarity search algorithms may, however, be too advanced when searching for occurrences and distributions of simple motifs such as repetitive sequences or transcription factor binding sites in genomic data. Regular expressions are widely used for pattern matching in text [3], and specialized versions of the familiar algorithms have been proposed for DNA and protein sequences [11]. Betel and Hogue showed that low-level pattern matching was valuable in identifying genetic targets in a cancer characterized by a high frequency of mutations in coding regions containing mononucleotide repeats [2].

Due to novel algorithms, improved implementations, and increased CPU speed, similarity search algorithms now have acceptable response times when running on large publicly funded and freely available supercomputers. Pattern matching algorithms are also impressive for some purposes, but the process is still not interactive when screening for complex patterns in large volumes of sequence data. We hypothesize that many ideas do not realize their full potential because biologists can not (i) query sequence databases with biological questions; (ii) view the results at the appropriate abstraction level; and (iii) explore the search space and develop their hypotheses interactively. For example, when searching the genome looking for disease genes that display some sequence features found in a family of known genes, a researcher may initially want to focus on genomic positions that are close to known promoter loci. Furthermore, comparing high-level results such as hit rate distributions from several different genomes might be valuable. Finally, observing the results while varying the search parameters such as distance bounds and fuzziness may reveal important differences between genomes.

We have developed a client-server solution that aims to provide interactive searches at different abstraction levels. The client's graphical user interface consists of four main panes that correspond to pattern, filter, dataset, and result specifications. A common feature for all panes is that layered specifications is possible; that is, the user can specify multiple questions and result views that will be executed simultaneously. A screening against all specified datasets is performed whenever the queries change, which means that result differences due to changes in query parameters are observed almost instantaneously. As queries are automatically scheduled for execution when they are constructed, we have removed the familiar "submit" button because we felt that it would prevent the application from being truly interactive. To avoid excessive scheduling, we have introduced a quiet time frame from the last character entry to query submission. Furthermore, to maintain interactivity, an ongoing search is automatically aborted if its results have not been reported at the time its corresponding query is altered in the client. The user may pose restrictions on queries by adding filters such as requiring that only hits in coding regions should be reported. We aim for result presentations that

enable researchers to quickly grasp the important information without being distracted by annotation that is only needed if the results justify careful investigation. That being said, the application provides linkouts to annotation from the region surrounding individual hits.

We use special purpose search processors on PCI search cards to accelerate standard workstations for pattern-matching purposes. The high performance of these processors is critical to get interactive searches in gigabytes of data. Each chip can screen anything from one to sixty-four patterns against 100 MB depending on the queries's complexity [4]. A single chip is three orders of magnitude faster for regular expression-searching than "nr-grep" [7] running on a 1 GHz Pentium III with 256 MB of memory [5]. Furthermore, there are sixteen chips with local memory on the search card, which means that the theoretical throughput of each card is 1.6 GB per second.

The search processor is designed to match fuzzy patterns in arbitrary data. We have developed the Interagon Query Language (IQL) that uses regular expression-like syntax to take advantage of the search processor's functionality. Although similar to regular expressions, the language has features not feasible in software. Especially, this is the case for *n of m* expressions; that is, "match n out of m subparts" where the latter can be everything from a single sequence of characters to complex patterns defined by the language. Moreover, the possibility of specifying that two patterns of arbitrary complexity should be separated by some length, or be present in a specific order, is useful. (See the tutorial in the supplementary information for practical examples on using IQL for interactive exploration of DNA sequence data.) IQL is neither specialized for DNA searching nor optimized with respect to this particular application, but should have sufficient functionality to illustrate the potential of interactive searching. The language does, for example, easily support both Prosite patterns and position weight matrices.

Our results show that we can interactively search entire genomes by using special-purpose pattern-matching hardware to accelerate a standard workstation. Typical response times are a few seconds depending on the query complexity. Because of network transfer limitations, simple patterns with high hit rates get high response times. To reduce this negative effect, we do not report all the results for queries with high hit rates. Simple patterns are however seldom very informative, and we therefore question whether this limitation has practical consequences. Furthermore, filters sometimes hamper the performance, especially if there are many hits being post processed. We are currently working on finding ways to implement faster filters.

We believe that our interactive search tool will be valuable for iterative hypothesis testing and refinement. By integrating a machine learning algorithm that automatically creates pattern hypotheses (for example, [10]), researchers

can also investigate problems where they only have some qualitative description of the desired solution, and thus cannot formulate the initial pattern hypothesis themselves. The interactive tool can then be used to further investigate or refine the pattern hypotheses generated by the machine learning algorithm. We are currently investigating this approach.

Supplementary information

A Java client querying a hosted server is freely available upon request at `http://www.interagon.com/demo/`. Four chromosomes are available in this demo version of the server. In addition to the demo application, a tutorial describing system requirements, installation, and use of the program is available along with a technical note on the Interagon Query Langauge.

Acknowledgements

We thank H.E. Krokan, F. Drabløs, A. Halaas, T.B. Grünfeld, S.H. Fjeldstad, and M. Nedland for valuable help during the development of this demo. The work was supported by the Norwegian Research Council, grants 151899/150 and 151521/330, and the bioinformatics platform at the Norwegian University of Science and Technology, Trondheim, Norway.

References

[1] Stephen F. Altschul, Warren Gish, Webb Miller, Eugene W. Myers, and David J. Lipman. Basic local alignment search tool. *J. Mol. Biol.*, 215 (3):403–410, 1990.

[2] Doron Betel and Christopher W.V. Hogue. Kangaroo - a pattern-matching program for biological sequences. *BMC Bioinformatics*, 3(1): 20–22, 2002.

[3] Jeffrey E.F. Friedl. *Mastering Regular Expressions*. O'Reilly, Cambridge, MA, 2nd edition, 2002.

[4] Arne Halaas, Børge Svingen, Magnar Nedland, Pål Sætrom, Ola Snøve Jr., and Olaf Renè Birkeland. A recursive MISD architecture for pattern matching. *IEEE Trans. on VLSI Syst.*, 12(7):727–734, 2004.

[5] Magnus Lie Hetland and Pål Sætrom. A comparison of hardware and software in sequence rule evolution. In *Eight Scandinavian Conference on Artificial Intelligence*, 2003.

[6] Bin Ma, John Tromp, and Ming Li. PatternHunter: faster and more sensitive homology search. *Bioinformatics*, 18(3):440–445, 2002.

[7] Gonzalo Navarro. NR-grep: a fast and flexible pattern matching tool. *Software Practice and Experience (SPE)*, 31:1265–1312, 2001.

[8] William R. Pearson and David J. Lipman. Improved tools for biological sequence comparison. *J. Mol. Biol.*, 85(8):2444–2448, 1988.

[9] Torbjørn Rognes. ParAlign: a parallel sequence alignment algorithm for rapid and sensitive database searches. *Nucleic Acids Res.*, 29(7):1647–1652, 2001.

[10] Pål Sætrom. Predicting the efficacy of short oligonucleotides in antisense and RNAi experiments with boosted genetic programming. *Bioinformatics*, 20(17):3055–3063, 2004.

[11] S.S. Sheik, Sumit K. Aggarwal, Anindya Poddar, N. Balakrishnan, and K. Sekar. A fast pattern matching algorithm. *J. Chem. Inf. Comput. Sci.*, 44(4):1251–1256, 2004.

[12] Temple F. Smith and Michael S. Waterman. Identification of common molecular subsequences. *J. Mol. Biol.*, 147(1):403–410, 1981.

Paper IV

Predicting non-coding RNA genes in *Escherichia coli* with boosted genetic programming

Nucleic Acids Research, 2005, Vol. 33, No. 10 ***3263–3270***
doi:10.1093/nar/gki644

Predicting non-coding RNA genes in *Escherichia coli* with boosted genetic programming

Pål Sætrom*, Ragnhild Sneve[1], Knut I. Kristiansen[1], Ola Snøve Jr., Thomas Grünfeld, Torbjørn Rognes[1] and Erling Seeberg[1]

Interagon AS, Medisinsk teknisk senter, NO-7489 Trondheim, Norway and [1]Centre for Molecular Biology and Neuroscience, Institute of Medical Microbiology, Rikshospitalet University Hospital, NO-0027 Oslo, Norway

Received November 24, 2004; Revised February 22, 2005; Accepted May 20, 2005

ABSTRACT

Several methods exist for predicting non-coding RNA (ncRNA) genes in *Escherichia coli* (*E.coli*). In addition to about sixty known ncRNA genes excluding tRNAs and rRNAs, various methods have predicted more than thousand ncRNA genes, but only 95 of these candidates were confirmed by more than one study. Here, we introduce a new method that uses automatic discovery of sequence patterns to predict ncRNA genes. The method predicts 135 novel candidates. In addition, the method predicts 152 genes that overlap with predictions in the literature. We test sixteen predictions experimentally, and show that twelve of these are actual ncRNA transcripts. Six of the twelve verified candidates were novel predictions. The relatively high confirmation rate indicates that many of the untested novel predictions are also ncRNAs, and we therefore speculate that *E.coli* contains more ncRNA genes than previously estimated.

INTRODUCTION

Non-coding RNAs (ncRNA) are transcripts, whose function lies in the RNA sequence itself and not as information carriers for protein synthesis. Although long believed to be a minor gene class, recent discoveries have revealed that ncRNA genes are far more prevalent than previously believed and that they have other important roles beyond protein synthesis (rRNA and tRNA) (1–5).

In *Escherichia coli*, the number of experimentally verified small RNA (sRNA) genes (ncRNA genes excluding rRNA and tRNA) has increased rapidly. Only 10 sRNA genes were known in 1999 (6), whereas a recent survey listed 55 known sRNA genes (7). Subsequent RNA cloning experiments increased the number of known sRNA genes to 62 (8).

Most of these sRNA genes were identified in six studies describing systematic searches for new sRNA genes (9–14). All but one of these studies (14) used computational methods to predict sRNA genes. The computational methods ranged from analysis of sequence (9,10) and structure (11) conservation; to promoter and terminator identification (9,13); and machine learning based on sequence composition, known ncRNA motifs and RNA secondary structure stability (12). Together, these six studies have predicted ~1000 non-redundant sRNA candidates that are yet to be confirmed (7). Note, however, that only 95 candidates were predicted by more than one study.

We describe a method that uses automatic discovery of sequence patterns to predict ncRNA genes in *E.coli*'s intergenic regions. The main strengths of the method as compared to other methods are that (i) it uses the DNA sequence directly as input, which helps to reduce any potential bias from input feature selection and encoding (12); (ii) it works well with a much larger number of intergenic sequences (negative examples) than known ncRNA sequences (positive examples) (12); (iii) it is very robust when it comes to noise in the training data, as for instance intergenic regions that actually are ncRNAs; and (iv) it does not rely on sequence conservation to predict ncRNA genes.

The method predicts several hundred intergenic regions to contain ncRNA genes, and over half of these overlap with previous predictions. We test the 10 top-scoring candidates and verify 9 of these by northern analysis. In addition, we test six candidates of varying prediction confidence; three of these are confirmed by northern analysis. Only 6 of these 12 new ncRNA genes have been predicted by previous methods.

Our results indicate that the number of ncRNA genes in *E.coli* is larger than what has previously been estimated (15). This is because the estimates of Zhang and colleagues were partly based on the number of ncRNA genes predicted by more than one method, which, until now, was 95. We have extended

*To whom correspondence should be addressed. Tel: +47 9820 3874; Fax: +47 4559 4458; Email: paal.saetrom@interagon.com

The authors wish it to be known that, in their opinion, the first two authors should be regarded as joint First Authors

this list by 44%, which is a significant increase. In addition, we have shown that our method detects ncRNA genes that have not been predicted by other methods.

MATERIALS AND METHODS

Sequence data

We downloaded the *E.coli* K-12 genome sequence (16) (U00096.1) and its annotations (release 73) from EMBL's FTP server (http://www.ebi.ac.uk/genomes/bacteria.html). Based on annotations and previous studies (9–11), we collected a set of 154 experimentally verified ncRNA sequences. These sequences consisted of 86 tRNAs, 22 rRNAs and 46 other sRNA genes. Note that one of these sRNAs was the strain-dependent uptR gene (17). The list of ncRNA sequences is given in the Supplementary Material.

Based on the positions of known ncRNA genes and protein coding sequences (CDS), we constructed a set of intergenic sequences (INT) by removing all parts of the genome containing ncRNAs and CDSs, along with 100 nt on each side. This resulted in 942 subsequences totaling 144 520 nt, which increased to 1884 sequences of 289 040 nt when we added the complement of each sequence.

Each ncRNA and INT sequence was then divided into 50 nt sequence windows with 25 nt overlap. If the final window in a sequence had <50 nt, we adjusted the overlap so that the final window also had 50 nt. For example, 90 nt sequences were divided into three 50 nt sequence windows consisting of nucleotides 1–50, 26–75 and 41–90. The 50 nt window size was chosen because the smallest ncRNA in our dataset was 53 nt (dicF). This procedure gave 1795 ncRNA sequence windows and 10 663 INT sequence windows; removing duplicates in the form of identical sequences reduced the number of ncRNA and INT sequence windows to 840 and 10 572. Of the 840 unique ncRNA sequence windows, 53% were from rRNAs, 30% from sRNAs and 17% from tRNAs.

Algorithms

We use a machine learning algorithm called GPboost$_{\mathrm{Reg}}$ to create classifiers that predict whether or not a sequence is an ncRNA gene. The algorithm has previously been used to predict the efficacy of short oligonucleotides in RNAi and antisense experiments (18,19). In the following, we will only give a basic description of the algorithm; interested readers should consult Sætrom (18) and the references therein for a complete description.

GPboost$_{\mathrm{Reg}}$ takes as input a set of positive and negative sequences and creates a classifier that predicts whether or not an unknown sequence belongs to the positive set. Here, the positive and negative sequences are the ncRNA and INT sequence windows described in the previous section. Thus, the classifier created by GPboost$_{\mathrm{Reg}}$ can predict whether or not a given sequence comes from an ncRNA.

To create the classifiers, GPboost$_{\mathrm{Reg}}$ combines genetic programming (GP) (20) and boosting algorithms (21). GP uses simulated evolution in a population of candidate solutions to solve problems, and here, each individual in the population is an expression in a formal query language (whitepaper available on request). GP evaluates how well each candidate solution separates between the positive and negative sequences and uses this fitness information to guide the simulated evolution. That is, our GP solution iteratively (i) selects candidate solutions based on fitness such that more fit solutions have a higher chance of being selected; (ii) introduces random changes in the selected solutions by exchanging subparts of two candidate solutions (crossover) or randomly changing a subpart of a candidate solution (mutation); and (iii) updates the solution population by replacing the old population with the randomly changed candidate solutions. We repeat this process a fixed number of iterations and choose, as the final solution of the GP run, the candidate solution that gave the best performance on the training set.

The classifiers created by our GP algorithm are sequence patterns that can only give binary answers. That is, given a sequence, each pattern answers either 'yes' (1) or 'no' (-1), as to whether the pattern matches parts of the sequence or not. To improve the confidence of our predictions, we combine the GP algorithm with a boosting algorithm. Boosting algorithms join several classifiers into a final weighted average of the individual classifiers such that the performance of the final classifier is increased compared to each of the single classifiers. To do this, the boosting algorithm guides each GP run's search for good solutions by adjusting the relative importance of each sequence in the training set. Then the boosting algorithm assigns a weight to the best expression from the GP run. This weight is based on the expression's performance in the corresponding training set and is assigned such that the output of the final classifier ranges from -1 to 1. As a result, the classifiers created by our algorithm are the weighted average of several different sequence patterns. We will occasionally refer to these classifiers as models. Note that GPboost$_{\mathrm{Reg}}$ uses regularized boosting (22) to handle noise in the training set.

To reduce the time needed to evaluate each individual expression in the GP population, we use a special purpose search processor designed to provide orders of magnitude higher performance than comparable regular expression matchers (23). The increased performance becomes important when the datasets are large, or when many expressions must be evaluated, for instance, in cross-validation experiments or when GP is used as the base learner in a boosting algorithm.

Quality measures

When a model is evaluated on a positive and negative set of sequences, four statistics (counts) can be defined: the number of true positives (*TP*), false positives (*FP*), true negatives (*TN*) and false negatives (*FN*). These represent the positive hits in the positive set, positive hits in the negative set, negative hits in the negative set and negative hits in the positive set, respectively. Several quality measures can be defined from these counts (24). This study uses the Matthews correlation *M* (Equation 1), false positive rate FP_p (Equation 2) and sensitivity *Se* (Equation 3):

$$M = \frac{FP \cdot TN + FP \cdot FN}{\sqrt{(TN + FN) \cdot (TN + FP) \cdot (TP + FN) \cdot (TP + FP)}} \quad \mathbf{1}$$

$$FP_p = \frac{FP}{FP + TN} \quad \mathbf{2}$$

$$Se = \frac{TP}{TP + FN} \quad \mathbf{3}$$

Strain and growth conditions

Escherichia coli K-12 strain MG1655 cells (from overnight cultures were diluted 1/50 in Luria–Bertani (LB) medium and subsequently grown at 37°C) were grown in LB broth and used for inoculation of liquid cultures. Cells were grown in 100-ml batch cultures in 500-ml Erlenmeyer flasks at 37°C with aeration by rotary shaking (250 r.p.m.). The culture media used was LB as described elsewhere (25). Growth was monitored at 600 nm on a Shimadzu UV-1601 UV-visible spectrophotometer. Cells were harvested in four different growth phases: lag ($OD_{600} < 0.2$), log ($0.2 < OD_{600} < 1.0$), early stationary ($1.0 < OD_{600} < 2.0$) and late stationary phase ($OD_{600} > 2.0$).

RNA isolation

Total RNA was isolated from the cells using a procedure based on trizol reagent combined with RNeasy microcolumns (Qiagen). One milliliter of trizol was added per 10^6 cells and stored at room temperature for 5 min; 0.2 µl chloroform was added per ml of trizol and the sample was shaken for 15 s. The sample rested before centrifugation for 15 min at 12000 *g* and 4°C. The aqueous phase was slowly added 1:1 to 70% EtOH to avoid precipitation. The sample was further loaded to the RNeasy column and washed and DNase treated according to the RNeasy protocol (Qiagen). Isolated RNA was resuspended in RNase-free water and quantitated using Eppendorf BioPhotometer.

Oligonucleotides

The complete list of oligonucleotides used to generate probes for northern analysis and primer extension experiments is provided as Supplementary Material.

Northern analysis

RNA samples (~10 µg) were denatured for 10 min at 60°C in a buffer containing 95% formamide, separated on urea–polyacrylamide (8%) gels, and transferred to nylon membranes by electroblotting. Radiolabeled strand-specific RNA probes were synthesized using *in vitro* transcription according to MAXIscript™ (Ambion). Hybridization signals were visualized on Typhoon 9410 (Amersham).

Primer extension assay

Primer extension assay was carried out with AMV reverse transcriptase (Promega), on ~10 µg total RNA and 5′ end-labeled primers. The primers were end-labeled by using [γ^{32}-P]ATP and polynucleotide kinase. Products of the extension reactions were separated on 8% polyacrylamide sequencing gels alongside sequencing reactions performed on the corresponding PCR products from the intergenic regions. Sequencing reactions were carried out with a Thermo Sequenase Radiolabeled Terminator Cycle Sequencing Kit (USB, Amersham).

RESULTS

ncRNA gene predictions

We used a variant of 10-fold cross-validation to train and test our machine learning algorithm (26,27). More specifically, we randomly divided the sets of ncRNA and INT sequence windows into 10 non-overlapping subsets. Then, we iteratively trained classifiers on 8 of the subsets and tested the classifiers on the remaining 2 subsets. We used one of these test subsets to estimate the optimal value of the regularization parameter in the $GPboost_{Reg}$ algorithm and the other test subset as a completely independent test set. We ran this training and testing procedure for 10 iterations such that all the 10 subsets had been used as the independent test set.

To estimate the optimal regularization value, we tried several different values and used the one with the highest average correlation in the 10 'parameter estimation' test subsets. These optimal models had an average correlation of 0.58 on the complete test set, and predicted on average 22 false positive sequence windows in the test subsets. This resulted in an average false positive rate of 2.1%. The models' average sensitivity was 54%. The following sections will examine the predictions in the original ncRNA set, the true positives and false negatives, and the potential new ncRNA genes, the false positives.

The algorithm identifies nearly 80% of the sRNAs in the database. As we used two subsets to test the classifiers, there was some overlap between each of the test sets (each unique sequence was present in two different test sets for two different models). The test set consisted of 840 unique sequences for a total of 1680 sequences: 913 of these were predicted as true positives and 767 were false negatives. When duplicates were removed from these sets, 564 of 840 were positive predictions and 491 of 840 were negative predictions. In other words, 215 sequences were predicted as being both positive and negative. This means that 42% of the sequences were strongly predicted by two models, and 26% were weakly predicted by a single model.

Two of 46 sRNA sequences were completely matched by the models and 10 were completely missed. The complete matches were the partially overlapping rydB and tpe7 found by Wassarman *et al*. (10) and Rivas *et al*. (11), and the misses were micF, oxyS, rybB, ryeE, ryhA, spf, sraB and sraE, and the overlapping ryhB and sraI found by Wassarman *et al*.(10) and Argaman *et al*. (9).

306 potential new ncRNA genes of which 152 confirm previous predictions. The models predicted a total of 438 false positive sequence windows; 57 of these were predicted by two models. Several of the predicted sequence windows overlapped or were located next to each other. When these were joined and treated as one continuous sequence, a total of 306 sequences remained.

A cross-reference of the 306 candidate ncRNA sequences with the list of predicted but unconfirmed ncRNA genes presented in (7) identified that 171 of the sequences overlapped with previous predictions; 152 of these were predicted to be on the same strand. Most of the predictions overlapped with the predictions of Carter and colleagues (12). This was expected, not only because their predictions were the most abundant in our INT set, but also because they base their predictions on the common sequence characteristics of ncRNAs, which is also the essence of our method.

Accounting for the number of predictions made by other methods that were significantly represented (>10 sequences) in our INT set, our predictions support 35, 51, 28 and 41% of the predictions of Rivas *et al*. (11), Carter *et al*. (12),

Chen *et al*. (13) and Tjaden *et al*. (14). Thus, there is relatively good correspondence between our predictions and the predictions of these four methods.

Our results confirm several previous predictions that were not supported by other methods. In total, the intergenic regions in our dataset contained 288 sequences that have been predicted by only one previous method to be part of an ncRNA gene. Our predictions overlapped 123 of these 288 sequences. Excluding the predictions that were unique to the Carter algorithm, our predictions supported 42 of the remaining 166 sequences. Thus, although our predictions increased the list of candidates that are unique to a single study by 15%, we increased the list of candidates predicted by more than one study from 95 to 218 (7). Even when excluding the Carter specific sequences, we increased the list of candidates predicted by more than one study by 44% (7). This is a significant increase.

Table 1 shows the 10 highest scoring intergenic sequence windows (the complete list of predictions are available as Supplementary Material). The table is sorted according to the model output for the highest predicted window in the sequence.

After we started our experiments, several new ncRNA genes in *E.coli* have been identified. Table 2 lists the ncRNA genes that were not included as known ncRNAs in our training set, but that were included with at least 50 nt in our set of intergenic sequences. That is, they were falsely included as negative sequences in the training set. The genes were mainly collected from the *E.coli* genome project's (www.genome.wisc.edu) ASAP database (28) (*E.coli* K-12 Strain MG1655 version m54) and from Refs (7,8).

Although, as Table 2 shows, our method only predicts 2 of the 10 genes to be on the correct strand, the performance is not poorer than that of other methods. For instance, the method of Carter and colleagues (12), which is comparable to our method, predicts only one gene (SroG) correctly. Thus, these genes may be too different to be predictable without combining several of the available methods.

We also cross-referenced our predictions with the unconfirmed transcripts in the cDNA library of Vogel *et al*. (8). Table 3 lists the transcripts that were included with at least 50 nt in our set of intergenic sequences. As the table shows, we predict 5 of the 7 transcripts to be ncRNA genes with the correct orientation. Again, our predictions are comparable to or slightly better than other methods.

Finally, Kawano *et al*. (29) describes several new ncRNA genes. Not all these new ncRNAs were present in our dataset; of the three genes that were present, our predictions match one (RyfB). The other two genes (SokE and SokX), like rdlA, rdlB, rdlC and rdlD, may be involved in anti-sense regulation of hok and ldr (29–31). As these ncRNAs' function is closely linked to their targets' sequences, they may not share many sequence characteristics with other ncRNAs. This can explain why our method has problems predicting these hok/ldr-related ncRNAs.

Table 1. Top ten predictions sorted by prediction confidence

ID	Position	Length	Strand	Score	Annotation
I001	271879	100	+	0.22	271880–272035 + Carter *et al*.
I002	4230937	150	−	0.22	4230927–4231086 − Carter *et al*.
I003	719883	75	+	0.21	719854–719973 + Carter *et al*.
I004	3766615	50	+	0.21	*Novel*
I005	303544	50	−	0.19	*Novel*
I006	262270	82	−	0.18	*Novel*
I007	4626216	75	+	0.17	*Novel*
I008	1702671	75	+	0.16	1702604–1702818 + Tjaden *et al*.
I009	1859481	125	+	0.16	1859567–1859646 + Carter *et al*.
I010	4527911	50	+	0.15	4527862–4527941 + Carter *et al*.

The given position is the 5′ end for predictions in the positive strand, and the 3′ end for predictions in the negative strand. The score is the classifier output for the highest scoring sequence window in a sequence.

Table 2. Known ncRNA genes included in the set of intergenic sequences

Gene	Overlap	Strand	Prediction	Previous predictions (7)
C0067 (12)	60 of 124	+	Not predicted	n/a
rdlA (30)	66 of 66	+	Predicted 50 nt (−)	?(11), − (12)
rdlB (30)	65 of 65	+	Not predicted	? (11), − (12)
rdlC (30)	67 of 67	+	Not predicted	? (11), − (12)
IS061 (13)	60 of 157	−	Not predicted	n/a
IS092 (13)	116 of 159	−	Not predicted	n/a
rygC (10)	76 of 150	+	Predicted 50 nt (+ and −)	+ (13), − (12)
SroG (8)	110 of 147	−	Predicted 89 nt (−)	− (12)
rdlD (30)	63 of 63	+	Not predicted	− (14), − (12)
SroH (8)	61 of 159	−	Not predicted	+ (13)

The overlap is the number of nucleotides from the ncRNA included as an intergenic sequence. The last column lists the strand and the reference to previous predictions overlapping the gene.

ncRNA gene validations

To test our predictions, we selected 16 predictions for experimental validation. These included all the top 10 predictions from Table 1 and 6 additional predictions with varying prediction confidence (summarized in Table 4). We chose the 6

Table 3. Unconfirmed transcripts from (8) included in the set of intergenic sequences

Contig	Overlap	Strand	Prediction	Previous predictions (7)
Contig_440	68 of 105	+	Predicted 50 nt (+) and 50 nt (−)	+ (13), − (12)
Contig_68	76 of 157	+	Predicted 49 nt (+)	+ (14), − (13)
Contig_606	83 of 103	+	Predicted 63 nt (+) and 50 nt (−)	+ (14), − (12)
Contig_223	80 of 141	−	Predicted 50 nt (−)	− (12)
Contig_496	73 of 73	+	Predicted 61 nt (+) and 49 nt (−)	− (14), ± (12)
Contig_286	102 of 102	+	Predicted 50 nt (−)	+ (14)
Contig_181	43 of 43	−	Not predicted	? (11), + (13)

See Table 2 for header explanations.

Table 4. Six predictions with varying confidence experimentally tested in the lab

ID	Position	Length	Strand	Score	Annotation
I014	4373943	60	−	0.14	*Novel*
I016	1218274	50	−	0.14	*Novel*
I035	914278	100	+	0.1	914218–914571 ± Rivas *et al*. 914259–914378 + Carter *et al*.
I044	4366175	50	+	0.1	*Novel*
I209	4006562	50	+	0.025	4006513–4006565 − Carter *et al*.
I211	214141	50	−	0.025	*Novel*

See Table 1 for details on the prediction position.

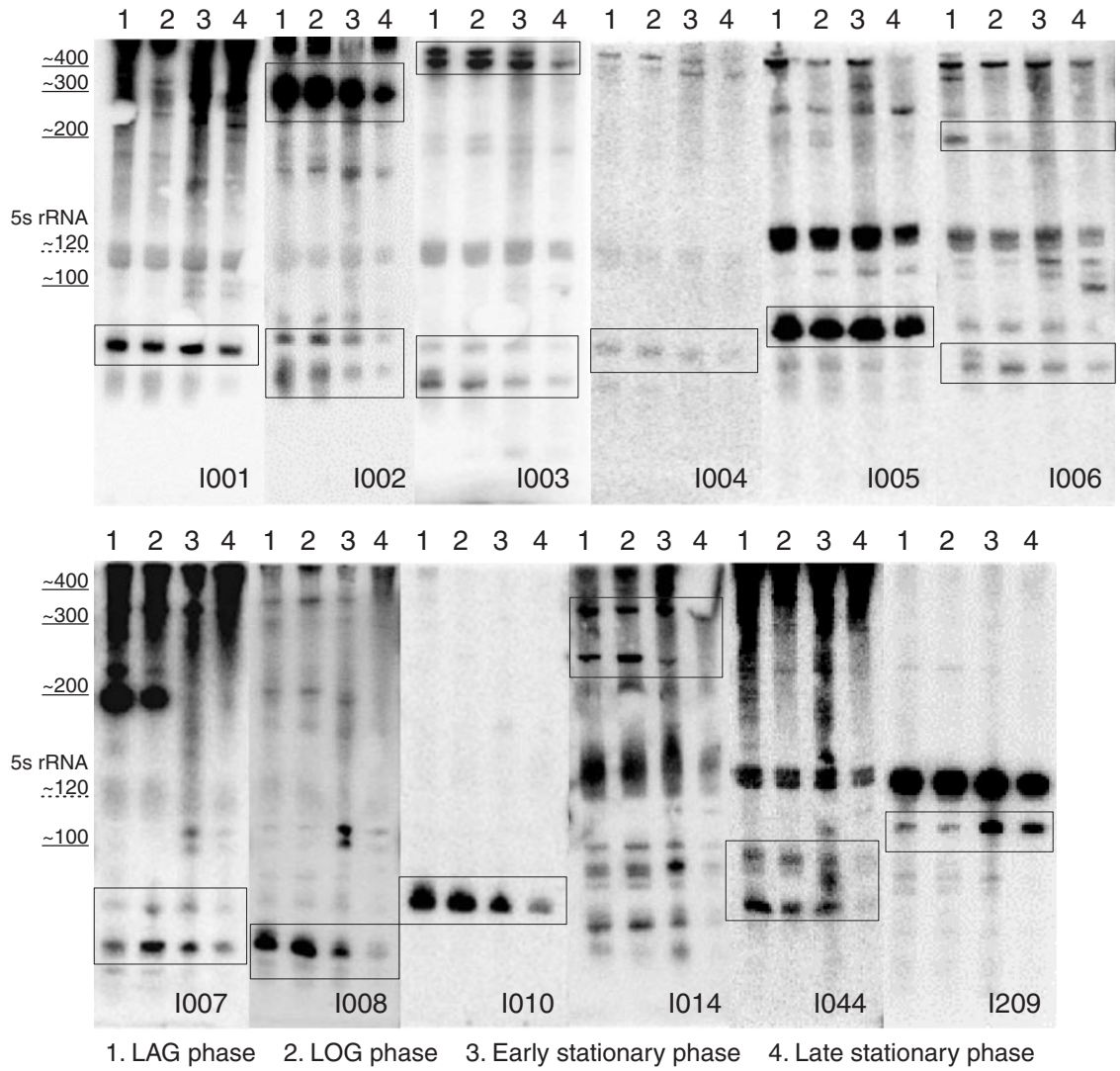

Figure 1. Northern hybridizations of selected predictions against total RNA from lag, log, and early and late stationary phases confirm 12 of 16 selected transcripts. The figure shows the complete northern blots after low stringency wash. The boxed bands indicate the bands that were still present after repeated washes of higher stringency, but the resulting blots are excluded because of poor resolution and picture quality. The indicated sizes are only approximate sizes because these are individual blots lined up together; see Supplementary Figure 2 for size estimates based on each individual blot. Note that most blots have a ∼120 nt band that corresponds to 5s rRNA.

additional predictions to have both high and low prediction confidence, and to be a mix of previously predicted and novel candidates. These 6 additions represented a more varying spectrum of predictions than did the top 10 predictions.

Figure 1 shows the results of northern hybridization with strand-specific probes from 12 of the 16 predictions against total RNA from the *E.coli* lag, log, and early and late stationary phases (see Materials and Methods). Most of the 12 confirmed transcripts were differentially expressed in the four phases, which is in agreement with previously known ncRNAs in *E.coli* (8–10). We did not detect transcripts from the four predictions not shown in Figure 1 (data not shown). The absence of detectable transcripts do, however, not imply that the predictions are wrong as some ncRNAs are only expressed under certain conditions [see for example (2,8,10)]. We also tried to map the 5′ start of 4 of the 12 verified transcripts (I001, I002, I004 and I014, chosen because these were a mix of high and low confidence, and previous and novel predictions). We identified potential 5′ start sites for all four transcripts (see Supplementary Material). Based on these results, we estimated the size of three of the transcripts; see Table 5 for additional information.

As Figure 1 shows, we detected more than one band for six of the predictions. These instances of multiple bands were either (i) a large sequence with one or two additional smaller sequences (I002, I003 and I006); (ii) two large sequences (I014); or (iii) two small sequences (I007 and I044). One possible explanation is that the multiple bands are processed or degraded forms of a single transcript. This may be the case for I002 and I014, as we saw only one 5′ start point for each region in the primer extension. These transcripts could be specifically processed by catalytically active enzymes,

Table 5. Transcripts detected by primer extension

Transcript	Strand	5′ start	Predicted distance	Size	5′ gene		3′ gene	
I001	+	271804	75	75	b0257	+	ykfC	+
I002	−	4231116	179	310	b4024 (‘lysC’)	−	b4025 (‘pgi’)	+
I004	+	3766359	256	n/a	o153 (‘yibG’)	+	yibH	−
I014	−	4374139	196	300	o188 (‘efp’)	+	o155 (‘sugE’)	+

The table lists the transcripts’ 5′ ends; their orientation; the distance between the 5′ ends and the predicted transcripts; the transcripts’ estimated size; and the name and orientation of 5′ and 3′ flanking genes (relative to the + strand). Note that the I004 5′ start point overlaps prediction HB_200 of Carter and colleagues (12), but we did not detect any northern signal that corresponded to this 5′ start (see Figure 1).

or unspecifically processed by ribonucleases. Several known ncRNAs in *E.coli* are specifically processed (32), and our results are similar to previously predicted and verified ncRNAs thought to be specifically processed (9).

It is possible that some of the larger transcripts detected could be processed 5′ or 3′ ends of neighboring mRNAs; e.g. I002 overlaps the 5′ CDS of lysC by 6 nt. The neighboring genes that the other large transcripts can and do overlap with (we did not establish the 5′ ends of I003 and I006, but I014 overlaps 4 nt in the 5′ CDS of efp) are on the opposite strand of the verified transcripts. Thus, it is possible that these transcripts can regulate their neighboring genes through an antisense mechanism.

Because the transcripts we have tested have not previously been detected, these transcripts may be unstable or of low abundance and therefore difficult to detect. Such instability may also explain some of the multiple bands. Another possible explanation could be that the strand-specific probes bind to other transcripts, but a Blast (33) search of the probes against the complete *E.coli* genome did not give any matches with *E*-values below 0.1, except for the intended target sites. Thus, it is unlikely that the multiple bands in the northern blots are caused by the probes hybridizing to other complementary transcripts.

Excluding tRNAs and rRNAs improves specificity

Our initial database of ncRNA genes was slightly biased towards rRNA and tRNA genes. As our main focus was to identify other small RNA genes, we did a separate analysis where we trained classifiers exclusively on the sRNA sequences. In this analysis, we used the query language and methodology from Saetrom (18), i.e. a classifier was the average of 10 GPboost runs instead of a single run as in our previous experiments.

Using this approach, we predicted 135 of 255 sRNA sequence windows, which included sequence windows from all but the micF and sraE genes. In addition, the approach identified 140 potential ncRNAs, 69 of which were novel.

A cross-reference of the potential ncRNAs identified by this method with the list of known genes (see Table 2) showed that it had correctly identified the rygC, SroG and rdlD genes. On the other hand, only Contig_496 of the sequences in Table 3 was correctly identified; two other predictions overlapped Contig_440 and Contig_286, but these were on the opposite strand.

As a comparison, we ran an experiment where we again used the approach of Saetrom (18), but also included the tRNAs and rRNAs. We now identified all the ncRNAs in the training set except spf, sraB, sraD and micF, and predicted 401 potential ncRNAs; 168 of these were novel. Although this approach identified slightly fewer of the sRNA genes in the training set compared to the classifiers that were trained only on the sRNA sequences, it identified all the tRNAs and rRNAs; the sRNA-based classifiers only identified 15 of 22 rRNAs and 21 of 86 tRNAs. Thus, as expected, when the rRNAs and tRNAs are excluded from the training set, the resulting classifiers become more specific. In accordance with this, the classifiers trained on the complete ncRNA set identified four of the known ncRNAs in our set of intergenic sequences (rdlA, rygC, SroG and rdlD), and seven of the nine contigs from Table 3 (Contig_440 and Contig_286 were identified on the wrong strand).

DISCUSSION

We have described a novel method for finding non-coding RNA genes and proved its applicability by analyzing *E.coli* intergenic regions, and testing and experimentally confirming 9 of the top 10 scoring predictions and 3 other predictions with lower score. Several groups have searched for new ncRNAs in *E.coli* (8–14), which have resulted in a list of about ∼1000 non-redundant and untested candidates (7). Our predictions mostly confirm the predictions of the other methods, but we also predict several new ncRNA genes, and, as our experimental verifications show, at least six of these new predictions are genuine ncRNAs: 12 of the 16 tested candidates, including 6 novel predictions, were verified. It would therefore be surprising if none of the other candidates are ncRNAs.

Northern analysis and primer extension showed that our method could not completely identify the true transcript of the verified predictions. That is, the algorithm either only predicted a portion of the transcript or misplaced its start and stop site. There are three main reasons for these errors. First, our data set consisted of 50 nt sequence windows with 25 nt overlap. Consequently, we could only predict the correct start and stop site if these regions aligned with any of the sequence windows in our data set. Here, we would expect that only 1 of 25 start sites would align by chance. Second, our algorithm did not recognize all the sequence windows of the known ncRNAs in the training set. We would therefore be surprised if it correctly predicted the complete sequence of any new transcripts. Third, our algorithm is biased in the sense that it will only detect regions that are similar to regions in the known ncRNAs. Thus, the algorithm would have trouble detecting the novel domains in the new transcripts.

Because of these three shortcomings, we did not expect the algorithm to correctly identify the complete sequence of any new transcripts. Rather, we developed the algorithm as

a complementary tool to the existing ncRNA prediction algorithms, which use other features to predict ncRNAs. As an analogy to standard protein coding gene prediction, our algorithm can be considered a content analyzer (34). To get more reliable predictions of complete ncRNAs, we can for example combine our algorithm with algorithms that look for signals such as transcription initiation and termination (9,13). We are currently looking into this.

When comparing our predictions to those of other methods and to the known ncRNAs included in our set of intergenic sequences (see Table 2), we found that some of our predictions were on the opposite strand. In addition, 47 of our predictions overlapped predictions that our algorithm made on the opposite strand (see Supplementary Material). Thus, it appears that the algorithm has problems identifying the correct strand for some transcripts. These results are, however, related to the above discussion on the algorithm's bias: the algorithm will only detect domains that have a similar sequence to those in the known ncRNAs. An ncRNA's function often lies in its secondary structure, however, and in general, several different sequences can fold into the same secondary structure. In particular, for certain sequences both the original and reverse complementary sequence fold into similar secondary structures. Thus, if the reverse complementary of such sequences more closely resembles the known ncRNAs than does the original sequences, our algorithm will predict the reverse complementary sequence to be an ncRNA domain. This is for instance the case for rdlA in Table 2. Our algorithm incorrectly predicted the reverse complementary sequence of rdlA to be an ncRNA, but the secondary structures of the correct sequence mirrors that of the reverse complementary (data not shown).

A recent study uses the sequence conservation of known ncRNA genes and intergenic regions to estimate the number of sRNAs (ncRNAs other than tRNA and rRNA) in *E.coli* to be between 118 and 260 (15). The authors then argue that because the number of sRNA genes that either have been experimentally verified or predicted by at least two different studies in *E.coli* were 150 (at that time), their estimates may be an upper limit to the number of sRNA genes in *E.coli* (15). Following their logic, our results indicate that the number of sRNA genes in *E.coli* may be closer to their highest estimate than to their lowest. This is because we have significantly extended the list of ncRNAs predicted by more than one method, and because we have shown that our method predicts new ncRNAs that have remained undetected by other methods.

To summarize, we have shown that our approach for ncRNA prediction is both accurate and complementary to existing methods. That is, it identifies genuine ncRNA genes, some of which have not been predicted by any other methods.

SUPPLEMENTARY MATERIAL

Supplementary Material is available at NAR Online.

ACKNOWLEDGEMENTS

We thank K. Lagesen for providing the initial database of *E.coli* ncRNA genes, Y. Esbensen for providing the RNA isolation protocol, and H.E. Krokan and M. Bjørås for valuable comments on the manuscript. The work was supported by the Norwegian Research Council, grants 151899/150, 152020/310 and 152001/150, and the bioinformatics platform at the Norwegian University of Science and Technology, Trondheim, Norway. Funding to pay the Open Access publication charges for this article was provided by the Norwegian Research Council.

Conflict of interest statement. None declared.

REFERENCES

1. Eddy,S.R. (2001) Non-coding RNA genes and the modern RNA world. *Nature Rev. Genet.*, **2**, 919–929.
2. Wassarman,K.M. (2002) Small RNAs in bacteria: diverse regulators of gene expression in response to environmental changes. *Cell*, **109**, 141–144.
3. Storz,G. (2002) An expanding universe of noncoding RNAs. *Science*, **296**, 1260–1263.
4. Cawley,S., Bekiranov,S., Ng,H.H., Kapranov,P., Sekinger,E.A., Kampa,D., Piccolboni,A., Sementchenko,V., Cheng,J., Williams,A.J., Wheeler,R., Wong,B., Drenkow,J., Yamanaka,M., Patel,S., Brubaker,S., Tammana,H., Helt,G., Struhl,K. and Gingeras,T.R. (2004) Unbiased mapping of transcription factor binding sites along human chromosomes 21 and 22 points to widespread regulation of noncoding RNAs. *Cell*, **116**, 499–509.
5. Mattick,J.S. (2004) RNA regulation: a new genetics? *Nature Rev. Genet.*, **5**, 316–323.
6. Wassarman,K.M., Zhang,A. and Storz,G. (1999) Small RNAs in *Escherichia coli*. *Trends Microbiol.*, **7**, 37–45.
7. Hershberg,R., Altuvia,S. and Margalit,H. (2003) A survey of small RNA-encoding genes in *Escherichia coli*. *Nucleic Acids Res.*, **31**, 1813–1820.
8. Vogel,J., Bartels,V., Tang,T.H., Churakov,G., Slagter-Jager,J.G., Huttenhofer,A. and Wagner,E.G.H. (2003) RNomics in *Escherichia coli* detects new sRNA species and indicates parallel transcriptional output in bacteria. *Nucleic Acids Res.*, **31**, 6435–6443.
9. Argaman,L., Hershberg,R., Vogel,J., Bejerano,G., Wagner,E.G.H., Margalit,H. and Altuvia,S. (2001) Novel small RNA-encoding genes in the intergenic regions of *Escherichia coli*. *Curr. Biol.*, **11**, 941–950.
10. Wassarman,K.M., Repoila,F., Rosenow,C., Storz,G. and Gottesman,S. (2001) Identification of novel small RNAs using comparative genomics and microarrays. *Genes Dev.*, **15**, 1637–1651.
11. Rivas,E., Klein,R.J., Jones,T.A. and Eddy,S.R. (2001) Computational identification of noncoding RNAs in *E. coli* by comparative genomics. *Curr. Biol.*, **11**, 1369–1373.
12. Carter,R.J., Dubchak,I. and Holbrook,S.R. (2001) A computational approach to identify genes for functional RNAs in genomic sequences. *Nucleic Acids Res.*, **29**, 3928–3938.
13. Chen,S., Lesnik,E.A., Hall,T.A., Sampath,R., Griffey,R.H., Ecker,D.J. and Blyn,L.B. (2002) A bioinformatics based approach to discover small RNA genes in the *Escherichia coli* genome. *Biosystems*, **65**, 157–177.
14. Tjaden,B., Saxena,R.M., Stolyar,S., Haynor,D.R., Kolker,E. and Rosenow,C. (2002) Transcriptome analysis of *Escherichia coli* using high-density oligonucleotide probe arrays. *Nucleic Acids Res.*, **30**, 3732–3738.
15. Zhang,Y., Zhang,Z., Ling,L., Shi,B. and Chen,R. (2004) Conservation analysis of small RNA genes in *Escherichia coli*. *Bioinformatics*, **20**, 599–603.
16. Blattner,F.R., Plunkett,G.,III, Bloch,C.A., Perna,N.T., Burland,V., Riley,M., Collado-Vides,J., Glasner,J.D., Rode,C.K., Mayhew,G.F., Gregor,J., Davis,N.W., Kirkpatrick,H.A., Goeden,M.A., Rose,D.J., Mau,B. and Shao,Y.S. (1997) The complete genome sequence of *Escherichia coli* K-12. *Science*, **277**, 1453–1474.
17. Guigueno,A., Dassa,J., Belin,P. and Boquet,P.L. (2001) Oversynthesis of a new *Escherichia coli* small RNA suppresses export toxicity of DsbA'-PhoA unfoldable periplasmic proteins. *J. Bacteriol.*, **183**, 1147–1158.
18. Sætrom,P. (2004) Predicting the efficacy of short oligonucleotides in antisense and RNAi experiments with boosted genetic programming. *Bioinformatics*, **20**, 3055–3063.
19. Sætrom,P. and Snøve,Jr,O. (2004) A comparison of siRNA efficacy predictors. *Biochem. Biophys. Res. Commun.*, **321**, 247–253.
20. Koza,J.R. (1992) *Genetic Programming: On the Programming of Computers by Natural Selection*. MIT Press, Cambridge, MA.

21. Meir,R. and Rätsch,G. (2003) An introduction to boosting and leveraging. In Mendelson,S. and Smola,A. (eds), *Advanced Lectures on Machine Learning*. Springer-Verlag, Vol. 2600, pp. 118–183.
22. Rätsch,G., Onoda,T. and Müller,K.-R. (2001) Soft margins for AdaBoost. *Mach. Learn.*, **42**, 287–320.
23. Halaas,A., Svingen,B., Nedland,M., Sætrom,P., Snøve,Jr,O. and Birkeland,O.R. (2004) A recursive MISD architecture for pattern matching. *IEEE Trans. VLSI Syst.*, **12**, 727–734.
24. Baldi,P., Brunak,S., Chauvin,Y., Andersen,C.A. and Nielsen,H. (2000) Assessing the accuracy of prediction algorithms for classification: an overview. *Bioinformatics*, **16**, 412–424.
25. Sambrook,J., Fritsch,E.F. and Maniatis,T. (1989) *Molecular Cloning: A laboratory Manual*. 2nd edn. Cold Spring Harbor Laboratory Press, Cold Spring Harbor, NY.
26. Stone,M. (1974) Cross-validatory choice and assessment of statistical predictions. *J. R. Stat. Soc. [Ser. B] (Methodological)*, **36**, 111–147.
27. Kohavi,R. (1995) A study of cross-validation and bootstrap for accuracy estimation and model selection. In *Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence*. Morgan Kaufmann, San Mateo, CA, 1137–1143.
28. Glasner,J.D., Liss,P., Plunkett,G.,III, Darling,A., Prasad,T., Rusch,M., Byrnes,A., Gilson,M., Biehl,B., Blattner,F.R. and Perna,N.T. (2003) ASAP, a systematic annotation package for community analysis of genomes. *Nucleic Acids Res.*, **31**, 147–151.
29. Kawano,M., Reynolds,A.A., Miranda-Rios,J. and Storz,G. (2005) Detection of 5′- and 3′-UTR-derived small RNAs and *cis*-encoded antisense RNAs in *Escherichia coli*. *Nucleic Acids Res.*, **33**, 1040–1050.
30. Kawano,M., Oshima,T., Kasai,H. and Mori,H. (2002) Molecular characterization of long direct repeat (LDR) sequences expressing a stable mRNA encoding for a 35-amino-acid cell-killing peptide and a *cis*-encoded small antisense RNA in *Escherichia coli*. *Mol. Microbiol.*, **45**, 333.
31. Pedersen,K. and Gerdes,K. (1999) Multiple *hok* genes on the chromosome of *Escherichia coli*. *Mol. Microbiol.*, **32**, 1090–1102.
32. Li,Z., Pandit,S. and Deutscher,M.P. (1998) 3′ Exoribonucleolytic trimming is a common feature of the maturation of small, stable RNAs in *Escherichia coli*. *Proc. Natl Acad. Sci. USA.*, **95**, 2856–2861.
33. Altschul,S.F., Gish,W., Miller,W., Myers,E.W. and Lipman,D.J. (1990) Basic local alignment search tool. *J. Mol. Biol.*, **215**, 403–410.
34. Mathé,C., Sagot,M.-F., Schiex,T. and Rouzé,P. (2002) Current methods of gene prediction, their strengths and weaknesses. *Nucleic Acids Res.*, **30**, 4103–4117.

Paper V

Many commonly used siRNAs risk off-target activity

Available online at www.sciencedirect.com
SCIENCE DIRECT®
Biochemical and Biophysical Research Communications 319 (2004) 256–263
BBRC
www.elsevier.com/locate/ybbrc

Many commonly used siRNAs risk off-target activity☆

Ola Snøve Jr.[a] and Torgeir Holen[b,*]

[a] Interagon AS, Medisinsk-teknisk senter, NO–7489, Trondheim, Norway
[b] Center for Molecular Biology and Neuroscience (CMBN), University of Oslo, P.b. 1105 Blindern, 0317 Oslo, Norway

Received 31 March 2004
Available online 14 May 2004

Abstract

Using small interfering RNA (siRNA) to induce sequence specific gene silencing is fast becoming a standard tool in functional genomics. As siRNAs in some cases tolerate mismatches with the mRNA target, knockdown of genes other than the intended target could make results difficult to interpret. In an investigation of 359 published siRNA sequences, we have found that about 75% of them have a risk of eliciting non-specific effects. A possible cause for this is the popular BLAST search engine, which is inappropriate for such short oligos as siRNAs. Furthermore, we used new special purpose hardware to do a transcriptome-wide screening of all possible siRNAs, and show that many unique siRNAs exist per target even if several mismatches are allowed. Hence, we argue that the risk of off-target effects is unnecessary and should be avoided in future siRNA design.

Keywords: siRNA; MicroRNA; RNA interference; Specificity

RNA interference (RNAi) is an ancient immune system on the genomic level that has been demonstrated to protect against viruses and transposons in lower animals and plants [1,2]. The active agents of RNAi are short RNAs with sequence complementarity to the target RNA. The potency and therapeutic potential of RNAi have been demonstrated by knockdown of mRNA in mammalian cells [3,4] and inhibition of disease-causing viruses like HIV [5].

Off-target activity by a small RNA can principally arise from two mechanisms: depletion on the mRNA level or translational suppression at the protein level. RNAi is generally believed to be exquisitely specific [6]. However, several groups have now observed that siRNAs can tolerate one mismatch to the mRNA target and at the same time retain good silencing capacity [7–14]. In some cases, siRNAs can tolerate several mismatches [7,11,12,15], or even tolerate mismatches while acting as a single-stranded antisense siRNA [16]. Furthermore, some domains of the siRNAs tolerate more of the mismatches than others [12,17]. A recent study also demonstrated tolerance for G:U wobble pairing between the RNA oligo and the target RNA [15]. While some microarray studies found a high specificity of siRNA effects [18,19], two other studies found large non-specific effects [20,21]. Large studies on siRNA mismatch tolerance have not yet been performed.

Another possible mechanism for off-target activity arises from the fact that the physical structure of siRNAs, ~21 nucleotide (nt) RNA oligomers, appears to be identical to the related class of microRNAs [22]. MicroRNAs are short endogenously transcribed RNAs that yield mRNA translation inhibition rather than mRNA degradation. MicroRNAs seem to have mismatches between RNA oligo and RNA target inherent in their structure. The rules regulating the functional structure of microRNA are not yet well known, but are under investigation [14,23,24].

Together the mechanisms of siRNA mismatch tolerance and microRNA translation inhibition create a risk of off-target activity when ~21 nt RNAs are introduced into human cells. We wanted to evaluate the risk of off-target activity in commonly used siRNAs in silico, and investigate the potential for finding oligomers without

☆ Supplementary data associated with this article can be found, in the online version, at doi:10.1016/j.bbrc.2004.04.175.

* Corresponding author.
E-mail address: torgeir.holen@basalmed.uio.no (T. Holen).

0006-291X/$ - see front matter © 2004 Elsevier Inc. All rights reserved.
doi:10.1016/j.bbrc.2004.04.175

an inherent off-target risk in the set of all possible siRNAs in the human transcriptome.

Materials and methods

The collection of siRNAs was built by selection from publications in the most prestigious journals, and from the websites of commercial companies Qiagen (www.qiagen.com) and Ambion (www.ambion.com) with the rationale that the possible impact of off-target effects would be increased for siRNAs in common use. No complete collection of published siRNAs was attempted, and the reported mismatch incidence is thus only strictly valid for this subset of all published siRNAs.

A special purpose processor was used—the Pattern Matching Chip (Interagon; Trondheim, Norway; www.interagon.com)—to search for mRNAs with sequence similarity to published siRNAs. Moreover, we extracted 58,151,368 oligonucleotides of length 21 from the known cDNA in Ensembl's 17.33.1 release and screened these for mismatch similarity with the rest of the database. The Pattern Matching Chip's architecture is massively parallel and ideal for high throughput screenings such as this [25]. We consider only the number of mismatches in ungapped alignments between the siRNA probe and the mRNA target, but the Pattern Matching Chip's functionality is not by any means limited to this (information on applicability is available upon request to the authors). The total throughput of the accelerated workstation was equivalent to 512 GB/s per 21mer with unlimited mismatch sensitivity. The performance for pattern-matching purposes is thus several orders of magnitudes higher than what is generally achievable with known regular expression algorithms on ordinary processors, and enabled a screen of all 21mers against the transcriptome in just above 10 h—a task that has not been undertaken before because it would require orders of magnitude longer computing time.

Results

BLAST [26] is frequently used to determine if an siRNA is target specific. It is important to notice that BLAST is a search heuristic that sacrifices some sensitivity to gain speed, and that different search parameters may yield very different results. For example, the word size—often denoted w—is important because a region that does not contain at least w successively matching characters will be missed by BLAST [27]. The loss of sensitivity is negligible for most applications, but short query searching in general and siRNA screening in particular, require careful attention: a fraction of all potentially relevant alignments may be missed by BLAST depending on the length of the query and the positions of the mismatches.

Table 1 shows the fraction of alignments that will remain undetected by BLAST given different word sizes and number of mismatches for 19mer and 21mer queries. It has been suggested that the 3′ overhang nucleotides of the siRNA duplexes do not contribute to sequence specificity [3], and this would make 19mer targets viable. If the word size is seven—as is the recommended word size when searching for short, nearly exact matches with NCBI's BLAST—about 6% of all possible alignments with three mismatches between

Table 1
Fraction of short sequences that remains undetected by BLAST given word size, w, mismatch threshold, and oligonucleotide size

	1 mismatch		2 mismatches		3 mismatches		4 mismatches		5 mismatches		6 mismatches	
	19-mer	21-mer	19-mer	21-mer	19-mer	21-mer	19-mer	21-mer	19-mer	21-mer	19-mer	21-mer
$w = 6$	0.000	0.000	0.000	0.000	0.036	0.008	0.168	0.070	0.363	0.207	0.563	0.387
$w = 7$	0.000	0.000	0.018	0.000	0.154	0.063	0.374	0.222	0.586	0.425	0.762	0.615
$w = 8$	0.000	0.000	0.088	0.029	0.325	0.185	0.574	0.411	0.762	0.621	0.881	0.779
$w = 9$	0.000	0.000	0.211	0.100	0.505	0.343	0.729	0.586	0.870	0.766	0.946	0.881
$w = 10$	0.053	0.000	0.368	0.214	0.653	0.504	0.837	0.724	0.935	0.864	0.978	0.940
$w = 11$	0.158	0.048	0.509	0.357	0.769	0.639	0.910	0.825	0.971	0.926	0.993	0.972

The statistics are calculated assuming a random distribution of mismatches between 21mers in the transcriptome.

Table 2
Twenty published siRNAs with risk of off-target activity

siRNA primary target (Acc#/RefSeq)	siRNA name	siRNA off-target hit (Acc#/RefSeq) ⟨mismatches|wobbles⟩	Source article of siRNA
Lamin A/C (70 kDa lamin) (NM_170707)	Lamin A/C	NDUFS4 (NM_002495) <8|19> 5'-AACUGGAGUUCCAGAAGAGCA-3' 3'-UUGACCUGAAGGUCUUCUUGU-5'	Elbashir et al. [3]
Lamin B2 (NM_032737)	Lamin B2	CLSTN2 (NM_022131) <11, 21|> 5'-AAGAGGAGGAAGAAGCCGAGG-3' 3'-UUCUCCUCCUCCUUCGGCUCA-5' Plus four other double mismatch hits, and 63 triple mismatch hits (Supplemental Table S3).	Harborth et al. [29]
Caspase-1 (NM_033292)	Caspase-1	COP (NM_052889) <|> 5'-AAGGGGCACAGGCAUGCCAAA-3' 3'-UUCCCCGUGUCCGUACGGUUU-5'	Lassus et al. [37]
Caspase-8 (NM_001228)	Caspase-8	TNRC9 (U80736) <14,16|> 5'-AAUCACAGACUUUUGCCAAAG-3' 3'-UUAGUGUCUGAAACCUGUUUC-5'	Chun et al. [30]
Serine/threonine E-protein kinase kist (NM_175866)	hKIS	SARS (NM_006513) <4,10|> 5'-AAGGAGUUCAUGCCGCCAGGA-3' 3'-UUCGUCAAGAACGGCGGUCCU-5'	Boehm et al. [31]
β-Arrestin 2 (NM_004313)	ARRB2	PABPC4 (NM_003819) <3,15|> 5'-AAUGACCGCAAAGUAUUUGUG-3' 3'-UUCCUGGCGUUUCACAAACAC-5' Plus O43713 (O43713) <3,15|>	Ahn et al. [38]
Suppressor of cytokine signaling 3 (NM_003955)	SOCS-3	ZNF323 (NM_030899)<8,9|> 5'-AAGACCCUAUCUGGGACCAAGAA-3' 3'-UUCUGGGUCAGACCCUGGUUCUU-5'	Leung et al. [32]
E1A-associated protein P300 (NM_001429)	P300	SLC1A2 (NM_004171) <4|1> 5'-GAUAGUGCUGAAGAGGAGGGG-3' 3'-UUACCACGACUUCUCCUCCCC-5'	Debes et al. [39]
HIF-prolyl hydroxylase 1 (NM_053046)	PHD1-II	SLC4A2 (NM_003040) <2|16> 5'-CCCCACCAUGGCCUUUGUGCG-3' 3'-GCGGUGGUACCGGAACCACGC-5'	Berra et al. [40]
Serine Protease HTRA2 (NM_013247)	HtrA2-2	CRSP2 (NM_004229) <5,8|> 5'-AAUGCCAUCAACAAACUCCCC-3' 3'-UUACCGUUGUUGUUUGAGGGG-5'	Martins et al. [48]
Tumor protein p53, TP53 (NM_000546)	p53	TP73 (NM_005427) <14|1,2> 5'-GGGCAUGAACCGGCGGCCCAU-3' 3'-UUCGUACUUGGCCUCCGGGUA-5'	Martinez et al. [41]

Table 2 (*continued*)

siRNA primary target (Acc#/RefSeq)	siRNA name	siRNA off-target hit (Acc#/RefSeq) ⟨mismatches|wobbles⟩	Source article of siRNA
Aminopeptidase PILS (NM_016442)	ERAAP	NM_152418 (NM_152418) <21,11|6> 5'-AACGUGGUGACGGGACACCAG-3' |||||:|||| ||||||||| 3'-UUGCAUCACUACCCUGUGGUA-5'	Serwold et al. [42]
Serine/threonine protein kinase PLK (NM_005030)	Plk1	Q8N8U7 (Q8N8U7) <1|6,9> 5'-CAGGGUGGUUUUGCCAAGUGC-3' ||||:||:||||||||||| 3'-UUCCCGCCGAAACGGUUCACG-5'	Liu and Erikson [43]
RAS-related protein RAL-A (NM_005402)	RalA-II	CNTN4 (NM_175607) <5,21|2> 5'-AGAUACCACUGCUCAGCUCUC-3' |:|| |||||||||||||| 3'-UUUAAGGUGACGAGUCGAGAC-5'	Moskalenko et al. [44]
C–C chemokine receptor type (NM_000579)	CCR5-1	CCR2 (NM_000647) <4,7|18> 5'-AAGUGCUUGACUGACAUUUAC-3' ||| || ||||||||||:||| 3'-UUCUCGUACUGACUGUAGAUG-5'	Qin et al. [45]
Likely ortholog of *Caenorhabditis elegans* anterior pharynx defective 1A (APH-1A) (NM_016022)	APH-1a-2	IRS1 (NM_005544)<4,7|21> 5'-AACAGAAGGAGAUGGGUGAUU-3' ||| || |||||||||||: 3'-UUGCCUACCUCUACCCACUAG-5'	Lee et al. [28]
ATR interacting protein (NM_032166)	ATRIP	ARHGEF6 (NM_004840) <9|7,12> 5'-AAGAAGGGACCUAGAAAAGCU-3' ||||||:| ||:||||||||| 3'-UUCUUCUCCGGGUCUUUUCGA-5' Plus 4 other triple mismatch hits (Supplemental Table S3).	Cortez et al. [46]
Protein kinase C-delta (NM_006254)	PKC-delta	Q96I56 (Q96I56) <1,20|2> 5'-UGCUGAGCGCCUCCUUCAGC-3' :||||||||||||||| | 3'-UUGACUCGCGGAGGAAGUAG-5' Plus 15 other triple mismatch hits (Supplemental Table S3).	Yoshida et al. [47]
Numa1 (NM_006185)	NuMa	FBXO21 (NM_015002) <10, 16|5> 5'-AAGGUGUGGAAGGAGCAGUUC-3' ||||:|||| ||||| ||||| 3'-UUCCGCACCGUCCUCUUCAAG-5' Plus 6 other triple mismatch hits (Supplemental Table S3).	Elbashir et al. [3]
Cylindromatosis (turban tumor syndrome) (NM_015247)	CYLD (19mer, from shRNA)	NM_021638 (NM_021638) <|13,14> 5'-CAAAGAGAACUGUGUGAGG-3' ||||||||||||::||||| 3'-GUUUCUCUUGACGUACUCC-5'	Kovalenko et al. [35]

The shown off-target hits, 9 with double-mismatches, 10 with triple mismatches, and 1 double-wobble mismatch from an shRNA, are excerpt of the full list of off-target hits available as Supplementary Table S3. Alignments of target areas of the most significant off-target hit are shown, with mRNA presented as the upper strand and the complementary siRNA strand being the lower strand. Unique accession numbers and Ensembl RefSeq numbers for the primary target mRNA and the possible off-targeted mRNA are given when available. In some cases dTT 3′ ends are presented as UU. A compressed presentation form is also utilized in the form $\langle X1, X2, \ldots, Xn | Y1, Y2, \ldots, Yn \rangle$, where *Xn* and *Yn* stand for mismatch positions and G:U wobble positions, respectively, relative to the 5′ end of the mRNA:siRNA alignment.

21mers will be missed. Moreover, the fraction of alignments that are missed increases to 15% if 19mers are used instead of 21mers. To summarize, increasing the word size or allowing more mismatches both contribute towards a higher rate of missed hits. Three or more mismatches may be biologically relevant since G:U

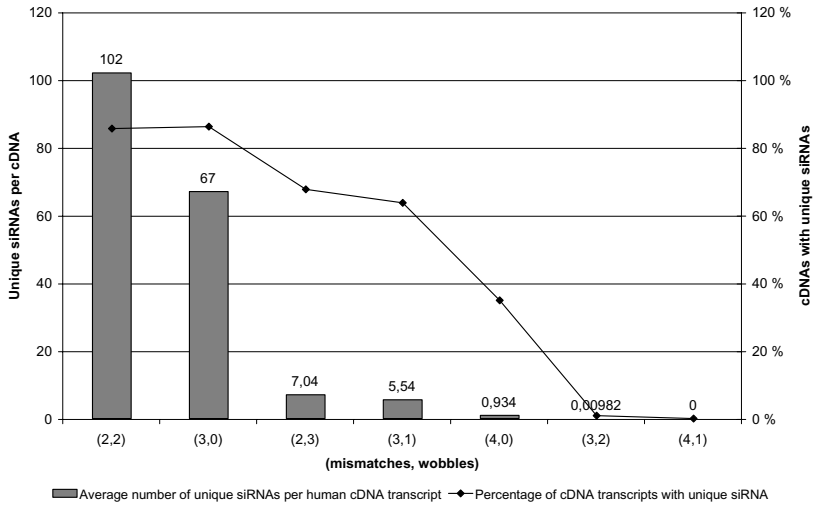

Fig. 1. siRNA uniqueness statistics for various dissimilarity thresholds in human cDNA. The bars relate to the left vertical axis and show how many unique siRNAs that exist given various mismatch and G:U wobble thresholds (G:U wobbles are not counted as regular mismatches). The line relates to the right vertical axis and shows the percentage of transcripts that contains at least one unique siRNA given the same threshold parameters.

mismatches seem to be more tolerable in the RNAi pathway than are regular mismatches [15].

We collected and screened siRNAs from public sources to test the practical consequences of missed hits from BLAST. PubMed lists *p.t.* 467 articles with keyword *siRNA*, up from 180 in 2002 and 25 in 2001, illustrating the rapid and widespread application of siRNAs. A complete collection of published siRNAs was not possible, because the exact sequences of oligonucleotides were not always stated in the papers. Altogether, we screened 359 human siRNAs against all known cDNAs from Ensembl's 17.33.1 release.

Table 2 is an excerpt from the results (full listing in Supplementary Table S3) and contains the siRNAs that were perceived as most interesting based on the number of mismatches, the relative mismatch positions, and the prestige of the journals in which the siRNAs were published. We were surprised to see that many siRNAs were identical to other sequences in all but zero, one or two positions. About 20% of the collected siRNAs had two or fewer mismatches in off-target alignments (Supplementary Table S3). We will briefly discuss some illustrative hits in the Discussion.

Silencing activity with three mismatches between siRNA and mRNA was demonstrated only recently, where three G:U wobble positions were found to be tolerated [15]. If this phenomenon is widespread, then siRNA off-target activity might affect the results of 10 studies listed in Table 2 [28]. Furthermore, approximately 75% of the 359 siRNAs collected had off-target alignments with three or fewer mismatches (Supplementary Table S3). Hence, if triple mismatch tolerance is confirmed, the risk of off-target activity seems high with publicly available siRNAs.

To test our predictions of siRNA off-target activity, we re-analyzed the data of two microarray studies performed by Chi et al. and Jackson et al. [19,20]. These papers concluded differently on the specificity of siRNA. While Chi et al. found their siRNAs to elicit no secondary responses, Jackson et al. [20] found that their siRNAs were unspecific, despite having designed the siRNAs, using BLAST, to display fewer than 18 nucleotides of identity to known genes other than the targeted gene. However, we found 54 instances of 18 nucleotides of identity with other genes in the transcriptome (Supplementary Table S4). Furthermore, we also found that two of these off-target predictions resulted in very significant downregulation of the off-target genes, while several others had less significant downregulation (Supplementary Table S6). We note that Chi et al.'s siRNAs had only five instances of off-target hits with less than four mismatches (Supplementary Table S5). Thus, it seems that Chi et al.'s siRNAs were by design more unique than the siRNAs of Jackson et al. though both studies had siRNAs that were generally more unique than many of the other siRNAs in the literature (Supplementary Table S3). Taken together, these results indicate a partial explanation of the differing siRNA specificity results of Chi et al. and Jackson et al.

The probability of finding unique 21mers decreases as more mismatches are allowed. The possibility of screening every possible oligonucleotide is usually dismissed as impractical since it would take months of computing time to use BLAST to align the approximately 60 million 21mers that can be extracted from known cDNA. We completed the task in approximately 10 h with newly developed pattern matching hardware. Since genomic records are updated regularly, the screen must be repeated to account for differences between the versions; thus, speed is important from this perspective as well.

We found that most transcripts contain 21mers that are unique even if three mismatches are allowed. Fig. 1

shows the average number of unique 21mers that can be found per transcript (bars) and the fraction of transcripts that contain unique 21mers (line) at various levels of dissimilarity. A 21mer is considered unique if it matches the target sequence only, and if its reverse complement does not match any region in the full cDNA database. Note that many genes produce alternative transcripts that have several exons in common, which means that they are less likely to contain entirely unique oligonucleotides. If the distinction between transcripts and genes were not important, a higher fraction of genes would contain unique siRNAs at the different levels of dissimilarity. Nevertheless, about 90% of all transcripts contains at least one unique siRNA even if two mismatches and two G:U wobbles are allowed. Moreover, almost 40% of all transcripts have target specific oligos even if siRNAs were to have silencing activity with as many as four mismatches to their targets. Hence, very specific siRNAs exist for most targets in the human transcriptome.

Discussion

In some cases, short interfering RNAs (siRNAs) can tolerate mismatches [7–16], while microRNAs have been reported to be able to act as siRNA [14,24] and vice versa [23]. The phenomena of mismatch tolerance and microRNA silencing thus creates a risk of knockdown of other genes besides the intended target. A limited number of illustrative cases from influential publications will serve as examples (Table 2). The first two examples of siRNAs are chosen from two of the founder articles of the siRNA field. The Lamin A/C and the NDUFS4 mRNA are identical in 19 of 21 positions (Table 2): one is a G:U wobble near the 3′ end of the alignment, and the other is a mismatch in the less sensitive 5′ end of the siRNA [12,17]. This kind of off-target hit could theoretically carry a high risk of biological off-target activity.

More serious off-target hits are generated by the siRNA against Lamin B2, a gene that induced cell into apoptosis when it was downregulated. The gene was therefore concluded to be essential [29]. If there is significant mRNA depletion from only one of the four other genes that the Lamin B2 siRNA have only two mismatches with, or from the 63 other genes that this siRNA have three mismatches with (Table 2 and Supplementary Table S3), then the conclusions of these particular siRNA-experiments seem weaker.

Other significant examples from Table 2 should also be briefly mentioned. Apoptosis is a complex cascade of events where an initial difference in parameters might give a completely different outcome. Two siRNAs used in apoptosis-studies, Caspase-1 and Caspase-8, have significant similarity with other genes (Table 2), and might thus theoretically affect some of the conclusions of these works [30]. Cell signaling is another field with complex chains of effects where off-target activity by siRNA might influence the results. Illustrative examples include siRNAs against β-arrestin, KIST, and Suppressor of Cytokine Signaling 3 (Table 2) [31,32]. Furthermore, cancer studies, as in the case of the siRNA against p300 (Table 2), are yet another field where mismatch effects might cause confusion, both in laboratory experiments and later in clinical studies. The double mismatches that have been discussed so far would have been detected by BLAST (Table 1). Nevertheless, about 20% of the 359 siRNAs that was screened are not specific if two mismatches are allowed.

Tolerance of three mismatches between siRNA and mRNA was reported recently [15]. If triple mismatch tolerance is confirmed in other studies, the risk of off-target activity is rather common, with approximately 75% of the 359 siRNAs collected having off-target alignments with three or fewer mismatches (Supplementary Table S3). As many scientists seem to prefer to use established siRNAs from existing publications rather than designing siRNAs de novo, the risk of off-target activity is propagated to many new studies.

The use of siRNA from hairpin siRNA constructs provides additional complexity and an additional risk factor. Although both Brummelkamp et al. [33] and Paddison et al. [34] have demonstrated production of both sense and antisense strands from hairpin siRNA transcripts, the exact position of double-stranded hairpin cleavage has not been studied to our knowledge. Furthermore, the exact hairpin RNA transcript has only been predicted [33]. Possibly, several siRNAs differing only slightly in sequence are produced, thus increasing the uncertainty of specific targeting when using hairpin siRNA as compared with synthetic siRNA. The siRNA against CYLD [35] has a double-wobble mismatch in the central 19-mer (Table 2), which might cause off-target activity depending on the exact transcript, hairpin RNA cleavage, and the role of the siRNA overhangs in target recognition [3].

Thus, in conclusion, siRNA design should take into account that:

(1) BLAST may miss important alignments for such short oligos as siRNAs;
(2) many commonly used siRNAs that have been published are (therefore) not sufficiently unique to avoid risk for off-target activity; but
(3) oligos that are more unique do exist, and can be found using algorithms with higher sensitivity such as Smith and Waterman [36].

Finally, we stress that this in silico study merely points to possible targets—none of the siRNAs mentioned in this study have yet directly been tested experimentally. When computing the distribution of unique siRNAs in the transcriptome, we find that it is possible to avoid siRNA candidates with three or fewer

mismatches. Thus, we would argue that when designing experiments taking months of effort and high costs, not least of them the cost of the siRNAs themselves, and especially in the likely event of siRNAs going onward to animal and clinical trials, the risk of off-target activity is unnecessary and should be avoided.

References

[1] P.D. Zamore, Ancient pathways programmed by small RNAs, Science 296 (2002) 1265–1269.

[2] R.H. Plasterk, RNA silencing: the genome's immune system, Science 296 (2002) 1263–1265.

[3] S.M. Elbashir, J. Harborth, W. Lendeckel, A. Yalcin, K. Weber, T. Tuschl, Duplexes of 21-nucleotide RNAs mediate RNA interference in cultured mammalian cells, Nature 411 (2001) 494–498.

[4] N.J. Caplen, S. Parrish, F. Imani, A. Fire, R.A. Morgan, Specific inhibition of gene expression by small double-stranded RNAs in invertebrate and vertebrate systems, Proc. Natl. Acad. Sci. USA 98 (2001) 9742–9747.

[5] N.S. Lee, T. Dohjima, G. Bauer, H. Li, M.J. Li, A. Ehsani, P. Salvaterra, J. Rossi, Expression of small interfering RNAs targeted against HIV-1 rev transcripts in human cells, Nat. Biotechnol. 20 (2002) 500–505.

[6] Editorial, Whither RNAi?, Nat. Cell Biol. 5 (2003) 489–490.

[7] A. Boutla, C. Delidakis, I. Livadaras, M. Tsagris, M. Tabler, Short 5′-phosphorylated double-stranded RNAs induce RNA interference in Drosophila, Curr. Biol. 11 (2001) 1776–1780.

[8] T.A. Vickers, S. Koo, C.F. Bennett, S.T. Crooke, N.M. Dean, B.F. Baker, Efficient reduction of target RNAs by small interfering RNA and RNase H-dependent antisense agents. A comparative analysis, J. Biol. Chem. 278 (2003) 7108–7118.

[9] J.Y. Yu, S.L. DeRuiter, D.L. Turner, RNA interference by expression of short-interfering RNAs and hairpin RNAs in mammalian cells, Proc. Natl. Acad. Sci. USA 99 (2002) 6047–6052.

[10] O. Pusch, D. Boden, R. Silbermann, F. Lee, L. Tucker, B. Ramratnam, Nucleotide sequence homology requirements of HIV-1-specific short hairpin RNA, Nucleic Acids Res. 31 (2003) 6444–6449.

[11] T. Holen, M. Amarzguioui, M.T. Wiiger, E. Babaie, H. Prydz, Positional effects of short interfering RNAs targeting the human coagulation trigger tissue factor, Nucleic Acids Res. 30 (2002) 1757–1766.

[12] M. Amarzguioui, T. Holen, E. Babaie, H. Prydz, Tolerance for mutations and chemical modifications in a siRNA, Nucleic Acids Res. 31 (2003) 589–595.

[13] J.M. Jacque, K. Triques, M. Stevenson, Modulation of HIV-1 replication by RNA interference, Nature 418 (2002) 435–438.

[14] Y. Zeng, B.R. Cullen, Sequence requirements for micro RNA processing and function in human cells, RNA 9 (2003) 112–123.

[15] S. Saxena, Z.O. Jonsson, A. Dutta, Small RNAs with imperfect match to endogenous mRNA repress translation: implications for off-target activity of siRNA in mammalian cells, J. Biol. Chem. (2003).

[16] T. Holen, M. Amarzguioui, E. Babaie, H. Prydz, Similar behaviour of single-strand and double-strand siRNAs suggests they act through a common RNAi pathway, Nucleic Acids Res. 31 (2003) 2401–2407.

[17] D.S. Schwarz, G. Hutvagner, T. Du, Z. Xu, N. Aronin, P.D. Zamore, Asymmetry in the assembly of the RNAi enzyme complex, Cell 115 (2003) 199–208.

[18] D. Semizarov, L. Frost, A. Sarthy, P. Kroeger, D.N. Halbert, S.W. Fesik, Specificity of short interfering RNA determined through gene expression signatures, Proc. Natl. Acad. Sci. USA 100 (2003) 6347–6352.

[19] J.T. Chi, H.Y. Chang, N.N. Wang, D.S. Chang, N. Dunphy, P.O. Brown, Genomewide view of gene silencing by small interfering RNAs, Proc. Natl. Acad. Sci. USA 100 (2003) 6343–6346.

[20] A.L. Jackson, S.R. Bartz, J. Schelter, S.V. Kobayashi, J. Burchard, M. Mao, B. Li, G. Cavet, P.S. Linsley, Expression profiling reveals off-target gene regulation by RNAi, Nat. Biotechnol. 21 (2003) 635–637.

[21] S.P. Persengiev, X. Zhu, M.R. Green, Nonspecific, concentration-dependent stimulation and repression of mammalian gene expression by small interfering RNAs (siRNAs), RNA 10 (2004) 12–18.

[22] J.C. Carrington, V. Ambros, Role of microRNAs in plant and animal development, Science 301 (2003) 336–338.

[23] J.G. Doench, C.P. Petersen, P.A. Sharp, siRNAs can function as miRNAs, Genes Dev. 17 (2003) 438–442.

[24] G. Hutvagner, P.D. Zamore, A microRNA in a multiple-turnover RNAi enzyme complex, Science 297 (2002) 2056–2060.

[25] A. Halaas, B. Svingen, M Nedland, P. Sætrom, O. Snøve, O.R. Birkeland, A recursive MISD architecture for pattern matching, IEEE Transactions on VLSI Systems (in press).

[26] S.F. Altschul, W. Gish, W. Miller, E.W. Myers, D.J. Lipman, Basic local alignment search tool, J. Mol. Biol. 215 (1990) 403–410.

[27] I. Korf, M. Yandell, J. Bedell, An essential guide to the basic local alignment tool. (2003), O'Reilly (Beijing).

[28] S.F. Lee, S. Shah, H. Li, C. Yu, W. Han, G. Yu, Mammalian APH-1 interacts with presenilin and nicastrin and is required for intramembrane proteolysis of amyloid-beta precursor protein and Notch, J. Biol. Chem. 277 (2002) 45013–45019.

[29] J. Harborth, S.M. Elbashir, K. Bechert, T. Tuschl, K. Weber, Identification of essential genes in cultured mammalian cells using small interfering RNAs, J. Cell Sci. 114 (2001) 4557–4565.

[30] H.J. Chun, L. Zheng, M. Ahmad, J. Wang, C.K. Speirs, R.M. Siegel, J.K. Dale, J. Puck, J. Davis, C.G. Hall, S. Skoda-Smith, T.P. Atkinson, S.E. Straus, M.J. Lenardo, Pleiotropic defects in lymphocyte activation caused by caspase-8 mutations lead to human immunodeficiency, Nature 419 (2002) 395–399.

[31] M. Boehm, T. Yoshimoto, M.F. Crook, S. Nallamshetty, A. True, G.J. Nabel, E.G. Nabel, A growth factor-dependent nuclear kinase phosphorylates p27(Kip1) and regulates cell cycle progression, EMBO J. 21 (2002) 3390–3401.

[32] K.C. Leung, N. Doyle, M. Ballesteros, K. Sjogren, C.K. Watts, T.H. Low, G.M. Leong, R.J. Ross, K.K. Ho, Estrogen inhibits GH signaling by suppressing GH-induced JAK2 phosphorylation, an effect mediated by SOCS-2, Proc. Natl. Acad. Sci. USA 100 (2003) 1016–1021.

[33] T.R. Brummelkamp, R. Bernards, R. Agami, A system for stable expression of short interfering RNAs in mammalian cells, Science 296 (2002) 550–553.

[34] P.J. Paddison, A.A. Caudy, E. Bernstein, G.J. Hannon, D.S. Conklin, Short hairpin RNAs (shRNAs) induce sequence-specific silencing in mammalian cells, Genes Dev. 16 (2002) 948–958.

[35] A. Kovalenko, C. Chable-Bessia, G. Cantarella, A. Israel, D. Wallach, G. Courtois, The tumour suppressor CYLD negatively regulates NF-kappaB signalling by deubiquitination, Nature 424 (2003) 801–805.

[36] T.F. Smith, M.S. Waterman, Identification of common molecular subsequences, J. Mol. Biol. 147 (1981) 195–197.

[37] P. Lassus, X. Opitz-Araya, Y. Lazebnik, Requirement for caspase-2 in stress-induced apoptosis before mitochondrial permeabilization, Science 297 (2002) 1352–1354.

[38] S. Ahn, C.D. Nelson, T.R. Garrison, W.E. Miller, R.J. Lefkowitz, Desensitization, internalization, and signaling functions of beta-arrestins demonstrated by RNA interference, Proc. Natl. Acad. Sci.USA 100 (2003) 1740–1744.

[39] J.D. Debes, L.J. Schmidt, H. Huang, D.J. Tindall, P300 mediates androgen-independent transactivation of the androgen receptor by interleukin 6, Cancer Res. 62 (2002) 5632–5636.
[40] E. Berra, E. Benizri, A. Ginouves, V. Volmat, D. Roux, J. Pouyssegur, HIF prolyl-hydroxylase 2 is the key oxygen sensor setting low steady-state levels of HIF-1alpha in normoxia, EMBO J. 22 (2003) 4082–4090.
[41] L.A. Martinez, I. Naguibneva, H. Lehrmann, A. Vervisch, T. Tchenio, G. Lozano, A. Harel-Bellan, Synthetic small inhibiting RNAs: efficient tools to inactivate oncogenic mutations and restore p53 pathways, Proc. Natl. Acad. Sci. USA 99 (2002) 14849–14854.
[42] T. Serwold, F. Gonzalez, J. Kim, R. Jacob, N. Shastri, ERAAP customizes peptides for MHC class I molecules in the endoplasmic reticulum, Nature 419 (2002) 480–483.
[43] X. Liu, R.L. Erikson, Activation of Cdc2/cyclin B and inhibition of centrosome amplification in cells depleted of Plk1 by siRNA, Proc. Natl. Acad. Sci. USA 99 (2002) 8672–8676.
[44] S. Moskalenko, D.O. Henry, C. Rosse, G. Mirey, J.H. Camonis, M.A. White, The exocyst is a Ral effector complex, Nat. Cell Biol. 4 (2002) 66–72.
[45] X.F. Qin, D.S. An, I.S. Chen, D. Baltimore, Inhibiting HIV-1 infection in human T cells by lentiviral-mediated delivery of small interfering RNA against CCR5, Proc. Natl. Acad. Sci. USA 100 (2003) 183–188.
[46] D. Cortez, S. Guntuku, J. Qin, S.J. Elledge, ATR and ATRIP: partners in checkpoint signaling, Science 294 (2001) 1713–1716.
[47] K. Yoshida, H.G. Wang, Y. Miki, D. Kufe, Protein kinase Cdelta is responsible for constitutive and DNA damage-induced phosphorylation of Rad9, EMBO J. 22 (2003) 1431–1441.
[48] L.M. Martins, I. Iaccarino, T. Tenev, S. Gschmeissner, N.F. Totty, N.R. Lemoine, J. Savopoulos, C.W. Gray, C.L. Creasy, C. Dingwall, J. Downward, The serine protease Omi/HtrA2 regulates apoptosis by binding XIAP through a reaper-like motif, J. Biol. Chem. 277 (2002) 439–444.

Paper VI

A comparison of siRNA efficacy predictors

Available online at www.sciencedirect.com
SCIENCE DIRECT®
Biochemical and Biophysical Research Communications 321 (2004) 247–253
BBRC
www.elsevier.com/locate/ybbrc

A comparison of siRNA efficacy predictors

Pål Sætrom, Ola Snøve Jr. *

Interagon AS, Medisinsk teknisk senter, NO-7489 Trondheim, Norway

Received 18 June 2004

Abstract

Short interfering RNA (siRNA) efficacy prediction algorithms aim to increase the probability of selecting target sites that are applicable for gene silencing by RNA interference. Many algorithms have been published recently, and they base their predictions on such different features as duplex stability, sequence characteristics, mRNA secondary structure, and target site uniqueness. We compare the performance of the algorithms on a collection of publicly available siRNAs. First, we show that our regularized genetic programming algorithm GPboost appears to have a higher and more stable performance than other algorithms on the collected datasets. Second, several algorithms gave close to random classification on unseen data, and only GPboost and three other algorithms have a reasonably high and stable performance on all parts of the dataset. Third, the results indicate that the siRNAs' sequence is sufficient input to siRNA efficacy algorithms, and that other features that have been suggested to be important may be indirectly captured by the sequence.

Keywords: siRNA; RNA interference; Efficacy prediction

RNA interference (RNAi) is a cellular process for sequence-specific depletion of mRNA [1]. Long double-stranded RNA duplexes or hairpin precursors are cleaved into short fragments by a ribonuclease III enzyme called Dicer. The resulting short interfering RNAs (siRNAs) are 21–23 nucleotides (nt) long and have characteristic 2 nt 3′ overhangs [2]. A ribonucleoprotein complex named RNA induced silencing complex (RISC) incorporates one of the siRNA strands, and cleaves mRNA with complementarity to the RNA component in an ATP-independent reaction [3]. Long RNA duplexes trigger the interferon response and yield non-specific degradation of mRNA when introduced into mammalian cells. The interferon response can, however, be circumvented by transfecting moderate concentrations of synthetic siRNAs into mammalian cells [4]. The knockdown effect is transient and diminishes after a few cell cycles [5]. A lasting knockdown effect can be obtained by endogenous transcription of hairpin precursors from vector [6] or virus-based [7] systems.

Several excellent reviews describe siRNA and RNAi [8–11].

The siRNAs must be optimized with respect to toxicity, specificity, and efficacy. First, both synthetic and endogenously transcribed siRNAs have been shown to induce the interferon response in a concentration-dependent manner [12–14]. Second, there is a risk that the siRNA may guide RISC to cleave mRNAs with sequence similarity to the target (shown indirectly in [15]) or that the siRNA may function as a microRNA and suppress protein translation [16]. Third, only a fraction of all siRNAs are effective at reducing the expression of their target genes, and two siRNAs that target mRNA sites that are separated by only a few nucleotides may have very different efficacies [5].

Genomewide specificity studies on the mRNA level have been published but the results are conflicting [14,17–19] and siRNAs' mismatch tolerance remains an open question. It seems clear, however, that central mismatches between the siRNA and the target mRNA

* Corresponding author. Fax: +47-23-01-12-35.
E-mail addresses: paal.saetrom@interagon.com (P. Sætrom), ola.snove@interagon.com (O. Snøve Jr.).

0006-291X/$ - see front matter © 2004 Elsevier Inc. All rights reserved.
doi:10.1016/j.bbrc.2004.06.116

are more likely to abolish silencing than mismatches at the ends, and that the tolerance for mismatches is higher at the 5′ end than at the 3′ end of the siRNA [15,20]. Very specific target sites are available for most genes but many published siRNAs have a flawed design and therefore risk off-target effects [21].

Algorithms that predict siRNA efficacy increase the probability for obtaining an siRNA that induces effective silencing of the desired gene. The Tuschl rules [22] were the only criteria available until Reynolds et al. [23] published their algorithm for rational design of effective siRNAs. Several other algorithms have emerged since [24–30]. We recently used a hardware accelerated [31] regularized genetic programming algorithm to develop siRNA efficacy classifiers [32]. We aim to provide a comparison of the algorithms' performance on a large collection of publicly available functionally validated siRNAs.

Materials and methods

Sequence data

We collected a non-redundant database of functionally validated siRNAs from seven publications [20,23–25,27,33,34]. The database contains 581 siRNAs that target 40 genes. Detailed information about the siRNAs, target genes, and the assays that were used when the siRNAs were validated is in Supplementary Table ST1. Note that the database is biased in that the selection of target genes and siRNAs has not been random in the works in which they were published. For example, Hsieh et al. [27] select siRNAs that comply with the Tuschl rules in addition to other criteria. Note also that the database contains fewer siRNAs with intermediate efficacies than would be expected if the selection was random. Moreover, one has to expect that there is considerable noise in the data due to (i) a variety of assays for measurement of siRNA efficacy; (ii) very different concentrations of siRNAs; and (iii) sub-optimal time intervals between transfection and down-regulation measurement. We aimed to limit the heterogeneity of the siRNA database; therefore, we included only datasets of a certain size with respect to either targets or siRNAs.

Algorithms

Both strands of the siRNA can potentially be absorbed by RISC to guide mRNA cleavage. The findings of Schwarz et al. [35] and Khvorova et al. [34] that RISC prefers the uptake of one strand based on the thermodynamic stability of an siRNA duplex provided a new criterion for design of effective siRNAs: The siRNA's thermodynamic properties must be such that the RISC prefers the incorporation of the strand that is complementary to the intended target site.

For the most part, siRNA efficacy prediction algorithms have been constructed by investigating single-base frequencies in relatively small datasets containing effective and ineffective siRNAs. Any statistically significant single-base correlations with efficacy, either positive or negative, are used to construct scoring algorithms [23–25,27,30]. (Note that Ui-Tei et al. [25] and Hsieh et al. [27] do not explicitly construct scoring algorithms in their papers. The sequence criteria that they do suggest, however, can easily be used to construct such an algorithm.)

Many authors have hypothesized that the accessibility of the mRNA target site determines siRNA efficacy as is the case for antisense DNA technologies. There are conflicting reports on whether target accessibility is a determinant for siRNA efficacy [26,36]. The differing results may be due to unreliable in silico secondary structure predictions or small and biased datasets. Luo and Chang [26] recently proposed an algorithm that predicts siRNA efficacy based on the target site's secondary structure.

Pancoska et al. [28] speculate that a sequence segment's uniqueness compared with the rest of the targeted mRNA and the duplex melting temperature determines the efficacy of an siRNA targeting that particular site. Unfortunately, it was not possible to reproduce their algorithm from the original publication, and we therefore decided to omit the algorithm from our comparisons.

We recently used a regularized genetic programming approach to obtain patterns that discriminated between effective and ineffective siRNAs [32]. We hypothesized that complex sequence patterns can capture all the information necessary to predict the efficacy of siRNAs and constructed classifiers whose score is a weighted sum of many patterns (see [32] for details).

Table 1 shows an overview of the features that the design algorithms rely on to make an efficacy prediction. Note that the thermodynamic stability of an RNA duplex is calculated from its sequence composition [37]. Table 2 shows how various algorithms score an siRNA based on individual nucleotides. For example, Reynolds 1+2 adds one to the score if the second sense strand nucleotide is adenine, whereas they subtract one if the fifteenth nucleotide is guanine. Note that many of the algorithms that are based on sequence characteristics prefer certain bases at the ends of the siRNA, which is probably because it yields the right difference between the 5′ and 3′ thermodynamic duplex stability. Reynolds 1+2 also adds one to the score if the siRNA's GC-content is between 30% and 50%. In addition to the single-base scores in Table 2, Ui-Tei counts the number of AU- and GC-pairs in positions 13–19, and adds one, respectively, subtracts one from the score if there are five or more AU- or five or more GC-pairs. Moreover, stretches of nine or more GC-pairs are considered negative and one is subtracted from the score, whereas one is added to the score if no such stretches are present.

Implementation details

Reynolds 1. We use the mfold web server [38] instead of the Oligo 6.0 software to predict the siRNA antisense melting temperature. We use a cutoff of 57 °C, as this both best mirrors previous results [23] and gives the highest absolute correlation on the Reynolds training data ($r = -0.14$).

Reynolds 2. This is the algorithm of Reynolds et al. [23] without the hairpin melting temperature scoring.

Table 1
There are important differences between the siRNA design algorithms

Algorithm	Citation	Description
GPboost	[32]	Weighted sum of sequence motifs/patterns
Ui-Tei	[25]	Sequence features
Amarzguioui	[24]	Sequence features
Hsieh	[27]	Sequence features
Takasaki	[30]	Sequence features
Reynolds 1	[23]	Hairpin potential, sequence features
Reynolds 2	[23]	Sequence features
Schwarz	[35]	Difference between 3′ and 5′ stability
Khvorova	[34]	Duplex stability profile
Stockholm 1	[29]	Energy features
Stockholm 2	[29]	Energy features
Tree	[29]	Sequence features in decision tree
Luo	[26]	mRNA secondary structure features

See Implementation details for additional information on the different algorithms.

Table 2
Sequence characteristics used by different algorithms

Algorithm	siRNA sense strand position																															
	1				2		3		6		7		8		9		10	11		13		15		16		17		18		19			
	A	C	G	U	A	U	A	U	A	C	G	U	A	G	G	U	U	C	G	A	G	A	U	A	G	U	A	U	A	C	G	U
Reynolds 1 and 2							1										1				−1	1	1	1		1	1	1	2	−1	−1	1
Ui-Tei	−1	1	1	−1																									1	−1	−1	1
Amarzguioui	−1	1	1	−2	−1	−1	−1	−1	1																1	1	1	1	2		−1	2
Hsieh										−1								1	1	1				1							−1	1
Takasaki	−3.97		7.4	−3.75					2.33		2.4	−2.59	3.02	−2.35	−2.35	2.3						2.7									−2	

Schwarz. We compute the duplex stability [37] for the four first nucleotides in the antisense and sense strands and use the difference as the classification score.

Khvorova. This algorithm creates two average internal stability profiles from a set of training sequences—one for effective siRNAs and another for ineffective siRNAs. Then, a siRNA's score is the difference of the correlation between its internal stability profile and the average effective and average ineffective siRNA profiles. The internal stability profile is found by computing the duplex stability [34] for each pentamer in the sequence.

Stockholm 1. This is our implementation of the Stockholm rules as described in [29]. We use the mfold web server [38] to predict the total hairpin energy and the nearest neighbor parameters of Xia et al. [37] for duplex stability calculations.

Stockholm 2. This is the modified Stockholm rules from the web server of Chalk et al. [29] (http://sisearch.cgb.ki.se/). In our experiments, we ran the prediction server with as few restrictions as possible, but some of the siRNAs in our database were still not evaluated. The web server missed about the same percentage of effective and ineffective siRNAs.

Tree. This is the decision tree score from the web server of Chalk et al. [29], with the low, moderate, and high categories mapped to 0, 1, and 2.

Comparing algorithms

We use the correlation between classifier output and siRNA efficacy, and ROC analysis to measure the performance of the different classifiers (see [39] for a review). The correlation R measures the classifier's overall performance: R^2 represents the proportion of variation in the observed efficacy that can be explained by the classifier. A Student's t test gives the statistical significance of a given correlation.

ROC analysis requires that all siRNAs are classified as either effective or ineffective, typically by using a cutoff on the measured siRNA efficacy. Given such a classification, a prediction made by a classifier can be either a true positive, a false positive, a true negative, or a false negative. That is, an effective siRNA will either be a true positive or a false negative prediction depending on what cutoff the classifier uses to signal positive predictions.

A ROC-curve is constructed by varying the classifier's positive cutoff and plotting the relative number of true positives and false positives identified by the classifier at each cutoff. This shows the classifier's sensitivity Se for varying levels of specificity Sp, as the relative number of false positives is $1 - Sp$. The ROC-score is the area under the ROC-curve and can be used to characterize a classifier's performance. Perfect classifiers identify all true positives before returning the false positives and have a ROC-score of 1.0; random classifiers return relatively as many false as true positives at each cutoff and have a ROC-score of 0.5.

We use the ROCKIT software [40] for statistical ROC analysis.

Results

The GPboost classifier is significantly better than the energy-based classifiers

We trained the GPboost and Khvorova classifiers on the training sets used to train the Ui-Tei, Amarzguioui, Hsieh, and Reynolds algorithms. The training set also included the 14 SEAP siRNAs from Khvorova et al. [34], for a total of 453 unique siRNA sequences. We classified all siRNAs that gave a remaining mRNA level of $\leqslant$20% as effective and the other siRNAs as ineffective. This gave 141 effective and 252 ineffective siRNAs.

We used 10-fold cross-validation to get an estimate of the algorithms' predictive accuracy, and measured the total ROC-score and correlation between algorithm output and siRNA efficacy in the 10 cross-validation test sets. This resulted in correlations −0.47, −0.39, and −0.23, and ROC-scores of 0.77, 0.69, and 0.63 for the GPboost, Schwarz, and Khvorova algorithms on the complete training set.

As the ROC-curves in Fig. 1 show, the GPboost classifier has higher sensitivity than the other two classifiers for all specificity levels. Indeed, the GPboost classifier's ROC-area is significantly greater than the ROC-areas of the other two classifiers ($p = 0.002$ and $p < 10^{-4}$ for the Schwarz and Khvorova classifiers). We also tested whether the GPboost classifier had a significantly higher sensitivity compared to the other two algorithms, in the important high specificity region (specificities 95%, 90%, 85%, and 80%). The GPboost classifier was better than that of Schwarz on 95% specificity ($p = 0.07$), and was significantly better (95% confidence level) than both classifiers on all other specificities.

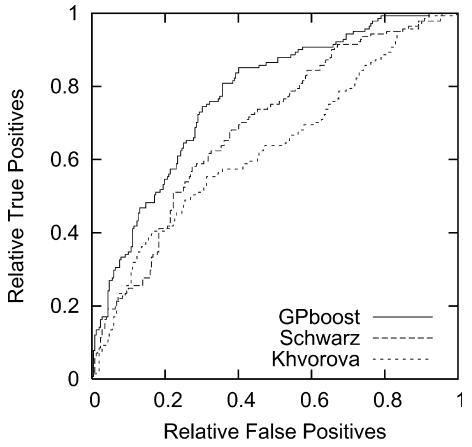

Fig. 1. ROC graphs for the GPboost, Schwarz, and Khvorova classifiers on the complete training set. The graphs are based on the test results from the 10-fold cross-validation procedure. The GPboost classifier has the highest sensitivity for all specificity levels.

The GPboost classifier has the best performance

It is often reasonable to expect that algorithms will be positively biased on their own training data as compared to independent test data. Indeed, when we tested the algorithms on their corresponding training data, the performance in terms of ROC-area and correlation was higher than the performance on the rest of the database (data not shown). The only exception was the Reynolds algorithms, which had a higher correlation on the rest of the database than on their training set. All the algorithms had a higher performance on their training sets than algorithms that were trained on other datasets (data not shown).

Table 3 shows the performance of the different classifiers when tested on the subsets of the database that did not include their corresponding training sets. Each classifier's performance is compared to the GPboost classifier's performance on the same data. Fig. 2 shows the Amarzguioui and Reynolds algorithms' ROC-curves compared to those of the GPboost classifiers. The ROC-curves for the other algorithms are in Supplementary figure SF1.

A closer inspection of the ROC-curves in Figs. 1 and 2 shows that the GPboost classifier generally has the best performance. It has the highest sensitivity for all specificity levels when compared to all the other algorithms. The ROC-curves and ROC-scores also show that some of the classifiers perform only slightly better than random. This is the case for the Luo classifier [26] and the modified Stockholm rules and decision tree of [29] from http://sisearch.cgb.ki.se/.

Statistical tests that compared the GPboost classifier to the other algorithms showed that the GPboost classifier

Table 3
Algorithm performance compared to that of the GPboost classifier

Algorithm	|siRNAs|		Algorithm		GPboost		
	$\|P\|$	$\|N\|$	ROC	R	ROC	R	p
Ui-Tei	112	229	0.65	−0.34	0.74	−0.42	0.008
Amarzguioui	107	206	0.72	−0.47	0.79	−0.48	0.05
Hsieh	140	145	0.67	−0.34	0.77	−0.50	0.02
Takasaki	137	242	0.62	−0.25	0.78	−0.48	$<10^{-4}$
Reynolds 1	53	161	0.64	−0.44	0.78	−0.46	0.0008
Reynolds 2	53	161	0.66	−0.46	0.78	−0.46	0.003
Stockholm 1	50	154	0.65	−0.31	0.78	−0.45	0.002
Stockholm 2	36	104	0.56	−0.21	0.78	−0.45	$<10^{-4}$
Tree	36	104	0.51	−0.24	0.78	−0.45	$<10^{-4}$
Luo	137	232	0.55	−0.14	0.78	−0.48	$<10^{-4}$

The algorithm performance is measured on the subset of the large training database that was not used to train the respective algorithm. $|P|$ and $|N|$ are the number of effective and ineffective siRNAs in the different sets; p is the p value for the test whether the GPboost classifier's ROC-score is significantly greater than that of the corresponding algorithm.

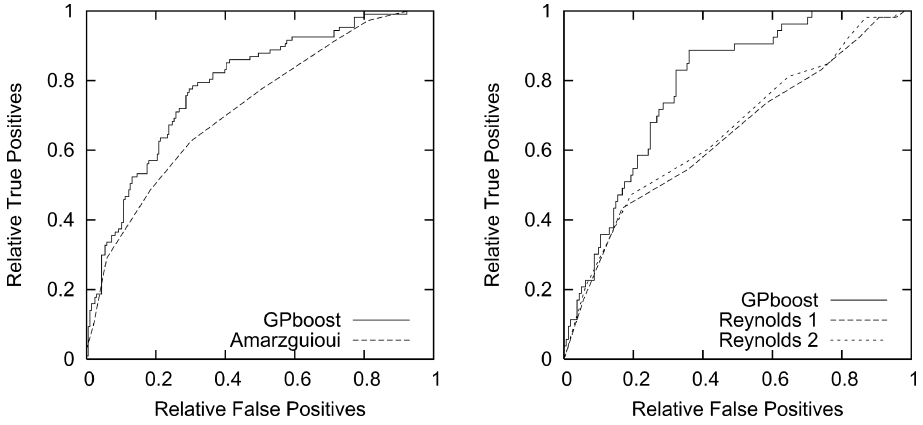

Fig. 2. The ROC graphs for the GPboost classifiers compared to those of the Amarzguioui and Reynolds classifiers; the ROC-curves for the other algorithms are in Supplementary Figure SF1. The GPboost classifier has the highest sensitivity for all specificity levels. The graphs were generated from different subsets of the large training database; see Table 3 and the main text for details.

had a significantly higher ROC-area than all the other algorithms (95% confidence level; p values in Table 3). Tests also showed that only the Amarzguioui and Reynolds algorithms have a performance that is comparable (95% confidence level) to that of the GPboost classifier in the high specificity region (the Amarzguioui and Reynolds 2 classifiers had p values 0.2, 0.1, 0.09, and 0.07, and 0.5, 0.3, 0.1, and 0.04 on specificities 95%, 90%, 85%, and 80%). Based on these results, one would expect that the GPboost classifier identifies more effective siRNAs.

Few classifiers have a stable and high performance

To further evaluate the classifiers' performance, we tested the different classifiers on three other datasets: the test set used by Reynolds et al. [23] to test their algorithm, the dataset of Harborth et al. [20], and the dataset of Vickers et al. [33]. To the best of our knowledge, none of these datasets were used to train any of the algorithms, except for the Vickers set, which was used to train the classifiers of Chalk et al. [29]. Since these sets are fairly large, come from three different sources, and have been generated using three different methods, they should give a fair estimate of the different classifiers' performance on unknown data.

Because the datasets were generated using different methods, and to get a representative number of effective and ineffective siRNAs in each set, we used different cutoffs for classifying the siRNAs as effective and ineffective. That is, we used 20%, 50%, and 10% for the Reynolds, Vickers, and Harborth data. This resulted in 17, 18, and 25 effective siRNAs, and 43, 58, and 19 ineffective siRNAs in the respective sets. Because of limitations in the web server of Chalk et al. [29], the Stockholm 2 and Tree classifiers were only tested on 13, 11, and 22 effective, and 32, 36, and 14 ineffective siRNAs.

Table 4 and Fig. 3 summarize the results on the three test sets (ROC-curves for the Vickers and Harborth data are in Supplementary figure SF2). The table and figure show that (i) the GPboost algorithm has the highest ROC-score on all datasets; (ii) only the GPboost, Amarzguioui, Ui-Tei, and Reynolds classifiers have a stable and high performance; and (iii) the performance of the remaining algorithms varies from random classification to intermediate performance. The Schwarz and Khvorova classifiers reach the performance of the best classifiers, but only on two of the three test sets.

Table 4
Results on the three independent test sets

Algorithm	Reynolds [23]		Vickers [33]		Harborth [20]	
	ROC	R	ROC	R	ROC	R
GPboost	0.84	−0.55	0.83	−0.35	0.82	−0.43
Ui-Tei	0.75	−0.47	0.77	−0.58	0.79	−0.31
Amarzguioui	0.75	−0.45	0.80	−0.47	0.76	−0.34
Hsieh	0.56	−0.03	0.51	−0.15	0.66	−0.17
Takasaki	0.49	−0.03	0.62	−0.25	0.51	0.01
Reynolds 1	0.70	−0.35	0.73	−0.47	0.79	−0.23
Reynolds 2	0.70	−0.37	0.71	−0.44	0.79	−0.23
Schwarz	0.71	−0.29	0.72	−0.35	0.51	0.01
Khvorova	0.68	−0.15	0.77	−0.19	0.60	−0.11
Stockholm 1	0.56	−0.05	0.58	−0.18	0.64	−0.28
Stockholm 2	0.63	0.00	0.56	−0.15	0.69	−0.41
Tree	0.50	−0.11	0.68	−0.43	0.54	0.06
Luo	0.50	−0.33	0.54	−0.27	0.71	−0.40

The GPboost algorithm has the highest ROC-score on all test sets and only a few algorithms (outlined in gray) have a stable, high performance on all sets.

Effective siRNAs are identified by sequence alone

The results for the Luo algorithm deserve some discussion. On most datasets, the algorithm has a ROC-score that is close to random classification, but at the same time the correlation between the algorithm's output and the siRNA efficacy can be well above random. Indeed, all the reported correlations for the Luo algorithm are

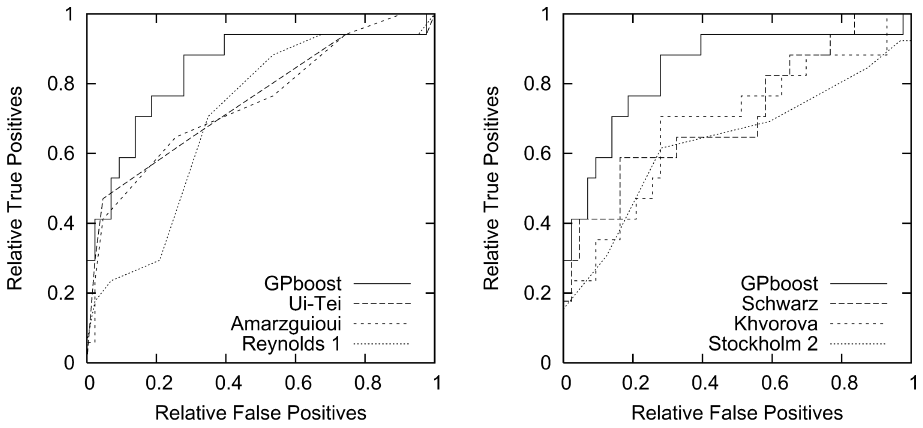

Fig. 3. ROC graphs for the seven highest scoring algorithms [23–25,29,32,34,35] on the Reynolds test sets. The GPboost classifier has the highest sensitivity for almost all specificity levels when compared to the other algorithms.

significant at the 95% confidence level. One possible explanation is that the mRNA secondary structure is important for siRNA efficacy, but that it is only a secondary effect compared to the siRNA sequence-based features, such as the duplex differential 5′/3′ free energy or sequence motifs. We tried to combine the Luo classifier with the GPboost classifier, which gave a small but insignificant improvement (the 10-fold cross-validation correlation and ROC-score were increased by approximately 0.02 and 0.005). Thus, it seems that on the data we examined here, highly effective siRNAs can be identified by the siRNA sequence alone, and that the secondary structure of the mRNA target sequence has limited influence on siRNA efficacy.

Discussion

We have shown that our regularized genetic programming approach (GPboost) [32] performs better than other published siRNA efficacy algorithms on a large collection of functionally validated siRNAs. We believe that the GPboost algorithm has a higher performance because (i) the algorithm was trained on a larger set of siRNAs than the other algorithms; (ii) the algorithm uses patterns that capture more complex characteristics of effective siRNAs than do the simpler motif algorithms; and (iii) the algorithm is very robust when it comes to noise in the training data, as, for instance, siRNAs that have been erroneously labeled as effective or ineffective.

Surprisingly, several algorithms gave close to random classification, and only the GPboost, Reynolds, Amarzguioui, and Ui-Tei algorithms have a high and stable performance on the whole dataset. This suggests that over-fitting is a problem with many algorithms, and that proper care needs to be taken when estimating the classification accuracy to avoid such effects.

The results suggest that it may not be critical to consider the target site's secondary structure, as the best algorithms only consider the sequence alone. Our analysis suggests that mRNA secondary structure has a minor influence on siRNA efficacy, but that highly effective siRNAs can be selected based on target sequence alone. This fact has not been proven, however, so secondary structure should still be investigated when analyzing new data.

We expect that the dataset we used is biased, as the siRNAs have not been randomly selected in the publications in which they appeared. Even so, we believe that the results of our comparison will generalize to other data as well, since all of the algorithms we investigated were trained on subsets of this dataset.

The RNAi field is maturing rapidly, and new siRNA efficacy prediction algorithms will emerge partly due to larger and better datasets. We expect that the need for a large publicly available set of randomly selected validated siRNAs will rise as more algorithms are published, since it is difficult to objectively compare their performance without an independent test set.

Acknowledgments

We thank A. Khvorova for providing details from [23], and H.E. Krokan, T. Holen, T.B. Grünfeld, and O.R. Birkeland for valuable comments on the manuscript.

Appendix A. Supplementary material

Supplementary data associated with this article can be found, in the online version, at doi:10.1016/j.bbrc.2004.06.116.

References

[1] A. Fire, S. Xu, M. Montgommery, S. Kostas, S. Driver, C. Mello, Potent and specific genetic interference by double-stranded RNA in *Caenorhabditis elegans*, Nature 391 (6593) (1998) 806–811.

[2] P. Zamore, T. Tuschl, P. Sharp, D. Bartel, RNAi: double-stranded RNA directs the ATP-dependent cleavage of mRNA at 21 to 23 nucleotide intervals, Cell 101 (1) (2000) 25–33.
[3] A. Nykanen, B. Haley, P. Zamore, ATP requirements and small interfering RNA structure in the RNA interference pathway, Cell 107 (3) (2001) 309–321.
[4] S. Elbashir, J. Harborth, W. Lendeckel, A. Yalcin, K. Weber, T. Tuschl, Duplexes of 21-nucleotide RNAs mediate RNA interference in cultured mammalian cells, Nature 411 (6836) (2001) 494–498.
[5] T. Holen, M. Amarzguioui, M.T. Wiiger, E. Babaie, H. Prydz, Positional effects of short interfering RNAs targeting the human coagulation trigger tissue factor, Nucleic Acids Res. 30 (8) (2002) 1757–1766.
[6] T. Brummelkamp, R. Bernards, R. Agami, A system for stable expression of short interfering RNAs in mammalian cells, Science 296 (5567) (2002) 550–553.
[7] D. Rubinson, C. Dillon, A. Kwiatkowski, C. Sievers, L. Yang, J. Kopinja, M. Zhang, M. McManus, F. Gertler, M. Scott, L. Parijs, A lentivirus-based system to functionally silence genes in primary mammalian cells, stem cells and transgenic mice by RNA interference, Nat. Genet. 33 (3) (2003) 401–406.
[8] D. Dykxhoorn, C. Novina, P. Sharp, Killing the messenger: short RNAs that silence gene expression, Nat. Rev. Mol. Cell Biol. 4 (6) (2003) 457–467.
[9] M. McManus, P. Sharp, Gene silencing in mammals by small interfering RNAs, Nat. Rev. Genet. 3 (10) (2002) 737–747.
[10] P. Zamore, RNA interference: listening to the sound of silence, Nat. Struct. Biol. 8 (9) (2001) 746–750.
[11] G. Hannon, RNA interference, Nature 418 (6894) (2002) 244–251.
[12] C. Sledz, M. Holko, M. de Veer, R. Silverman, B. Williams, Activation of the interferon system by short-interfering RNAs, Nat. Cell Biol. 5 (9) (2003) 834–839.
[13] A. Bridge, S. Pebernard, A. Ducraux, A.-L. Nicoulaz, R. Iggo, Induction of an interferon response by RNAi vectors in mammalian cells, Nat. Genet. 34 (3) (2003) 263–264.
[14] S. Persengiev, X. Zhu, M. Green, Nonspecific, concentration-dependent stimulation and repression of mammalian gene expression by small interfering RNAs, RNA 10 (1) (2004) 12–18.
[15] M. Amarzguioui, T. Holen, E. Babaie, H. Prydz, Tolerance for mutations and chemical modifications in a siRNA, Nucleic Acids Res. 31 (2) (2003) 589–595.
[16] J. Doench, C. Petersen, P. Sharp, SiRNAs can function as miRNAs, Genes Dev. 17 (4) (2003) 438–442.
[17] D. Semizarov, L. Frost, A. Sarthy, P. Kroeger, D. Halbert, S. Fesik, Specificity of short interfering RNA determined through gene expression signatures, Proc. Natl. Acad. Sci. USA 100 (11) (2003) 6347–6352.
[18] J.-T. Chi, H. Chang, N. Wang, D. Chang, N. Dunphy, P. Brown, Genomewide view of gene silencing by small interfering RNAs, Proc. Natl. Acad. Sci. USA 100 (11) (2003) 6343–6346.
[19] A. Jackson, S. Bartz, J. Schelter, S. Kobayashi, J. Burchard, M. Mao, B. Li, G. Cavet, P. Linsley, Expression profiling reveals off-target gene regulation by RNAi, Nat. Biotechnol. 21 (6) (2003) 635–637.
[20] J. Harborth, S.M. Elbashir, K. Vandenburgh, H. Manninga, S.A. Scaringe, K. Weber, T. Tuschl, Sequence, chemical, and structural variation of small interfering RNAs and short hairpin RNAs and the effect on mammalian gene silencing, Antisense Nucleic Acid Drug Dev. 13 (2003) 83–106.
[21] O. Snøve, T. Holen, Many commonly used siRNAs risk off-target activity, Biochem. Biophys. Res. Commun. 319 (1) (2004) 256–263.
[22] S. Elbashir, J. Harborth, K. Weber, T. Tuschl, Analysis of gene function in somatic mammalian cells using small interfering RNAs, Methods 26 (2) (2002) 199–213.
[23] A. Reynolds, D. Leake, Q. Boese, S. Scaringe, W.S. Marshall, A. Khvorova, Rational siRNA design for RNA interference, Nat. Biotechnol. 22 (3) (2004) 326–330.
[24] M. Amarzguioui, H. Prydz, An algorithm for selection of functional siRNA sequences, Biochem. Biophys. Res. Commun. 316 (4) (2004) 1050–1058.
[25] K. Ui-Tei, Y. Naito, F. Takahashi, T. Haraguchi, H. Ohki-Hamazaki, A. Juni, R. Ueda, K. Saigo, Guidelines for the selection of highly effective siRNA sequences for mammalian and chick RNA interference, Nucleic Acids Res. 32 (3) (2004) 936–948.
[26] K. Luo, D. Chang, The gene-silencing efficiency of siRNA is strongly dependent on the local structure of mRNA at the targeted region, Biochem. Biophys. Res. Commun. 318 (1) (2004) 303–310.
[27] A. Hsieh, R. Bo, J. Manola, F. Vazquez, O. Bare, A. Khvorova, S. Scaringe, W. Sellers, A library of siRNA duplexes targeting the phosphoinositide 3-kinase pathway: determinants of gene silencing for use in cell-based screens, Nucleic Acids Res. 32 (3) (2004) 893–901.
[28] P. Pancoska, Z. Moravek, U. Moll, Efficient RNA interference depends on global context of the target sequence: quantitative analysis of silencing efficiency using Eulerian graph representation of siRNA, Nucleic Acids Res. 32 (4) (2004) 1469–1479.
[29] A. Chalk, C. Wahlestedt, E. Sonnhammer, Improved and automated prediction of effective siRNA, Biochem. Biophys. Res. Commun. 319 (1) (2004) 264–274.
[30] S. Takasaki, S. Kotani, A. Konagaya, An effective method for selecting siRNA target sequences in mammalian cells, Cell Cycle, (2004) Epub ahead of print.
[31] A. Halaas, B. Svingen, M. Nedland, P. Sætrom, O. Snøve, O.R. Birkeland, A recursive MISD architecture for pattern matching, IEEE Trans. VLSI Syst. 12 (7) (2004) 727–734.
[32] P. Sætrom, Predicting the efficacy of short oligonucleotides in antisense and RNAi experiments with boosted genetic programming, Bioinformatics, (2004) Epub ahead of print.
[33] T.A. Vickers, S. Koo, C.F. Bennett, S.T. Crooke, N.M. Dean, B.F. Baker, Efficient reduction of target RNAs by small interfering RNA and RNase H-dependent antisense agents. A comparative analysis, J. Biol. Chem. 278 (9) (2003) 7108–7118.
[34] A. Khvorova, A. Reynolds, S.D. Jayasena, Functional siRNAs and miRNAs exhibit strand bias, Cell 115 (2003) 209–216.
[35] D.S. Schwarz, G. Hutvágner, T. Du, Z. Xu, N. Aronin, P.D. Zamore, Asymmetry in the assembly of the RNAi enzyme complex, Cell 115 (2003) 199–208.
[36] K. Yoshinari, M. Miyagishi, K. Taira, Effects on RNAi of the tight structure, sequence and position of the targeted region, Nucleic Acids Res. 32 (2) (2004) 691–699.
[37] T. Xia, J. SantaLucia Jr., M.E. Burkard, R. Kierzek, S.J. Schroeder, X. Jiao, C. Cox, D.H. Turner, Thermodynamic parameters for an expanded nearest-neighbor model for formation of RNA duplexes with Watson–Crick base pairs, Biochemistry 37 (1998) 14719–14735.
[38] M. Zuker, Mfold web server for nucleic acid folding and hybridization prediction, Nucleic Acids Res. 31 (13) (2003) 3406–3415.
[39] P. Baldi, S. Brunak, Y. Chauvin, C. Andersen, H. Nielsen, Assessing the accuracy of prediction algorithms for classification: an overview, Bioinformatics 16 (5) (2000) 412–424.
[40] C.E. Metz, B.A. Herman, C.A. Roe, Statistical comparison of two ROC-curve estimates obtained from partially-paired datasets, Med. Decis. Making 18 (1) (1998) 110–121.

Paper VII

Designing effective siRNAs with off-target control

Available online at www.sciencedirect.com
SCIENCE DIRECT®
Biochemical and Biophysical Research Communications 325 (2004) 769–773
BBRC
www.elsevier.com/locate/ybbrc

Designing effective siRNAs with off-target control

Ola Snøve Jr.[1], Magnar Nedland[1], Ståle H. Fjeldstad, Håkon Humberset, Olaf R. Birkeland, Thomas Grünfeld, Pål Sætrom*

Interagon AS, Medisinsk teknisk senter, NO-7489 Trondheim, Norway

Received 7 October 2004
Available online 6 November 2004

Abstract

Successful gene silencing by RNA interference requires a potent and specific depletion of the target mRNA. Target candidates must be chosen so that their corresponding short interfering RNAs are likely to be effective against that target and unlikely to accidentally silence other transcripts due to sequence similarity. We show that both effective and unique targets exist in mouse, fruitfly, and worm, and present a new design tool that enables users to make the trade-off between efficacy and uniqueness. The tool lists all targets with partial sequence similarity to the primary target to highlight candidates for negative controls.

Keywords: siRNA design; RNAi; Gene silencing; Efficacy prediction; Uniqueness; Specificity; Off-target effects

Sequence-specific knockdown of mRNA is a naturally occurring mechanism in many organisms: posttranscriptional gene silencing in plants [1], quelling in fungi [2], and RNA interference (RNAi) in flies [3], nematodes [4], and mammals [5]. They all have in common that Dicer, a ribonuclease III enzyme, initiates the silencing pathways by cleavage of long double-stranded RNA into shorter duplexes [6]. These short interfering RNAs (siRNAs) are 21–23 nucleotides long and have characteristic 3′ overhangs of two nucleotides [7]. The thermodynamic properties of the siRNA determine which of the two strands is incorporated into the RNA induced silencing complex [8,9], a ribonucleoprotein complex that mediates sequence-specific cleavage of mRNA by recognition of sites complementary to its RNA component [10].

The sequence-specificity of RNAi is still unclear. For example, one group reported that a single central mismatch between the siRNA and its target mRNA is enough to abolish silencing in *Drosophila* [3], whereas another group published conflicting results [11]. Yet another group has shown that siRNAs targeting the human tissue factor generally tolerated single mismatches but that mismatches at the 3′ end of the strand complementary to the mRNA were more harmful than 5′ mismatches [12]. Microarray approaches have not been able to settle the controversy as both widespread [13,14] and non-existing [15,16] off-target gene regulation has been reported. Moreover, there is also a risk that siRNAs may function as microRNAs [17,18], a class of non-coding RNAs that are incorporated in a ribonucleoprotein complex called microRNP that may repress protein translation of mRNA with partial complementarity to the microRNA (see [19] for a review).

Even the earliest siRNA design rules stated that potential sequences should be checked for similarity with

* Corresponding author. Fax: +47 455 94 458.
E-mail addresses: ola.snove@interagon.com (O. Snøve Jr.), magnar.nedland@interagon.com (M. Nedland), staale.fjeldstad@interagon.com (S.H. Fjeldstad), haakon.humberset@interagon.com (H. Humberset), olaf.birkeland@interagon.com (O.R. Birkeland), thomas.grunfeld@interagon.com (T. Grünfeld), paal.saetrom@interagon.com (P. Sætrom).
[1] These authors contributed equally to this work.

0006-291X/$ - see front matter © 2004 Elsevier Inc. All rights reserved.
doi:10.1016/j.bbrc.2004.10.097

other genes to ensure that only a single transcript is targeted [20]. We recently showed, however, that most commonly used siRNAs risk off-target gene regulation due to sequence similarity with other transcripts [21].

About one in five randomly selected siRNAs will be effective at silencing their targets. Several criteria and algorithms for rational design of siRNAs have been proposed [20,8,9,22–30]. The methods aim to increase the probability of selecting effective siRNAs. We recently reported that only the Reynolds et al. [22], Ui-Tei et al. [24], Amarzguioui and Prydz [27], and Sætrom (GPboost) [30] methods seem to have a high and stable performance across several independent datasets [31].

Several commercial vendors have offered pools of siRNAs targeting the same transcript to increase the probability of getting target knockdown. (Note that a pool of four siRNAs where each has a 50% probability of being effective has an accumulated 94% chance of being effective assuming independent probabilities.) This approach may not be appropriate, at least not for therapeutic purposes. First, the risk for off-target gene regulation increases with pooled siRNAs as more potential targets with (partial) similarity exist. Second, even siRNAs have been shown to trigger the interferon response [32,33], apparently in a concentration-dependent manner [14]. Third, RNAi may be prone to saturation, which means that unprocessed siRNAs remain in the cell free to enter other cellular pathways [34]. It is therefore important to find targets that are effectively silenced at the lowest possible concentrations and pooled approaches may not be amenable to this requirement.

We aim to bridge the gap between efficacy algorithms and uniqueness requirements and will show that many siRNA target sites that are predicted to be highly effective and sufficiently unique are available for most targets. An online application where the users are able to make qualified trade-offs between predicted efficacy and risk for off-target activity accompanies the results and will be presented throughout this article.

Materials and methods

Datasets. For this study, we performed complete off-target screenings on the mouse, fruitfly, and worm transcriptomes from Ensembl [35]; more specifically, *Mus musculus* version 32b (NCBI: m32), *Drosophila melanogaster* version 3a (NCBI: BGDP 3.1), and *Caenorhabditis elegans* version 116a (NCBI: WS 116).

Hardware. Our online server runs Debian Woody Linux on a 2.8 GHz Pentium 4 processor with 1024 MB RAM and special purpose search processors. The Pattern Matching Chip (PMC; Interagon AS, Trondheim, Norway) is an application-specific integrated circuit (ASIC) designed to provide orders of magnitude higher performance than that of comparable regular expression matchers [36]. The server is equipped with five PCI cards, which amounts to 80 PMCs and a capacity to screen 64 complex patterns against 8.0 GB/s.

Uniqueness algorithm. We evaluate the risk for off-target effects by screening siRNAs for uniqueness in the transcriptome using our special purpose search processors. BLAST [37] is not applicable for this purpose as it is prone to miss potentially important matches when the queries are short [21]. Other heuristics such as FASTA [38], ParAlign [39], and PatternHunter [40] use similar pruning schemes as BLAST to avoid searching parts that are unlikely to contain matches; thus, only the time-consuming Smith–Waterman algorithm [41] is guaranteed to yield complete results. In this particular application we run ungapped Smith–Waterman with special treatment of G:U wobble basepairing; more advanced constructs with insertions and deletions as well as weighting of mismatch positions are also possible using our hardware.

Efficacy algorithms. Several siRNA efficacy predictors run on our online server, including that of Sætrom [30], Amarzguioui and Prydz [27], Hsieh et al. [23], Reynolds et al. [22], Schwarz et al. [8], Chalk et al. [28], Takasaki et al. [29], and Ui-Tei et al. [24]. The algorithms are implemented as previously described by our group in [31].

Availability. Our online demo version screens the mouse transcriptome and is available on http://www.interagon.com/demo/ (requires registration). Full siRNA libraries are available for all sequenced species in commercial and academic partnerships.

Results

Our siRNA design tool is largely based on our previous work with siRNA efficacy [31] and off-target risk [21]. Fig. 1 shows several screenshots from the demo version that is available online.

We have previously shown that unique siRNAs are available, at least for the human transcriptome [21], and that four publicly available efficacy algorithms have a high and stable performance across several datasets [31]. But how many sufficiently unique siRNAs have a high efficacy prediction, and vice versa? Assuming that the probabilities p_u that a sequence is unique and p_e that a sequence is effective are independent yields the probability $p_u p_e$ that the sequence is both unique and effective (according to efficacy predictors).

Table 1 shows the average value of p_e with 95% confidence intervals for *M. musculus*, *D. melanogaster*, and *C. elegans*, given different levels of specificities for the GPboost efficacy predictor [30]. (The output of a GPboost classifier is on a scale from -1 to 1, and 19mers with scores above a threshold value are regarded as effective. Increasing the threshold yields higher specificity, but at the expense of a lower sensitivity.) The values are the average of p_e as has been exhaustively computed for different levels of specificity under the assumption that efficacy is independent of uniqueness.

Fig. 2 shows that unique 19mers will be available for most transcripts of a certain length on all levels up to three mismatches; that is, the siRNAs are unique for one target site even if up to three mismatches are allowed between the siRNA and other sites in the transcriptome. The average transcript of 2000 bp will therefore contain more than one unique siRNA on all uniqueness levels except for (3,0) even if you allow only one percent false negative efficacy predictions ($Sp = 0.99$). Note that about half of the transcripts are expected to contain effective siRNAs at the (3,0) level.

ID	Target sequence	Eff.	0mm	1mm	2mm	3mm	4mm	Un.
0	ACCGTACTAACCAGGGAAG	0.052	1	0	0	1	74	7.9
1	CACCGTACTAACCAGGGAA	0.082	1	0	0	1	89	7.9
2	AGTCAGATCCTAGCGTCGA	0.055	1	0	0	2	41	7.8
3	CCCCGGACGATATTGAACA	0.1	1	0	0	2	88	7.8
4	AGTCTACCTCCCGCCATAA	0.14	1	0	0	2	76	7.8
5	GGCCCACTTCACCGTACTA	0.14	1	0	0	2	82	7.8
6	AGGGTTAGTTTACAATCAG	0.12	1	0	0	3	99	7.7
7	AGTTAAGGGTTAGTTTACA	0.23	1	0	0	3	103	7.7
8	GGTAATCTACTGGGACGGA	0.2	1	0	0	2	60	7.8
9	GAAATTTGCGTGTGGAGTA	0.45	1	0	0	3	108	7.7
10	CGATATTGAACAATGGTTC	0.17	1	0	0	3	103	7.7
11	GCTGTCCCCGGACGATATT	0.23	1	0	0	3	79	7.7
12	GACACTTTGCGTTCGGGCT	0.049	1	0	0	3	82	7.7
13	CGCTAAAAGTTTTGAGCTT	0.32	1	0	0	3	102	7.7
14	ACGATATTGAACAATGGTT	0.31	1	0	0	4	130	7.6
15	CTGTGACTTGCACGTACTC	0.047	1	0	0	4	121	7.6
16	GGAAATTTGCGTGTGGAGT	0.42	1	0	0	4	107	7.6
17	AGTGGTAATCTACTGGGAC	0.08	1	0	0	4	132	7.6
18	CGTGAGCGCTTCGAGATGT	0.1	1	0	0	3	71	7.7
19	GTGAGCGCTTCGAGATGTT	0.36	1	0	0	4	105	7.6

Fig. 1. Screenshots of (A) the selection screen where the users input the RNA sequence or accession number and choose an siRNA efficacy predictor; (B) the scatter plot showing predicted efficacy versus uniqueness for each siRNA; (C) the result overview where the siRNAs are ranked according to predicted efficacy and uniqueness; and (D) the alignment of the ranked siRNAs with potential off-target regions.

Table 1
p_e with 95% confidence intervals when the siRNA efficacy predictor has specificities of 0.99, 0.95, 0.90, and 0.75

Species	Specificity			
	0.99	0.95	0.90	0.75
M. musculus	0.127 ± 0.008	0.214 ± 0.011	0.282 ± 0.012	0.392 ± 0.012
D. melanogaster	0.120 ± 0.002	0.203 ± 0.003	0.270 ± 0.005	0.378 ± 0.009
C. elegans	0.184 ± 0.034	0.294 ± 0.049	0.376 ± 0.056	0.498 ± 0.063

Surprisingly, the assumption that efficacy and uniqueness are independent features may not be entirely valid as demonstrated by the line for *C. elegans* in Fig. 3. These are the true dependencies as the curves have been calculated exhaustively; that is, we have determined both predicted efficacy and uniqueness level for all

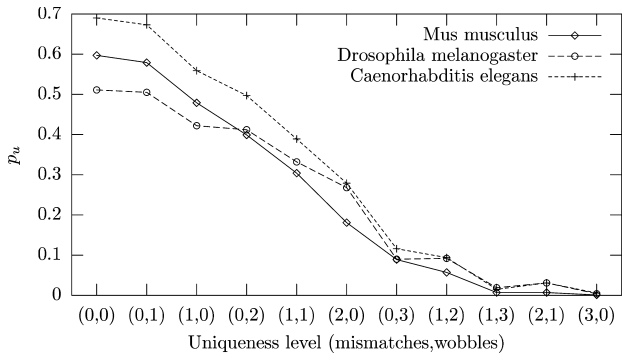

Fig. 2. Probabilities of an siRNA being unique even if a certain number of mismatches and G:U wobbles are allowed between the siRNA and its target.

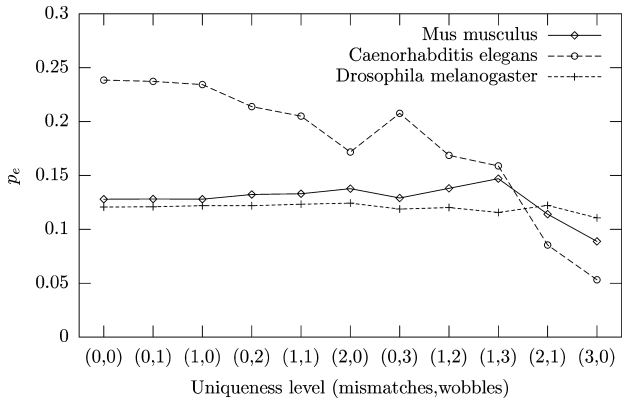

Fig. 3. p_e depends on the uniqueness level, but shows small variations. The given curves for p_e are given for specificity 0.99 for the efficacy predictions; other specificities show similar results (data not shown).

19mers in the given transcriptomes. The averages presented in Table 1 should, however, represent good approximations as the differences in p_e are relatively small. Moreover, it appears that *C. elegans* has a higher fraction of effective siRNAs, and that efficacy decreases with higher uniqueness, whereas *D. melanogaster* and *M. musculus* do not show this dependency. We hypothesized that the lower GC-content in *C. elegans* resulted in higher efficacy prediction values; however, such a dependency was not confirmed when generating random 19mers with identical base composition (data not shown).

It may not be important if a siRNA unintentionally targets a mRNA that is known to be unrelated to the pathway under study. We therefore list all potential off-target matches so that the users are able to evaluate the risk with respect to the biology of their experiments.

Discussion

We have shown that the average transcript of 2000 bp in *M. musculus*, *D. melanogaster*, and *C. elegans* will contain several siRNAs that are both unique and effective. Thus, effective RNAi with risk of off-target effects should be viable for most transcripts from these genomes.

The presented siRNA design tool lists all candidates for off-target mRNA depletion, given that the mechanism depends only on mismatches and G:U wobbles between siRNA and target. We suggest that targets on this list become candidates for negative controls in silencing experiments. We propose, however, that the potential for translational repression by microRNAs will become an even bigger challenge in siRNA design. We therefore work on including algorithms for microRNA off-target effect predictions in future versions of the design tool.

Acknowledgments

We thank O. Sætrom, F. Drabløs, and A. Halaas for valuable comments on the manuscript.

References

[1] D. Baulcombe, Fast forward genetics based on virus-induced gene silencing, Curr. Opin. Plant Biol. 2 (2) (1999) 109–113.
[2] N. Romano, G. Macino, Quelling: transient inactivation of gene expression in *Neurospora crassa* by transformation with homologous sequences, Mol. Microbiol. 6 (22) (1992) 3343–3353.
[3] S. Elbashir, J. Martinez, A. Patkaniowska, W. Lendeckel, T. Tuschl, Functional anatomy of siRNAs for mediating efficient RNAi in *Drosophila melanogaster* embryo lysates, EMBO J. 20 (23) (2001) 6877–6888.
[4] A. Fire, S. Xu, M. Montgommery, S. Kostas, S. Driver, C. Mello, Potent and specific genetic interference by double-stranded RNA in *Caenorhabditis elegans*, Nature 391 (6593) (1998) 806–811.
[5] S. Elbashir, J. Harborth, W. Lendeckel, A. Yalcin, K. Weber, T. Tuschl, Duplexes of 21-nucleotide RNAs mediate RNA interference in cultured mammalian cells, Nature 411 (6836) (2001) 494–498.
[6] E. Bernstein, A. Caudy, S. Hammond, G.J. Hannon, Role for a bidentate ribonuclease in the initiation step of RNA interference, Nature 409 (6818) (2001) 295–296.
[7] P. Zamore, T. Tuschl, P. Sharp, D. Bartel, RNAi: double-stranded RNA directs the ATP-dependent cleavage of mRNA at 21 to 23 nucleotide intervals, Cell 101 (1) (2000) 25–33.
[8] D.S. Schwarz, G. Hutvágner, T. Du, Z. Xu, N. Aronin, P.D. Zamore, Asymmetry in the assembly of the RNAi enzyme complex, Cell 115 (2003) 199–208.
[9] A. Khvorova, A. Reynolds, S.D. Jayasena, Functional siRNAs and miRNAs exhibit strand bias, Cell 115 (2003) 209–216.
[10] J. Martinez, T. Tuschl, RISC is a 5′ phosphomonoester-producing rna endonuclease, Genes Dev. 18 (9) (2004) 975–980.
[11] A. Boutla, C. Delidakis, I. Livadaras, M. Tsagris, M. Tabler, Short 5′-phosphorylated double-stranded RNAs induce RNA interference in *Drosophila*, Curr. Biol. 11 (22) (2001) 1776–1780.
[12] M. Amarzguioui, T. Holen, E. Babaie, H. Prydz, Tolerance for mutations and chemical modifications in a siRNA, Nucleic Acids Res. 31 (2) (2003) 589–595.
[13] A. Jackson, S. Bartz, J. Schelter, S. Kobayashi, J. Burchard, M. Mao, B. Li, G. Cavet, P. Linsley, Expression profiling reveals off-

target gene regulation by RNAi, Nat. Biotechnol. 21 (6) (2003) 635–637.
[14] S. Persengiev, X. Zhu, M. Green, Nonspecific, concentration-dependent stimulation and repression of mammalian gene expression by small interfering RNAs, RNA 10 (1) (2004) 12–18.
[15] J.-T. Chi, H. Chang, N. Wang, D. Chang, N. Dunphy, P. Brown, Genomewide view of gene silencing by small interfering RNAs, Proc. Natl. Acad. Sci. USA 100 (11) (2003) 6343–6346.
[16] D. Semizarov, L. Frost, A. Sarthy, P. Kroeger, D. Halbert, S. Fesik, Specificity of short interfering RNA determined through gene expression signatures, Proc. Natl. Acad. Sci. USA 100 (11) (2003) 6347–6352.
[17] J. Doench, C. Petersen, P. Sharp, siRNAs can function as miRNAs, Genes Dev. 17 (4) (2003) 438–442.
[18] S. Saxena, Z. Jonsson, A. Dutta, Implications for off-target activity of small inhibitory RNA in mammalian cells, J. Biol. Chem. 278 (45) (2003) 44312–44319.
[19] D.P. Bartel, Micro RNAs: genomics, biogenesis, mechanism, and function, Cell 116 (2) (2004) 281–297.
[20] S. Elbashir, J. Harborth, K. Weber, T. Tuschl, Analysis of gene function in somatic mammalian cells using small interfering RNAs, Methods 26 (2) (2002) 199–213.
[21] O. Snøve Jr., T. Holen, Many commonly used siRNAs risk off-target activity, Biochem. Biophys. Res. Commun. 319 (1) (2004) 256–263.
[22] A. Reynolds, D. Leake, Q. Boese, S. Scaringe, W.S. Marshall, A. Khvorova, Rational siRNA design for RNA interference, Nat. Biotechnol. 22 (3) (2004) 326–330.
[23] A. Hsieh, R. Bo, J. Manola, F. Vazquez, O. Bare, A. Khvorova, S. Scaringe, W. Sellers, A library of siRNA duplexes targeting the phosphoinositide 3-kinase pathway: determinants of gene silencing for use in cell-based screens, Nucleic Acids Res. 32 (3) (2004) 893–901.
[24] K. Ui-Tei, Y. Naito, F. Takahashi, T. Haraguchi, H. Ohki-Hamazaki, A. Juni, R. Ueda, K. Saigo, Guidelines for the selection of highly effective siRNA sequences for mammalian and chick RNA interference, Nucleic Acids Res. 32 (3) (2004) 936–948.
[25] K. Luo, D. Chang, The gene-silencing efficiency of siRNA is strongly dependent on the local structure of mRNA at the targeted region, Biochem. Biophys. Res. Commun. 318 (1) (2004) 303–310.
[26] P. Pancoska, Z. Moravek, U. Moll, Efficient RNA interference depends on global context of the target sequence: quantitative analysis of silencing efficiency using Eulerian graph representation of siRNA, Nucleic Acids Res. 32 (4) (2004) 1469–1479.
[27] M. Amarzguioui, H. Prydz, An algorithm for selection of functional siRNA sequences, Biochem. Biophys. Res. Commun. 316 (4) (2004) 1050–1058.
[28] A. Chalk, C. Wahlestedt, E. Sonnhammer, Improved and automated prediction of effective siRNA, Biochem. Biophys. Res. Commun. 319 (1) (2004) 264–274.
[29] S. Takasaki, S. Kotani, A. Konagaya, An effective method for selecting siRNA target sequences in mammalian cells, Cell Cycle, Epub ahead of print.
[30] P. Sætrom, Predicting the efficacy of short oligonucleotides in antisense and RNAi experiments with boosted genetic programming, Bioinformatics (in press), Epub ahead of print).
[31] P. Sætrom, O. Snøve Jr., A comparison of siRNA efficacy predictors, Biochem. Biophys. Res. Commun. 321 (1) (2004) 247–253.
[32] C. Sledz, M. Holko, M. de Veer, R. Silverman, B. Williams, Activation of the interferon system by short-interfering RNAs, Nat. Cell Biol. 5 (9) (2003) 834–839.
[33] A. Bridge, S. Pebernard, A. Ducraux, A.-L. Nicoulaz, R. Iggo, Induction of an interferon response by RNAi vectors in mammalian cells, Nat. Genet. 34 (3) (2003) 263–264.
[34] L. Scherer, J. Rossi, Approaches for the sequence-specific knock-down of mRNA, Nat. Biotechnol. 21 (12) (2003) 1457–1465.
[35] E. Birney, D. Andrews, P. Bevan, M. Caccamo, G. Cameron, Y. Chen, L. Clarke, G. Coates, T. Cox, J. Cuff, V. Curwen, T. Cutts, T. Down, R. Durbin, E. Eyras, X. Fernandez-Suarez, P. Gane, B. Gibbins, J. Gilbert, M. Hammond, H. Hotz, V. Iyer, A. Kahari, K. Jekosch, A. Kasprzyk, D. Keefe, S. Keenan, H. Lehvaslaiho, G. McVicker, C. Melsopp, P. Meidl, E. Mongin, R. Pettett, S. Potter, G. Proctor, M. Rae, S. Searle, G. Slater, D. Smedley, J. Smith, W. Spooner, A. Stabenau, J. Stalker, D. Storey, A. Ureta-Vidal, C. Woodwark, M. Clamp, T. Hubbard, Ensembl 2004, Nucleic Acids Res. 32 (1) (2004) 468–470.
[36] A. Halaas, B. Svingen, M. Nedland, P. Sætrom, O. Snøve Jr., O.R. Birkeland, A recursive MISD architecture for pattern matching, IEEE Trans. VLSI Syst. 12 (7) (2004) 727–734.
[37] S. Altschul, W. Gish, W. Miller, E. Myers, D. Lipman, Basic local alignment search tool, J. Mol. Biol. 215 (3) (1990) 403–410.
[38] W. Pearson, D. Lipman, Improved tools for biological sequence comparison, J. Mol. Biol. 85 (8) (1988) 2444–2448.
[39] T. Rognes, ParAlign: a parallel sequence alignment algorithm for rapid and sensitive database searches, Nucleic Acids Res. 29 (7) (2001) 1647–1652.
[40] B. Ma, J. Tromp, M. Li, PatternHunter: faster and more sensitive homology search, Bioinformatics 18 (3) (2002) 440–445.
[41] T. Smith, M. Waterman, Identification of common molecular subsequences, J. Mol. Biol. 147 (1) (1981) 403–410.

Paper VIII

Weighted sequence motifs as an improved seeding step in microRNA target prediction algorithms

Weighted sequence motifs as an improved seeding step in microRNA target prediction algorithms

OLA SÆTROM,[1] OLA SNØVE JR.,[2] and PÅL SÆTROM[2]

[1]Department of Computer and Information Science, Norwegian University of Science and Technology, Trondheim, Norway
[2]Interagon AS, Medisinsk teknisk senter, NO-7489 Trondheim, Norway

ABSTRACT

We present a new microRNA target prediction algorithm called TargetBoost, and show that the algorithm is stable and identifies more true targets than do existing algorithms. TargetBoost uses machine learning on a set of validated microRNA targets in lower organisms to create weighted sequence motifs that capture the binding characteristics between microRNAs and their targets. Existing algorithms require candidates to have (1) near-perfect complementarity between microRNAs′ 5′ end and their targets; (2) relatively high thermodynamic duplex stability; (3) multiple target sites in the target's 3′ UTR; and (4) evolutionary conservation of the target between species. Most algorithms use one of the two first requirements in a seeding step, and use the three others as filters to improve the method's specificity. The initial seeding step determines an algorithm's sensitivity and also influences its specificity. As all algorithms may add filters to increase the specificity, we propose that methods should be compared before such filtering. We show that TargetBoost's weighted sequence motif approach is favorable to using both the duplex stability and the sequence complementarity steps. (TargetBoost is available as a Web tool from http://www.interagon.com/demo/.)

Keywords: miRNA target prediction; genetic programming; boosting; machine learning

INTRODUCTION

MicroRNAs (miRNAs) belong to an abundant class of short noncoding RNAs (Lagos-Quintana et al. 2001; Lau et al. 2001; Lee and Ambros 2001) shown to mediate suppression of protein translation (Moss et al. 1997; Olsen and Ambros 1999; Reinhart et al. 2000) and cleavage of mRNA (Zeng et al. 2002; Yekta et al. 2004). Homologs exist across many species (Pasquinelli et al. 2000), which shows that miRNAs' function as gene regulators has been conserved through evolution. A total of 1340 miRNA genes from 11 species are listed in the 5.0 release of the miRNA registry (Griffiths-Jones 2004). Computational approaches have estimated that about 1% of all predicted genes in the human (Lim et al. 2003a), fruitfly (Lai et al. 2003), and worm (Lim et al. 2003b) genomes are miRNA genes.

It seems that miRNAs function as siRNAs and silence genes by mRNA cleavage when targets with near-perfect complementarity exist (Zeng et al. 2002; Yekta et al. 2004), whereas inhibition of translation occurs when miRNAs are only partially complementary to their targets (Lee et al. 1993; Wightman et al. 1993). MicroRNAs known to induce translational suppression predominantly target 3′ UTRs (Bartel 2004) with neighboring binding sites (Olsen and Ambros 1999; Reinhart et al. 2000), but it has been demonstrated that a siRNA targeting a single coding site with partial complementarity can induce translational suppression as well (Saxena et al. 2003). Regardless, the inhibition of protein synthesis is more effective when targeting multiple sites (Doench et al. 2003).

Several miRNA target prediction algorithms have appeared recently, and results for fruitfly (Enright et al. 2003; Stark et al. 2003; Rajewsky and Socci 2004) and mammals (Lewis et al. 2003; John et al. 2004; Kiriakidou et al. 2004) suggest that about 10% of protein-coding genes are regulated by miRNAs (John et al. 2004). Computational approaches for identifying miRNA targets generally use sequence complementarity, thermodynamic stability calculations, and evolutionary conservation among species to determine whether a miRNA:mRNA duplex is a likely target interaction (Bartel 2004; Lai 2004).

The RNAhybrid algorithm by Rehmsmeier et al. (2004) computes minimum-free energy hybridization sites for miRNAs, while forcing perfect complementarity in nucleotides

Reprint requests to: Pål Sætrom, Interagon AS, Medisinsk teknisk senter, NO-7489 Trondheim, Norway; e-mail: paal.saetrom@interagon.com.

Article published online ahead of print. Article and publication date are at http://www.rnajournal.org/cgi/doi/10.1261/rna.7290705.

(nt) 2–7. Potential sites are normalized by the product of a miRNA and its potential target to avoid high-scoring, but unlikely hybridizations to long target sequences. Extreme value statistics similar to that used in sequence-similarity searching is used to determine the likelihood of a candidate site being due to random hits in a large database. The DIANA-microT algorithm also minimizes the duplex-binding energy in its initial step (Kiriakidou et al. 2004). Most of the existing miRNA target prediction algorithms use similar thermodynamic calculations in post-processing steps following a requirement of near-perfect complementarity with the targets in the miRNAs' 5′ ends.

Rajewsky and Socci (2004) define a binding nucleus of consecutive base pairs, and calculate a weighted sum typically consisting of six to eight addends favoring more hydrogen bonds. The algorithm is referred to as Nucleus throughout this article, and its position-specific weights differ only slightly from the weights of a similar algorithm called miRanda (Enright et al. 2003). In a subsequent post-processing step, Nucleus uses folding-free energy as determined by mfold (Zuker 2003) to make the final predictions. Simpler algorithms that use a seed of perfect complementarity in the miRNA's 5′ region include TargetScan (Lewis et al. 2003) and an algorithm from EMBL (Stark et al. 2003), but these run the risk of loosing targets that do not exactly meet their seed criteria.

We have developed a machine-learning algorithm called TargetBoost that creates classifiers for predicting miRNA target sites, and this is a novel approach to miRNA target site prediction. The algorithm, which is an adaptation of the boosted genetic programming algorithm of Sætrom (2004), creates weighted sequence motifs that characterize the probable binding characteristics between miRNAs and target sites. That is, given a miRNA and a potential target site, this classifier returns a score that represents the likelihood of the site being targeted by the miRNA. We used our classifiers to predict target sites in a set of genes important for fly body patterning in *Drosophila melanogaster*.

TargetBoost compares favorably to the algorithms of Rajewsky and Socci (2004) and Rehmsmeier et al. (2004) that were described previously. First, it rediscovers that miRNAs' 5′ ends bind well to targets. Second, it proves to be a classifier with a high and stable performance across several targets. Third, and most importantly, it discovers more true targets than the aforementioned algorithms. As other known algorithms use variants of the Nucleus and RNAhybrid approaches, the performance of these two algorithms should be representative of the other algorithms' performance as well.

We have not included additional filters, such as requiring conservation of the target sites or the presence of multiple target sites in the 3′ UTRs, in our algorithm comparisons. The reason is that these filters can be used independently of the initial method used to predict the target sites. Thus, improving the quality of the initial candidates will also improve the final predictions.

In summary, our main contributions are a new algorithm for predicting miRNA target sites, and an objective comparison of its performance to that of existing algorithms.

RESULTS

A machine learning algorithm that predicts miRNA target sites

GPboost is a machine-learning algorithm that, from a training set of positive and negative sequences, creates a sequence-based classifier that recognizes the positive sequences (Sætrom 2004). The classifier is the sum of several differentially weighted sequence patterns, where each pattern answers either yes (1) or no (−1) as to whether the pattern matches a given sequence or not. We have previously used variants of GPboost to predict the efficacy of short interfering RNAs (Sætrom 2004; Sætrom and Snøve Jr. 2004) and noncoding RNA genes in *Escherichia coli* (P. Sætrom, R. Sneve, K.I. Kristiansen, O. Snøve Jr., T. Grünfeld, T. Rognes, and E. Seeberg, in prep.).

To create the classifier, GPboost combines genetic programming (GP) (Koza 1992) and boosting (Meir and Rätsch 2003). More specifically, GP evolves the individual sequence patterns from a population of candidate patterns, and the boosting algorithm guides GP's search by adjusting the importance of each sequence in the training set. Then, the boosting algorithm assigns weights to the sequence patterns based on the patterns' performance in the corresponding training set. The final classifier is the average of several such boosted GP classifiers. Sætrom (2004) gives a more thorough description of the algorithm.

To train the miRNA target site predictors, we use a variant of the GPboost program, called TargetBoost, with two main differences. First, in Sætrom (2004) the patterns were simple queries, but the patterns we use here are template queries. That is, the sequence patterns are general expressions that describe the common properties of miRNA target sites. When using the patterns to search for target sites, we translate the general expressions into queries that are specific for each miRNA. Second, we use a different language to define what patterns are legal solutions. In the Materials and Methods, we give a formal definition of this pattern language along with additional details on how TargetBoost translates the patterns into miRNA-specific queries.

TargetBoost finds a good, stable miRNA target site predictor

To train and test the TargetBoost classifiers, we used a set of 36 experimentally verified target sites as positive data and a larger set of random sequences as negative data (see Materials and Methods for details). We compared TargetBoost's performance with the performance of Nucleus (Rajewsky and Socci 2004) and RNAhybrid (Rehmsmeier et al. 2004)—two recently published methods for identifying miRNA targets.

To test the algorithms, we used 10-fold and leave-one-miRNA-out cross-validation, and used receiver operating characteristics (ROC) analysis to compare the algorithms' performance; see Materials and Methods for further descriptions.

Figure 1 shows the 10-fold cross-validation ROC-curves for TargetBoost, RNAhybrid, and Nucleus. When comparing the curves for the different algorithms, we see that TargetBoost and RNAhybrid are better than Nucleus on high-specificity levels, with TargetBoost slightly better than RNAhybrid on specificity levels above 0.9.

Figure 2 shows the leave-one-miRNA-out cross-validation results as it displays the ROC-curves for TargetBoost, RNAhybrid, and Nucleus for each miRNA in the training set individually. We see that RNAhybrid and TargetBoost have approximately the same ROC-curves for every miRNA, with TargetBoost being slightly better for every miRNA except *miR-13a* and the high-specificity regions of *lin-4*. Nucleus has the highest performance for *lin-4*.

To compare the overall performance of the three algorithms, we computed the ROC-score for each algorithm on each miRNA. Then, on each individual miRNA, we tested whether the best algorithm was significantly better than the other algorithms. As Table 1 shows, TargetBoost not only had the best overall ROC-score, it was also the most stable of the three target site predictors, as for each individual miRNA, TargetBoost was either the best algorithm (*let-7* and *bantam*) or as good as the best algorithm (Nucleus for *lin-4* and RNAhybrid for *miR-13a*). Both RNAhybrid and Nucleus, however, were significantly worse than the best algorithm on at least one miRNA.

Although the overall performance of the classifiers is important, when using a classifier to predict miRNA target sites in genes, the most important characteristics of the classifier is that the top predictions made by the classifier have a high probability of being true target sites. That is, the best classifier has higher sensitivity than the other classifiers when approaching maximal specificity.

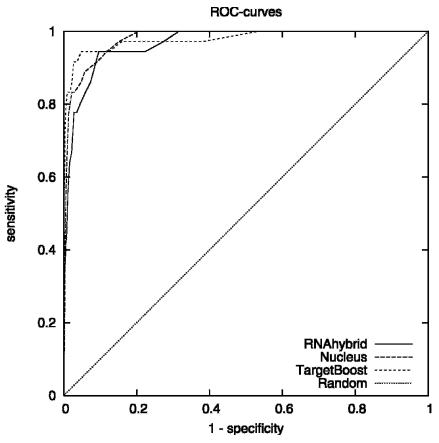

FIGURE 1. Overall ROC-curves for each algorithm.

The true-positive frequency (TPF) test determines whether there is a significant difference in the sensitivity of two classifiers at a given significance level (see Materials and Methods). For each miRNA, we tested whether the best classifier was significantly more sensitive than the other classifiers (99% confidence level) on specificities 0.995, 0.99, 0.98, 0.97, 0.96, and 0.95. On all specificities, TargetBoost was either the best or as good as the highest-scoring algorithm on all genes. RNAhybrid performed well on all specificities for all miRNAs except *lin-4*, where the algorithm was significantly less sensitive than Nucleus on all specificities. Nucleus, however, suffered from lower sensitivity in the high-specificity area; TargetBoost was significantly more sensitive than Nucleus for *let-7* (specificity 0.995—*P*-value 0.006) and *miR-13a* (specificities 0.995 and 0.99—*P*-values 0.005 and 0.006). Thus, as for the overall ROC-score, TargetBoost was the most stable of the three algorithms.

A possible explanation for RNAhybrid and Nucleus being less stable than TargetBoost is that the different miRNAs have slightly different binding characteristics. For example, *lin-4* and its target sites have a lower binding energy compared with the three other miRNAs, but may have other characteristics that the sequence-based methods, TargetBoost and Nucleus, have used to identify the target sites. This can explain RNAhybrid's poorer performance on this miRNA. The motif-based classifiers of TargetBoost, however, seem to be robust and capture both the thermodynamic and sequence characteristics of the miRNA target sites in our database.

TargetBoost finds more true target sites than do RNAhybrid and Nucleus

When we search for target sites, there will be far more negative than positive target sites. We are therefore interested in a classifier that finds as many positive target sites as possible, before the number of negative target sites in the result set becomes too large. The ROC_{50}-score, which is the area under the ROC curve until 50 false positives are found, reflects this interest, as the score takes into account that a user is seldom concerned with true positives that occur after the first page (about 50) of false positives (Gribskov and Robinson 1996). We ran a ROC_{50} test on the different algorithms to compare their performance on low frequencies of false positives; Table 2 lists the scores.

We found that TargetBoost performs better than RNAhybrid and Nucleus. Both RNAhybrid and Nucleus can be given extra constrains, such as forcing miRNA 5′ helices in RNAhybrid and increasing the free-energy cutoff in Nucleus, to improve their predictive power. Both perform much better when they are given extra constraints, and

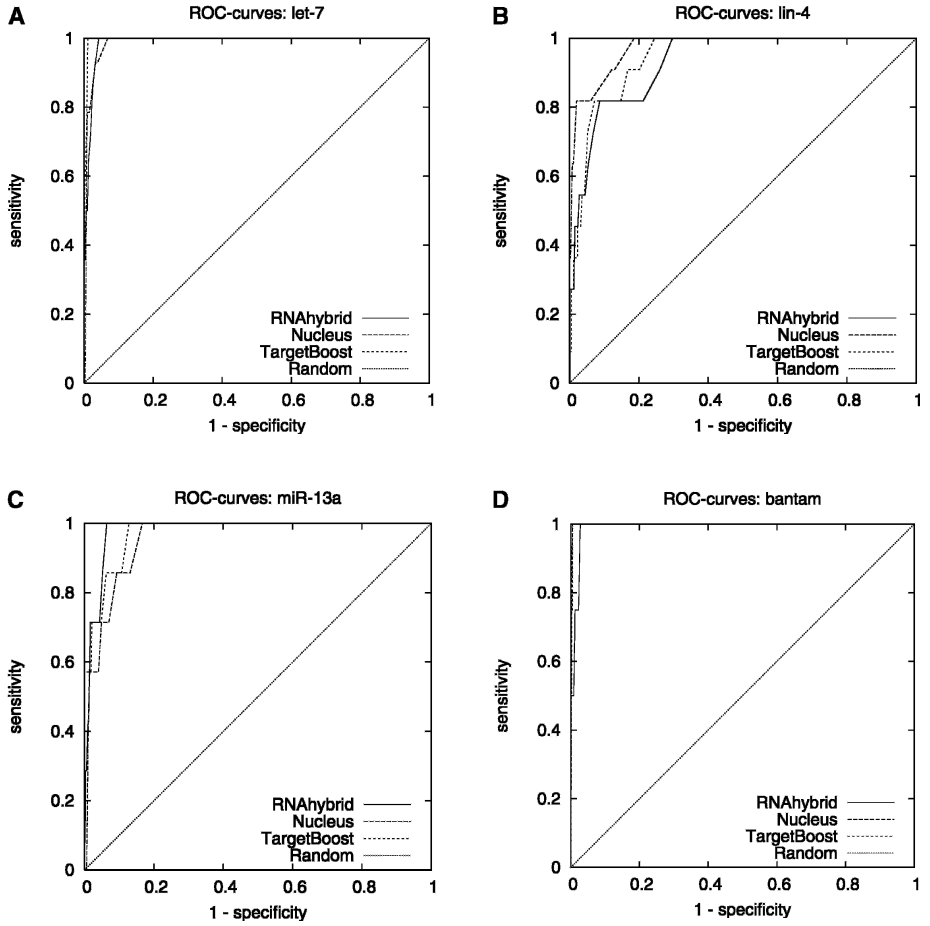

FIGURE 2. ROC-curves. *A–D* compare the performance of RNAhybrid, Nucleus, and TargetBoost at predicting the true target sites of *let-7*, *lin-4*, *miR-13a*, and *bantam*.

especially RNAhybrid get a much higher sensitivity for high levels of specificity (see Fig. 3A). The drawback is that the algorithms will miss several miRNA target sites when they are using these constraints. Figure 3 gives the complete ROC-curves for the different versions of RNAhybrid and Nucleus. TargetBoost does not have this problem, as each target site will get a score by TargetBoost and no target site will automatically be discarded. What is more, as Table 2 shows, TargetBoost finds more true target sites, even when the constraints are introduced in RNAhybrid and Nucleus.

TABLE 1. TargetBoost is the most stable algorithm

Algorithm	*let-7*	*lin-4*	*miR-13a*	*bantam*	All
TargetBoost	**0.997**	**0.944**	**0.972**	**0.998**	0.979
RNAhybrid	**0.989**	0.931	**0.979**	**0.991**	0.967
Nucleus	0.988	**0.962**	0.928	**0.998**	0.973

ROC-scores that are not significantly different from the highest score on a particular miRNA are in boldface (90% confidence level; see Materials and Methods for details on each algorithm).

TargetBoost rediscovers that 5′ ends bind with near-perfect complementarity

Earlier methods that identify miRNA target sites have used the property that the miRNA tends to bind perfectly to the target site on the 5′ end of the miRNA. Enright et al. (2003), Kiriakidou et al. (2004), Lewis et al. (2003), and Stark et al. (2003) use this property directly by demanding perfect binding at the 5′ end as a seed. Nucleus (Rajewsky and Socci 2004) uses the property indirectly by demanding a long GC-rich sequence of matches. This sequence will most often appear at the 5′ end of the miRNA. RNAhybrid (Rehmsmeier et al. 2004) can also incorporate this property by demanding that parts of the miRNA have to form a perfect helix. By demanding a perfect helix on nt 2–7 on

TABLE 2. ROC_{50} scores for the algorithms on the complete data set

Algorithm	ROC_{50}-score
TargetBoost	0.0025
$RNAhybrid_1$	0.0012
$RNAhybrid_2$	0.0017
$Nucleus_1$	0.0006
$Nucleus_2$	0.0011
$Nucleus_3$	0.0014

See Materials and Methods for descriptions of the different algorithms.

the 5′ end of the miRNA, better results were observed (see Rehmsmeier et al. 2004; Fig. 3A; Table 2).

TargetBoost confirmed the tendency of perfect matching in the 5′ end. The production rules used to create the classifiers demand a segment of near-perfect pairing between miRNA and target site, but the position and length of this pairing is not encoded in the production. This is decided entirely by the training process. Almost every individual trained at the first boosting iteration resembled the expression in Figure 4. That is, in most expressions, the consecutive sequence part of the expressions (the rightmost {...} subexpression in Fig. 4) used positions 17–24 in the miRNA counted from the 3′ end. These positions correspond to the first eight bases on the 5′ end. As explained in the Materials and Methods, the $P \geq 6$ means that six of the eight bases have to match at the 5′ core, and this indicates that almost every target site in the training set demands a near-perfect match in the 5′ end of the miRNA. This corresponds to experimental evidence in the literature (Doench and Sharp 2004; Kiriakidou et al. 2004).

Target candidates in *Drosophila melanogaster*

We searched a set of genes important for fly body patterning in *D. melanogaster* for candidate target sites. This set is the same as was used in Rajewsky and Socci (2004) and Rehmsmeier et al. (2004). In the search, we used a set of 78 *D. melanogaster* miRNAs downloaded from the miRNA Registry version 5.0 (Griffiths-Jones 2004). We compared the target sites found in our search with the target sites predicted by Rehmsmeier et al. (2004) and Rajewsky and Socci (2004).

Figure 5 displays target sites predicted by either TargetBoost, RNAhybrid, or both. When comparing our results to the top five hits predicted by Rehmsmeier et al. (2004), we found that TargetBoost did not predict the potential *miR-92a* site in *tailless* and the potential *miR-210* site in *hairy* reported by RNAhybrid. This is because of the number of G:U wobbles in the target sites reported by RNAhybrid; for example, the *miR-92a* target in *tailless* has three G:U wobbles, two of them residing in the 5′ core (see Fig. 5). The *miR-210* site in *hairy* has five G:U wobbles, with three wobbles in the first eight bases of the 5′ core. As TargetBoost treats G:U wobbles as normal mismatches, we would not find potential target sites with a high number of G:U wobbles; especially if the sites resided in the 5′ core. This may, however, be a strength of our method, as recent experimental results suggest that G:U wobbles may be detrimental to translational repression (Doench and Sharp 2004).

Although we did not find the same *miR-210* site in *hairy* as did RNAhybrid, TargetBoost did predict that *miR-7* has a potential target site in *hairy*. The target site is the same as the ones predicted by RNAhybrid and Nucleus, and it has only one G:U wobble. Stark et al. (2003) has shown that *hairy* is a target for *miR-7*.

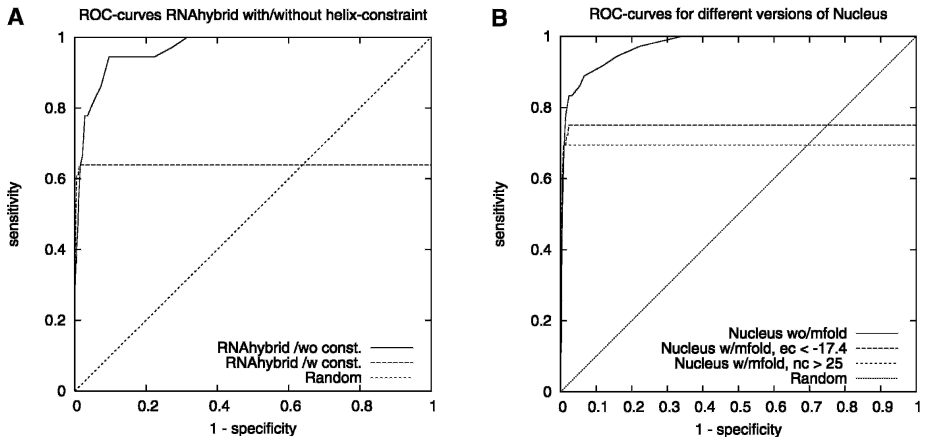

FIGURE 3. ROC-curves comparing different parameter settings on RNAhybrid (*A*) and Nucleus (*B*). We can see increased sensitivity for high-specificity values for RNAhybrid in *A*.

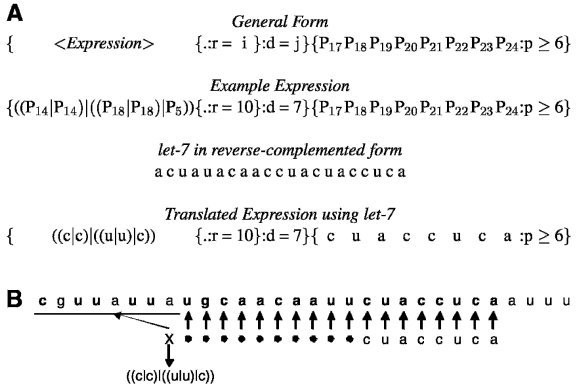

FIGURE 4. MicroRNA target query examples. (*A*) General pattern generated after the first boosting iteration in TargetBoost. This pattern has been translated using the *let-7* miRNA in a complemented form. (*B*) The translated query matched against the *lin-14* target site.

Other differences in the predicted target sites come from the constraint used in RNAhybrid. To get better predictions with RNAhybrid, you can demand a perfect helix for nt 2–7 in the 5′ end of the miRNA. TargetBoost does not need this constraint, and therefore a larger set of potential target sites will be considered with TargetBoost. For example, the *miR-9c* target in *crocodile* predicted by TargetBoost, shown in Figure 5, have a mismatch in position 5 at the 5′ end. Because of this, the target is automatically disqualified when running RNAhybrid with the perfect helix constraint.

Finally, Figure 5 shows the highest-scoring target site predicted by TargetBoost. *let-7* has few mismatches with this *buttonhead* target site, and the target site also has the characteristics of single miRNA target sites as outlined by Kiriakidou et al. (2004).

DISCUSSION

We have presented a program, TargetBoost, that finds miRNA target sites. We compared the performance of TargetBoost against two recently published algorithms for finding miRNA target sites, and found that the performance of TargetBoost is good and stable compared with the other algorithms. A possible reason for this is that TargetBoost has found a pattern in the miRNA–mRNA binding that predicts target sites better than just looking at the free-energy score and binding in the 5′ core. It is known that by incorporating knowledge of binding in the 5′ core to the free-energy calculation, better classification is achieved (Rehmsmeier et al. 2004). Perhaps by discovering other patterns in the miRNA–mRNA binding and incorporating those, TargetBoost has made a better classifier.

Another potential explanation for TargetBoost's performance compared with the other algorithms is that TargetBoost does not allow G:U wobbles between miRNAs and target sites. Mutation studies of target sites (Doench and Sharp 2004) and miRNAs (Kloosterman et al. 2004) indicate that G:U wobbles in the 5′ region of the miRNA reduces target site activity more than what is expected by their thermodynamic stability. Thus, algorithms that rely on thermodynamic calculations to predict target sites will return more false-positive predictions.

Computational methods that predict miRNA target sites generally use sequence complementarity, thermodynamic stability calculations, evolutionary conservation among species, number of target sites in a mRNA, or a combination of the four. We chose to compare TargetBoost against Nucleus and RNAhybrid because these two algorithms cover both the group of algorithms that uses sequence complementarity and the group of algorithms that uses thermodynamic stability calculations. We have disregarded other methods to further refine the set of candidate sites, as evolutionary conservation and the number of target sites are used as a post-processing step on the more basic methods for finding candidate sites. They can therefore also easily be used as a post-processing step for TargetBoost.

Be aware that all miRNA target prediction algorithms are based on the assumption that all targets share characteristics with the set of experimentally verified targets in lower organisms. There is a possibility that (1) new families of targets with fundamentally different characteristics from the training set exist, and (2) targets in mammalian species differ from those of lower organisms. For example, Smalheiser and Torvik (2004) compared the complementarity interactions between miRNAs and mRNA with that between miRNAs and scrambled controls in humans. They found that the discriminative characteristics of putative targets are longer stretches of perfect complementarity, higher overall complementarity allowing for gaps, mismatches, and wobbles, and multiple proximal sites that are complementary to one or several miRNAs. Note that these results suggest that mammalian miRNA targets may possess other characteristics than do targets from

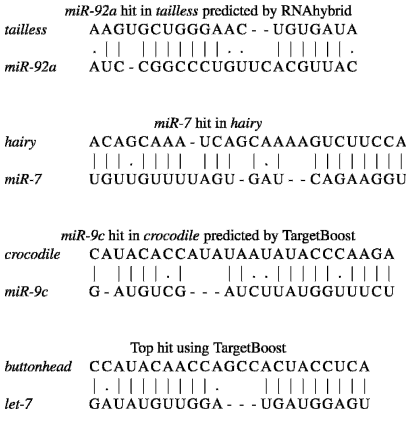

FIGURE 5. Aligned *D. melanogaster* target sites.

D. melanogaster and *Caenorhabditis elegans*. Specifically, the stretches of perfect complementarity may be longer, targets in the protein-coding region may be present, and the bias toward perfect complementarity in the miRNA's 5′ region may be weaker. If this is true, current miRNA target prediction algorithms may have limited value when used to predict targets in mammals.

In summary, we have presented a new algorithm for predicting miRNA target sites. The algorithm uses machine learning to train a sequence-based target site predictor, and this is a novel approach to miRNA target site prediction. Our algorithm compares favorably to other algorithms, both in terms of overall performance and when making highly specific predictions. We believe that our algorithm will be an important tool, not only for finding the target sites of known miRNAs, but also for predicting potential miRNA off-target effects in RNAi experiments (Saxena et al. 2003; Scacheri et al. 2004).

MATERIALS AND METHODS

Algorithm and implementation

TargetBoost ensures that all patterns evolved in the genetic programming process are valid expressions in a pattern language (Sætrom 2004). Figure 6 shows the grammar and semantics of the pattern language used to create the miRNA target predictors. The grammar is in Backus-Naur form (Knuth 1964) and shows the legal production rules in the language, with nonterminals represented by uppercase letters and terminals represented by boldface letters. Syntactical elements in the language, such as parentheses and operators, are in normal typeface, alternatives are represented as separate productions, adjacent symbols are concatenated, and P_i represents position i in the miRNA-sequence, counted from the 3′ end.

	Production	Semantic Rule
(1)	$S \rightarrow (D)(O)$	$S.hit := D.hit\ AND\ O.hit$
(2)	$D \rightarrow \{F : \mathrm{d} = N\}$	$D.hit := linger(F.hit, N)$
(3)	$F \rightarrow (R)(W)$	$F.hit := R.hit\ AND\ W.hit$
(4)	$R \rightarrow \{C : \mathrm{p} \geq N\}$	$R.hit := C.count \geq N.cutoff$
(5)	$R \rightarrow C$	$R.hit := C.hit$
(6)	$C \rightarrow (C_1)(C_2)$	$C.count := C_1.count + C_2.count$
		$C.hit := C_1.hit\ AND\ C_2.hit$
(7)	$C \rightarrow A$	$C.count := A.count$
		$C.hit := A.hit$
(8)	$A \rightarrow A_1 \mid A_2$	$A.count := A_1.count + A_2.count$
		$A.hit := A_1.hit\ OR\ A_2.hit$
(9)	$A \rightarrow L$	$A.count := L.count$
		$A.hit := L.hit$
(10)	$L \rightarrow \mathbf{a}$, for some $\mathbf{a} \in \{P_1, ..., P_{24}\}$	$L.count := match(\mathbf{a})$
		$L.hit := match(\mathbf{a})$
(11)	$N \rightarrow \mathbf{n}$, for some $\mathbf{n} \in \{1, 2, \ldots\}$	$N.cutoff := \mathbf{n}$
(12)	$W \rightarrow \{. : r = N\}$	$W.hit := 1$
(13)	$O \rightarrow \{LC : \mathrm{p} \geq N\}$	$O.hit := LC.count \geq N.cutoff$
(14)	$LC \rightarrow (LC_1)(LC_2)$	$LC.count := LC_1.count + LC_2.count$
(15)	$LC \rightarrow L$	$LC.count := L.count$
	A	**B**

FIGURE 6. The grammar (*A*) and semantics (*B*) of the pattern language used by TargetBoost. The grammar and semantics are explained in the main text.

Unknown pattern	*Variable distance*	*Consecutive pattern*
Q_1: $\{\{P_1P_2(P_3\|P_3)P_4 : \mathrm{p} \geq 3\}$	$\{.:r = 8\}:d = 7$	$\}\{P_6P_7P_8P_9P_{10}P_{11} : \mathrm{p} \geq 2\}$
Q_2: $\{\ P_5P_{20}((P_3\|P_{24})\|P_{16})$	$\{.:r = 4\}:d = 10$	$\}\{P_{13}P_{14}P_{15}P_{16}P_{17} : \mathrm{p} \geq 5\}$

FIGURE 7. Two example patterns from our pattern language. P_i denotes nucleotide i in the miRNA counted from the 3′ end.

Figure 6B shows the language's semantics. A pattern matches a sequence if *S.hit* is true. *match*(**a**) returns 1 if the character in the position indicated by **a** is identical to the character it is compared with. *linger*(*F.hit, N*) is a function that if *F.hit* is true, *F.hit* will be returned for *N* clock-ticks (see Halaas et al. (2004) for details on the *linger*-function). The production for *W* creates a sequence of *N* wild cards. This production will return a hit for any sequence of *N* characters it is compared with.

Each individual generated by these production rules consists of two parts as follows: an unknown pattern *R*, and a consecutive sequence *O* of near perfect matches. The two parts are separated by a variable amount of nucleotides, decided by the displacement *D*. The number of wild cards in the *W*-production gives the lower bound of the number of nucleotides, and the number of wild cards, plus the displacement d in the *D*-production gives the upper bound of the number of nucleotides.

Figure 7 shows two example patterns from our pattern language. In the first query, the unknown pattern and the consecutive sequence are separated by 8–15 nt, and in the second query, by 4–14 nt. As in Sætrom (2004), we use the pattern n-of-m operator ($P \geq N$ in productions 4 and 13 in Fig. 6) to introduce fuzzy matching. That is, the numeral *N* in productions 4 and 13 indicates the minimum number of terminals in the *C* and *LC* productions that must match. For example, in Q_1, only two of six nucleotides must match, but in Q_2, all five nucleotides must match. This is also the case for the unknown pattern; the complete expression must match in Q_2, as it does not use the pattern n-of-m operator, but only three of four nucleotides must match in Q_1.

The terminals in the expressions represent positions in the miRNA-sequence; the expressions are therefore translated before searching. During translation, the terminals that represent positions are replaced with the corresponding complemented nucleotide in the miRNA sequence. The positions in the miRNA are numbered from $P_1$to P_{24}, with P_{24} corresponding to the 5' end of the miRNA. Our current implementation translates the miRNAs from 5' to 3', but only uses the 21 first nucleotides—P_1 to P_3 defaults to wild cards that match any nucleotide. TargetBoost evaluates a candidate pattern by using the translated queries to search the training set of positive and negative sequences. It then scores the pattern based on the number of true and false positive/negative hits and the relative weights the boosting algorithm has assigned to the sequences.

Reference algorithms for comparison

We compared the performance of TargetBoost with the performance of Nucleus (Rajewsky and Socci 2004) and RNAhybrid (Rehmsmeier et al. 2004) (these algorithms are described in the Introduction). Nucleus has two cut-off parameters that can be tuned—the weighted sum cut-off and the free energy cut-off—and when comparing the performance of this algorithm with the performance of our algorithm, we made certain modifications.

$Nucleus_1$ does not use mfold, and therefore, has only one cut-off parameter to tune. $Nucleus_2$ has a free-energy cut-off of −17.4, while the weighted sum cut-off is tunable. This was the cut-off recommended in Rajewsky and Socci (2004). $Nucleus_3$ has a weighted sum cut-off of 25, while the free-energy cut-off is tunable. Again, this cut-off was recommended in Rajewsky and Socci (2004).

We ran RNAhybrid in two modes; $RNAhybrid_1$ ran without forcing miRNA 5′ helices, and $RNAhybrid_2$ forced miRNA 5′ helices from position two to seven, as suggested by Rehmsmeier et al. (2004). Throughout this work, RNAhybrid and Nucleus are short for $Nucleus_1$ and $RNAhybrid_1$.

Positive data set

The positive data set consisted of 36 experimentally confirmed target sites for the miRNAs *let-7*, *lin-4*, *miR-13a*, and *bantam* in *C. elegans* and *D. melanogaster* (Boutla et al. 2003; Brennecke et al. 2003; Rajewsky and Socci 2004). Each target site was padded with their respective sequences, such that the length of the sequences was 30 nt. Target sites longer than 30 nt were discarded from the data set.

Negative data set

The negative data set consisted of 3000 random strings, all 30 nt long. The frequencies used in the generation of the random strings were the same as the frequencies used in Rajewsky and Socci (2004), ($P_A = 0.34$, $P_C = 0.19$, $P_G = 0.18$, $P_U = 0.29$), and correspond to the nucleotide composition of *D. melanogaster* 3′ UTRs.

Cross-validation

Cross-validation is a common method to evaluate the performance of a classifier on data not used to train the classifier. Here, we used 10-fold cross-validation (Breiman et al. 1984) and an approach we call "leave-one-miRNA-out" cross-validation. A 10-fold cross-validation usually gives a good estimate of a classifier's predictive accuracy (Kohavi 1995). In this case, however, the number of verified target sites for each miRNA varied greatly, so that the miRNA having the most target sites (*let-7*) had a high chance of being present in both the training and test sets in many of the 10-folds. As this may cause a bias in the classifier performance estimated by the 10-fold cross-validation method, we tried a second cross-validation approach that did not have this bias. In the "leave-one-miRNA-out" cross-validation approach, we used all of the target sites from all of the miRNAs, but one, as training set; we then used the remaining miRNA's target sites as test set. This gave four training and test sets.

Comparing algorithms

We compared the algorithms by analyzing their receiver operating characteristics (ROC) curves. A ROC-curve describes the relationship between the specificity $Sp = TN/(FP + TN)$ and the sensitivity $Se = TP/(TP + FN)$ of a classifier. Here, *TP*, *FP*, *TN*, and *TN* are the number of true positives, false positives, true negatives, and false negatives.

We did three analyses on the ROC-curves, i.e., area tests, TPF tests, and ROC_{50} tests. In the area tests, we calculate the area under the ROC-curve—the ROC-score. An area of 1 indicates a perfect classification, and an area of 0.5 indicates a random classification. In the TPF tests, we calculate the true-positive frequency ($TPF = S_e$) for a classifier for a given false-positive frequency ($FPF = 1 - S_p$), or the amount of correctly classified positive samples given a specified amount of false-positive samples. In the ROC_{50} tests, we calculate the ROC_{50} score, which is the area under the ROC-curve plotted until 50 true negative samples are found (Gribskov and Robinson 1996).

We used ROCKIT (Metz et al. 1998) for statistical comparisons of ROC area and TPF values.

Availability

TargetBoost is available as a Web tool from http://www.interagon.com/demo/. Currently, the Web tool searches the 3′ UTRs of *C. elegans*; other data sets are available for both commercial and strategic academic collaborations.

ACKNOWLEDGMENTS

We thank O.R. Birkeland for valuable comments on the manuscript and N. Rajewsky for sharing his data set of miRNA target sites. The work was supported by the Norwegian Research Council, grant 151899/150, and the bioinformatics platform at the Norwegian University of Science and Technology, Trondheim, Norway.

Received December 29, 2004; accepted April 7, 2005.

REFERENCES

Bartel, D.P. 2004. MicroRNAs: Genomics, biogenesis, mechanism, and function. *Cell* **116:** 281–297.

Boutla, A., Delidakis, C., and Tabler, M. 2003. Developmental defects by antisense-mediated inactivation of micro-RNAs 2 and 13 in *Drosophila* and the identification of putative target genes. *Nucleic Acids Res.* **31:** 4973–4980.

Breiman, L., Friedman, J.H., Olshen, R.A., and Stone, C.J. 1984. *Classification and regression trees.* Wadsworth, Belmont, CA.

Brennecke, J., Hipfner, D.R., Stark, A., Russell, R.B., and Cohen, S.M. 2003. *bantam* Encodes a developmentally regulated miRNA that controls cell proliferation and regulates the proapoptotic gene *hid* in *Drosophila*. *Cell* **113:** 25–36.

Doench, J.G. and Sharp, P.A. 2004. Specificity of microRNA target selection in translational repression. *Genes & Dev.* **18:** 504–511.

Doench, J., Petersen, C., and Sharp, P. 2003. siRNAs can function as miRNAs. *Genes & Dev.* **17:** 438–442.

Enright, A.J., John, B., Gaul, U., Tuschl, T., Sander, C., and Marks, D.S. 2003. MicroRNA targets in *Drosophila*. *Genome Biol.* **5:** R1.

Gribskov, M. and Robinson, N.L. 1996. The use of reciever operator characteristic (ROC) analysis to evaluate sequence matching. *Comput. Chem.* **20:** 25–34.

Griffiths-Jones, S. 2004. The microRNA registry. *Nucleic Acids Res.* **32:** D109–D111.

Halaas, A., Svingen, B., Nedland, M., Sætrom, P., Snøve Jr., O., and Birkeland, O.R. 2004. A recursive MISD architecture for pattern matching. *IEEE Trans. on VLSI Syst.* **12:** 727–734.

John, B., Enright, A.J., Aravin, A., Tuschl, T., Sander, C., and Marks, D.S. 2004. Human microRNA targets. *PLoS Biol.* **2:** e363.

Kiriakidou, M., Nelson, P.T., Kouranov, A., Fitziev, P., Bouyioukos, C., Mourelatos, Z., and Hatzigeorgiou, A. 2004. A combined computational-experimental approach predicts human microRNA targets. *Genes & Dev.* **18:** 1165–1178.

Kloosterman, W.P., Wienholds, E., Ketting, R.F., and Plasterk, R.H. 2004. Substrate requirements for *let-7* function in the developing zebrafish embryo. *Nucleic Acids Res.* **32:** 6284–6291.

Knuth, D.E. 1964. Backus normal form vs. Backus Naur form. *Commun. ACM* **7:** 735–736.

Kohavi, R. 1995. A study of cross-validation and bootstrap for accuracy estimation and model selection. In *Proceedings of the Fourtheen International Joint Conference on Artificial Intelligence*, pp. 1137–1143. Morgan Kaufmann Publishers, Montreal Canada.

Koza, J.R. 1992. *Genetic programming: On the programming of computers by natural selection.* MIT Press, Cambridge, MA.

Lagos-Quintana, M., Rauhut, R., Lendeckel, W., and Tuschl, T. 2001. Identification of novel genes coding for small expressed RNAs. *Science* **294:** 853–858.

Lai, E.C. 2004. Predicting and validating microRNA targets. *Genome Biol.* **5:** 115.

Lai, E.C., Tomancak, P., Williams, R.W., and Rubin, G.M. 2003. Computational identification of *Drosophila* microRNA genes. *Genome Biol.* **4:** R42.

Lau, N.C., Lim, L.P., Weinstein, E.G., and Bartel, D.P. 2001. An abundant class of tiny RNAs with probable regulatory roles in *Caenorhabditis elegans*. *Science* **294:** 858–862.

Lee, R.C. and Ambros, V. 2001. An extensive class of small RNAs in *Caenorhabditis elegans*. *Science* **294:** 862–864.

Lee, R.C., Feinbaum, R., and Ambros, V. 1993. The *C. elegans* heterochronic gene *lin-4* encodes small RNAs with antisense complementarity to *lin-14*. *Cell* **75:** 843–854.

Lewis, B.P., hung Shih, I., Jones-Rhoades, M.W., Bartel, D.P., and Burge, C.B. 2003. Prediction of mammalian microRNA targets. *Cell* **115:** 787–798.

Lim, L.P., Glasner, M.E., Yekta, S., Burge, C.B., and Bartel, D.P. 2003a. Vertebrate microRNA genes. *Science* **299:** 1540.

Lim, L.P., Lau, N.C., Weinstein, E.G., Abdelhakim, A., Yekta, S., Rhoades, M.W., Burge, C.B., and Bartel, D.P. 2003b. The microRNAs of *Caenorhabditis elegans*. *Genes & Dev.* **17:** 991–1008.

Meir, R. and Rätsch, G. 2003. An introduction to boosting and leveraging. In *Advanced lectures on machine learning* (eds. S. Mendelson and A. Smola), Vol. 2600, pp. 118–183. Springer-Verlag, GmbH.

Metz, C.E., Herman, B.A., and Roe, C.A. 1998. Statistical comparison of two ROC-curve estimates obtained from partially-paired datasets. *Med. Decis. Making* **18:** 110–121.

Moss, E.G., Lee, R.C., and Ambros, V. 1997. The cold shock domain protein LIN-28 controls developmental timing in *C. elegans* and is regulated by the *lin-4* RNA. *Cell* **88:** 637–646.

Olsen, P.H. and Ambros, V. 1999. The *lin-4* regulatory RNA controls developmental timing in *Caenorhabditis elegans* by blocking LIN-14 protein synthesis after the initiation of translation. *Dev. Biol.* **216:** 671–680.

Pasquinelli, A.E., Reinhart, B.J., Slack, F., Martindale, M.Q., Kuroda, M.I., Maller, B., Hayward, D.C., Ball, E.W., Degnan, B., Müller, P., et al. 2000. Conservation of the sequence and temporal expression of *let-7* heterochronic regulatory RNA. *Nature* **408:** 86–89.

Rajewsky, N. and Socci, N.D. 2004. Computational identification of microRNA targets. *Dev. Biol.* **267:** 529–535.

Rehmsmeier, M., Steffen, P., Höchsmann, M., and Giegerich, R. 2004. Fast and effective prediction of microRNA/target duplexes. *RNA* **10:** 1507–1517.

Reinhart, B., Slack, F., Basson, M., Pasquinelli, A., Bettinger, J., Rougvie, A., Horvitz, H., and Ruvkun, G. 2000. The 21-nucleotide *let-7* RNA regulates developmental timing in *Caenorhabditis elegans*. *Nature* **403:** 901–906.

Sætrom, P. 2004. Predicting the efficacy of short oligonucleotides in antisense and RNAi experiments with boosted genetic programming. *Bioinformatics* **20:** 3055–3063.

Sætrom, P. and Snøve Jr., O. 2004. A comparison of siRNA efficacy predictors. *Biochem. Biophys. Res. Commun.* **321:** 247–253.

Saxena, S., Jonsson, Z., and Dutta, A. 2003. Implications for off-target acitivity of small inhibitory RNA in mammalian cells. *J. Biol. Chem.* **278:** 44312–44319.

Scacheri, P.C., Rozenblatt-Rosen, O., Caplen, N.J., Wolfsberg, T.G., Umayam, L., Lee, J.C., Hughes, C.M., Selvi Shanmugam, K., Bhattacharjee, A., Meyerson, M., et al. 2004. Short interfering RNAs can induce unexpected and divergent changes in the levels of untargeted proteins in mammalian cells. *Proc. Natl. Acad. Sci.* **101:** 1892–1897.

Smalheiser, N.R. and Torvik, V.I. 2004. A population-based statistical approach identifies parameters characteristic of human microRNA-mRNA interactions. *BMC Bioinformatics* **5:** 139.

Stark, A., Brennecke, J., Russell, R.B., and Cohen, S.M. 2003. Identification of *Drosophila* microRNA targets. *PLoS Biol.* **1:** E60.

Wightman, B., Ha, I., and Ruvkun, G. 1993. Posttranscriptional regulation of the heterochronic gene *lin-14* by *lin-4* mediates temporal pattern formation in *C. elegans*. *Cell* **75:** 855–862.

Yekta, S., Shih, I., and Bartel, D.P. 2004. MicroRNA-directed cleavage of *HOXB8* mRNA. *Science* **304:** 594–596.

Zeng, Y., Wagner, E., and Cullen, B. 2002. Both natural and designed micro RNAs can inhibit the expression of cognate mRNA when expressed in human cells. *Mol. Cell.* **9:** 1327–1333.

Zuker, M. 2003. Mfold Web server for nucleic acid folding and hybridization prediction. *Nucleic Acids Res.* **31:** 3406–3415.