

KATHOLIEKE UNIVERSITEIT
LEUVEN

Arenberg Doctoral School of Science, Engineering & Technology
Faculty of Engineering
Department of Electrical Engineering

COMPUTATIONAL DISCOVERY OF *CIS*-REGULATORY MODULES BASED ON ITEMSET MINING

Hong SUN

Dissertation presented in partial
fulfillment of the requirements for
the degree of Doctor
in Electrical Engineering

July 2011

COMPUTATIONAL DISCOVERY OF *CIS*-REGULATORY MODULES BASED ON ITEMSET MINING

Hong SUN

Jury:
Prof. dr. ir. Ann Haegemans, chair
Prof. dr. ir. Bart De Moor, promotor
Prof. dr. ir. Kathleen Marchal, co-promotor
Prof. dr. ir. Jos Vanderleyden
Prof. dr. ir. Yves Moreau
Prof. dr. ir. Annemieke Verstuyf
Dr. ir. Tim Van den Bulcke (Universitair Ziekenhuis Antwerpen & Universiteit Antwerpen, Belgium)

Dissertation presented in partial fulfillment of the requirements for the degree of Doctor in Electrical Engineering

July 2011

© Katholieke Universiteit Leuven – Faculty of Engineering
Address Kasteelpark Arenberg 10, B-3001 Leuven (Belgium)

Alle rechten voorbehouden. Niets uit deze uitgave mag worden vermenigvuldigd en/of openbaar gemaakt worden door middel van druk, fotocopie, microfilm, elektronisch of op welke andere wijze ook zonder voorafgaande schriftelijke toestemming van de uitgever.

All rights reserved. No part of the publication may be reproduced in any form by print, photoprint, microfilm or any other means without written permission from the publisher.

D/2011/7515/92
ISBN 978-94-6018-390-4

Acknowledgements

When I chose my Master's thesis, I wanted to go in the direction of Bioinformatics, because I was interested in it and it matches my background (I have a Bachelor degree in Electrical Engineering). Since one of the Master proposals supervised by Prof. Kathleen Marchal looked very interesting to me, I quickly went to her office. Unfortunately her office was locked that afternoon, so I just waited for her there. Luckily she came back after one hour together with Sigrid and she gave a quick introduction to the topic on motif detection since she had to catch the train, so we finished our talk. On that Saturday, she sent me some paper relating to that topic. The tutor for this master thesis was Marleen Claeys, but at that time she was in Qatar, so Kathleen directly supervised me. We met once every week and I guess she must be terribly tortured by my English. One day she told me, she might need a PhD student on the topic of motif detection, but she couldn't promise me this position. She said: "if you are interested, you first have to get good scores for your exams". I was extremely surprised at that moment, it was beyond my imagination that Kathleen would give such an opportunity to me. My English was not that OK, moreover my knowledge of the field was very limited. Although I was not sure whether I could make it, nevertheless I felt extremely encouraged and motivated, I was telling myself absolutely I shouldn't disappoint her, at least I should try my very best. Unfortunately I didn't get a distinction in the exams. But Kathleen still would like to enroll me as her PhD student and I am really grateful for that. Unfortunately there was no funding available at that time, and Kathleen recommend me to several professors. Finally Kathleen talked to Bart, and Bart was glad to let me first start a pre-doctoral program. Thank you Bart for giving such a great opportunity to me!

At the beginning, I was living my life relying on my poor English, more seriously I didn't know much about biology. Kathleen was very patient with me and taught me everything that I didn't know. Her attitude towards work as well as her charming character influenced me a lot and I realized I first had to correct my attitude. I could learn from the beginning, might be slow but must be certain.

I am very grateful for my colleagues who gave me many chances and trained me

actively as a scientist with their knowledge, comments and discussions. Thank you Karen, during my PhD you helped me a lot. I felt depressed when you were leaving us, and in fact I didn't intend to let you eat the extremely salty beancurd all at once when we were at the conference in Shanghai. Tim and Thomas, I am very gratefully for the scientific discussions and help when I was sitting at ESAT, and helping me with a lot of practical matters. I want to thank Tias Guns, Dr. Siegfried Nijssen and Prof. Luc De Raedt for the interesting, and instructive discussions on constraint programming for itemset mining and also the excellent collaboration we have. I also should express a word of thanks to Prof. Tijl De Bie, thanks for the great on DISTILLER and ModuleDigger. Furthermore, I would like to thank Dr. Kristof Engelen, Dr. Pieter Monsieurs, Dr. Carolina Fierro, Dr. Inge Thijs, Dr. Abeer Fadda, Dr. Hui Zhao, Dr. Riet De Smet, Dr. Valerie Storms, Marleen Claeys, Aminael Sanchez Rodriguez, Ivan Ischukov, Peyman Zarrineh, Pieter Meysman, Lore Cloots, Lyn Venken, Yan Wu, Dries De Maeyer, Dr. Sigrid De Keersmaecker, Prof. Kevin Verstrepen and Prof. Jozef Vanderleyden for the pleasant and interesting collaboration within or outside my PhD project. I am very grateful for the great environment we always had for New Year party at Kathleen's place, brainstorming sessions, BBQ events, seashore walking, zoo events, paintball events, skiing holidays, game evenings and many other activities which always invigorated me with renewed energy to work. In addition, I want to thank little Mira. What wonderful times we always have! You are such a cute and cool kid, we have lots of happiness together!

I would also like to thank the chair Prof. Ann Haegemans and members of the jury: Prof. Bart De Moor, Prof. Kathleen Marchal, Prof. Jozef Vanderleyden, Prof. Yves Moreau, Prof. Annemieke Verstuyf and Dr. Tim Van den Bulcke for providing me with valuable comments and suggestions that improved this PhD text.

I finally want to thank my family and all my friends for all their aid and support during my Master and PhD period. Especially to my friends (alphabetical order of family name): Jiaci Cai, Yuanyuan Cao, Beiwen Chen, Wei Dai, Jiyin He, Ping Hou, Hao Hu, WeiDa Hu, Xiaoyan Huang, Yingli Kan, Tong Li, Zhiqiang Ma, Lele Qin, Jianxiong Sheng, Jiabin Song, Ding Sun, Dandan Wang, Wei Wu, Yanfei Wu, Xiaoli Wu, Zhaojun Yu, Qiyun Zhang and Yuan Zhao who are always ready for me. I am so lucky to have all of you! And of course *grandpa*, *grandma*, mum, dad and brother, thanks for your trusts on all of what I did, and for all the things you taught me in my whole life. Thanks to Ilse, Ida, John, Anita, Elsy and Mimi for the nice administrative work and practical arrangements. At finally, my greatest thanks again to my promoters, who made tremendous efforts to turn me into a scientist!

Sunny, July 2011

Abstract

The main topic of this PhD is the development of computational tools for the detection of *cis*-regulatory module (CRMs) using *itemset mining* techniques.

A first method ModuleDigger, is a CRM detection method to detect *cis*-regulatory modules based on set of coregulated sequences, relying on CHARM to enumerate possible motif combinations and well-designed statistical scoring scheme to prioritize biologically valid CRMs. We benchmarked ModuleDigger with existing tools and tested its validity on a real dataset. However, as ModuleDigger doesn't take into account the proximity of binding sites composed a certain CRM, it still oversimplifies the biological problem. Although it performs well in detecting the true regulatory modules it can not specify the true binding sites that compose the modules.

Therefore we developed CPModule, a CRM detection method that relies on a *constraint programming* framework for *itemset mining*. CPModule enumerates all possible CRMs that meet the following biologically motivated constraints: a certain CRM should occur in a minimal number of sequences (frequency constraint) and its composing motif sites should occur within a maximal genomic distance from each other (proximity constraint). The first constraint allows tuning the degree of overrepresentation that we expect in a set of intergenic sequences, while the second constraint reflects that sites of combinatorially acting TFs occur in each others neighbourhood. A last constraint (redundancy constraint) reduces the level of redundancy amongst the valid CRMs. Firstly, we experimentally validate our approach and compare it with state-of-art techniques using a literature existing synthetic data. Secondly, we propose CRM detection in combination with ChIP-Seq by performing a real case study on ChIP-Seq experiments of five transcription factor KLF4, NANOG, OCT4, SOX2 and STAT3 on mouse embryonic stem cell. Epigenetic information is also used to check whether surrounding chromatin stability for TFBSs is permissive for the binding of TFs.

Besides for detecting CRMs, we also developed ViTraM, a tool for visualizing expression module i.e. gene sets that are coexpressed under a specific set of conditions with or without their regulatory program (sets of transcription factors

that are responsible for the observed coregulation). It uses a input the result of biclustering or network inference algorithms. ViTraM is capable of visualizing overlapping these transcriptional/expression modules in an intuitive way by allowing for a dynamic visualization and using multiple methods for obtaining the optimal layout. In addition to visualizing multiple modules, ViTraM also allows to display additional information on the regulatory program of the modules, which consists of the transcription factors and their corresponding motifs. Information on the regulatory program is either obtained from curated databases or from the outcome of the inference tool itself. By visualizing not only the modules but also the regulatory program, ViTraM can provide more insight into the modules and facilitates the biological interpretation of the identified modules.

Korte inhoud

Het doel in deze doctoraatsthesis is het ontwikkelen van computationele tools voor de detectie van *cis*-acting-regulatory modules (CRMs) gebruik makend van *itemset mining*.

Een initieel ontwikkelde methode is ModuleDigger: een methode die op basis van een set van co-gereguleerde sequenties, CRMs detecteert. ModuleDigger combineert de computationele efficientie van CHARM met een goed ontworpen statistisch scoringsschema dat toe laat de statistisch meest relevante modules te prioritizeren. ModuleDigger werd vergeleken met bestaande state-of-the-art tools en de biologische relevantie van de tool werd aangetoond a.h.v. een echte dataset. Omdat ModuleDigger geen rekening houdt met het aantal bindingsites van een transcriptiefactor en hun relatieve positionering op het genoom, oversimplifieert ModuleDigger het CRM detectie probleem. Hoewel ModuleDigger dus perfect in staat is de juiste CRM te detecteren, is het niet mogelijk om ook af te leiden welke specifieke binding sites bijdroegen tot de CRM.

Daarom werd CPModule ontwikkeld, een CRM methode gebaseerd op *constraint programming* voor *itemset mining*. CPModule somt alle mogelijke CRMs op die voldoen aan de volgende biologische gemotiveerde beperkingen: een bepaalde CRM moet in een minimaal aantal sequenties voorkomen (frequentiebeperking) en zijn motiefplaatsen moeten zich binnen een maximale genomische afstand van elkaar bevinden (afstandsbeperking). De eerste beperking laat toe de graad van overrepresentatie, die we verwachten in een set van intergenische sequenties, te regelen, terwijl de tweede beperking weerspiegelt dat de bindingsplaatsen van TFs, die voor combinatoriele regulatie zorgen, voorkomen in elkaars buurt. Een laatste beperking (redundantiebeperking) reduceert de hoeveelheid redundantie tussen geldige CRMs. Eerst valideren we onze aanpak experimenteel en vergelijken we deze met state-of-the- art technieken, gebruik makende van synthetische data uit de literatuur. Ten tweede stellen we CRM detectie voor in combinatie met ChIP-Seq door een echte case study uit te voeren op ChIP-Seq experimenten van vijf transcriptiefactoren, zijnde KLF4, NANOG, OCT4, SOX2 en STAT3, op muis embryonische stamcellen. Epigenetische informatie werd eveneens gebruikt om na

te gaan of omliggende chromatine stabiliteit voor transcriptiefactor bindingsites het binden van transcriptiefactoren toelaat.

Behalve voor het detecteren van CRMs, werd in deze thesis ook ViTraM ontwikkeld, een methode voor het visualiseren van expressie modules i.e., gen set die coexpressed is onder een subset van de condities in een expressie compendium al of niet in combinatie met hun regulatorisch programma (set van transcriptiefactoren verantwoordelijk voor het waargenomen coexpressie gedrag). ViTraM gebruikt als input het resultaat van een biclustering of netwerk inferentie programma. ViTraM maakt het mogelijk om op een intuitieve manier overlappende transcriptionele modules te visualiseren door gebruik te maken van een dynamische visualisatie en meerdere methodes aan te bieden om een optimale layout voor de overlappende modules te bekomen. Naast enkel het visualiseren van meerdere modules, laat ViTraM ook toe additionele informatie over het regulatieprogramma van de modules te tonen. Het regulatieprogramma bestaat uit de transcriptiefactoren en hun overeenkomstige motieven. Informatie over het regulatieprogramma kan verkregen worden uit gecureerde databanken of uit het resultaat van de module inferentiemethode zelf. Beide types van informatie over het regulatieprogramma kunnen toegevoegd worden door ViTraM. Door niet enkel de modules maar ook hun regulatieprogramma te visualiseren, kan ViTraM biologische interpretatie van de modules vergemakkelijken.

Abbreviations and terminology

Abbreviations

ARM	Association rule mining
BP	Base pair
ChIP	Chromatin immunoprecipitation
ChIP-chip	Chromatin immunoprecipitation (ChIP) on a microarray (chip)
ChIP-Seq	Chromatin immunoprecipitation (ChIP) and sequencing
CP	Constraint programming
CRM	*Cis*-regulatory module
CPModule	*Cis*-regulatory module detection using constraint programming
DISTILLER	Data Integration System To Identify Links in Expression Regulation
DNA	Deoxyribonucleic acid
FN	False negative
FP	False positive
GEO	Gene Expression Omnibus
GO	Gene ontology
HMM	Hidden Markov model
IUPAC	International Union of Pure and Applied Chemistry
KB	Kilo base
mRNA	Messenger RNA
ModuleDigger	*Cis*-regulatory module detection framework based on
NCBI	National Centre for Biotechnology Information
NT	Nucleotide(s)
PSSM	Position specific scoring matrix
PWM	Position weight matrix
RNA	Ribonucleic acid
TF	Transcription factor
TFBS	Transcription factor binding site
TN	True negative
TP	True positive
TSS	Transcription start site
ViTraM	Visualize transcriptional module network

Terminology

Closed itemset	Frequent itemset that cannot be extended with an additional item without changing the support.
Frequent itemset	Itemset of which the support exceeds the support threshold
Item	A basic element in association rule mining algorithms. Items are grouped together to form itemsets.
Itemset	A group of items.
Maximal itemset	Frequent itemset that will not meet the support threshold anymore upon addition of an extra item.
Motif	The representation of a set of binding sites.
Motif instance	A binding site in the promoter region of a gene.
Regulatory program	Regulators and/or motifs.
Support	The number of transactions in which the items of an itemset appear together.
Support threshold	The minimum support.
Transaction	A property shared by a group of items.

Contents

Acknowledgements i

Abstract iii

Korte inhoud v

Abbreviations and terminology viii

Contents xiii

1 Introduction 1

1.1 Systems biology 1

1.2 Transcriptional regulation in eukaryotes 2

1.3 Regulatory motif 3

1.4 Motif representation 3

1.5 *Cis*-regulatory module 5

1.6 Traditional *cis*-regulatory module screening methods 6

1.7 Limitations of current CRM screening methods 9

1.7.1 Limitations of single transcription factor screening 9

1.7.2 Limitations of combinatorial search 9

1.8 Possible epigenetic features for CRM screening 10

1.8.1 Nucleosome occupancy feature 10

1.8.2 Histone modification features 11
1.8.3 DNA methylation feature and CpG islands 11
1.9 Achievements 12
1.9.1 Part I: ModuleDigger 12
1.9.2 Part II: CPModule 13
1.9.3 Part III: ViTraM 13
1.9.4 Summary 14

2 Association rules mining algorithms 15

2.1 ARM algorithms 15
2.2 CHARM algorithm 16
2.3 Application of ARM algorithms on Bioinformatics 18

3 ModuleDigger: *Cis*-regulatory module detection based on itemset mining 22

3.1 Introduction 22
3.2 Module detection framework 23
3.2.1 Enumerating all frequent closed CRMs 25
3.2.2 Assigning a rank to each CRM 25
3.2.3 Module specificity score 25
3.2.4 Iterative p-value updating of the module specificity score . 26
3.3 Results 28
3.3.1 Dataset 28
3.3.2 Running parameters for ModuleDigger 29
3.3.3 Benchmarking ModuleDigger 30
3.3.4 Running parameters for other tools 32
3.3.5 Comparing with other tools 34
3.4 Conclusion 34

4 CPModule: Unveiling combinatorial regulation in mouse embryonic stem cell 35

4.1 Introduction . . . 35

4.2 Results . . . 37

4.2.1 Analysis flow . . . 37

4.2.2 Motif screening with filtering based on epigenetic signals . . . 39

4.2.3 CPModule: CRM detection based on constraint programming for itemset mining . . . 42

4.2.4 Detecting CRMs involved in mouse embryonic stem cell . . . 45

4.2.5 Effect of the screening procedures on the final modules . . . 50

4.2.6 Literature supports for detected CRMs/TFs in mouse embryonic stem cell (ESC) biology . . . 52

4.3 Conclusion . . . 56

4.4 Materials and Methods . . . 57

4.4.1 Datasets . . . 57

4.4.2 Step 1: Motif screening based on epigenetic features . . . 57

4.4.3 Step 2: CRM detection based on constrained-based itemset mining . . . 58

4.4.4 Motif level correlation coefficient (mCC) and nucleotide level correlation coefficient (nCC) . . . 61

4.4.5 Running parameters . . . 62

5 ViTraM: VIsualize TRAnscriptional gene Module network 64

5.1 Motivation . . . 64

5.2 Introduction . . . 65

5.3 XMLCreator . . . 66

5.3.1 Requirements for installation of XMLCreator . . . 67

5.3.2 Installation of the XMLCreator . . . 67

5.3.3 Data for XMLCreator . . . 67

5.4 ViTraM . . . 70

5.4.1 Inputs for ViTraM 71
5.4.2 Structure of ViTraM 72
5.4.3 Layout algorithm 73
5.4.4 Dynamic visualization 74
5.4.5 Export images 77
5.4.6 Overview on some of the functionalities of ViTraM 77
5.5 Case study 77
5.5.1 General introduction of DISTILLER 77
5.5.2 Visualizing gene regulatory network constructed by DISTILLER 78
5.6 Conclusion 79

6 Conclusions and Perspectives **84**

6.1 Conclusions 84
6.2 Perspectives 85

Bibliography **87**

List of Publications **103**

Chapter 1

Introduction

1.1 Systems biology

Systems biology is a term used to describe a number of trends in bioscience research, and a movement which draws on those trends. Proponents describe systems biology as a biology based inter-disciplinary study field that focuses on complex interactions in biological systems, claiming that it uses a new perspective (holism instead of reduction). Particularly from year 2000 onwards, the term is used widely in the biosciences, and in a variety of contexts. An often stated ambition of systems biology is the modeling and discovery of emergent properties, properties of a system whose theoretical description is only possible using techniques which fall under the field of systems biology.

The diverse physiological and phenotypic changes that a cell undergoes in its lifetime are governed by gene expression. At the initial step of gene expression, transcription is shaped mainly by the interaction between the RNA polymerase, the transcription factors (TFs) and the promoter sequence of a gene. Although transcription is not the sole determinant of gene expression, it is the bottleneck in this complex pathway. Hence, a full understanding of the interplay between TFs and their target sequences would provide the means to interpret and model the responses of the cell to diverse stimuli. And therefore, the reconstruction of the transcriptional network becomes a vital objective.

Traditional molecular biology methods for resolving the transcriptional regulatory program have relied on the analysis of single genes. These methods, although fairly reliable, are tedious and slow. The need for an efficient 'line of production' of information had led to the 'omics' era. Advances in experimental procedures allowed for the study of hundreds of genes and proteins simultaneously. Terms such

as proteomics, transcriptomics, metabolomics, *etc*, became commonplace. With the flood of information created by the new techniques, came the need for an informatics approach to the problem, also known as *in silico* analysis, which is the topic of this thesis.

1.2 Transcriptional regulation in eukaryotes

Transcription is the process during which genetic information is transcribed from DNA to RNA. The "expression" of a gene designates the level of messenger RNA (mRNA) present in the cell transcribed from that gene. For most protein coding genes the level of expression varies along with the circumstances, i.e. developmental stage, cell types, nutrient level, *etc*. The expression level of each individual gene is mostly controlled at the level of transcription (Wray et al., 2003). Transcription regulation is a highly dynamic process that involves a combination of factors: the general transcription initiation factors that make up the basal transcription apparatus, sequence-specific DNA binding factors that bind to up or downstream regulatory elements and associated accessory factors. Eukaryotic protein coding genes are transcribed by the RNA polymerase II (RNAPII) holoenzyme complex (Lee & Young, 2000). This complex consists of RNAPII and a set of basal transcription factors (TFs), namely TFII A, B, C, D, E, F and H.

Assembly of the RNAPII holoenzyme complex on the basal promoter initiates transcription. Although basal promoter sequences differ among genes, for many genes the critical binding site is the TATA box, usually located circa 25-30 bp upstream of the transcription start site (TSS). In such promoters, the attachment of the TATA-binding protein (TBP, also known as TFIID) to the TATA box is a crucial step in transcription initiation. Some genes, however, contain an initiator element instead of the TATA box or neither of both. In these cases, TBP binds to the DNA in a sequence independent manner, protein that bind to other motifs in the basal promoter facilitate this. Once TBP attaches to the DNA, several TBP-associated factors (TAFs) guide the RNAPII holoenzyme complex to DNA. Transcription factors binding at other sites can modulate this attachment in positive or negative way (Lee & Young, 2000; Lemon & Tjian, 2000). After the RNAIIP holoenzyme complex assembles onto the DNA a second contact is established at the TSS (transcription start site). By itself a basal promoter initiates transcription at a very low rate. Moreover, the transcription initiation factors binding to the basal promoter and assisting the initiation of transcription are omnipresent, providing little regulatory specificity. Producing functionally significant levels of mRNA requires the sequence specific binding of transcription factors (TFs) to DNA motifs, i.e. transcription factor binding sites (TFBSs), outside the basal promoter (Lemon & Tjian, 2000).

1.3 Regulatory motif

Transcription factor binding sites (TFBSs) or regulatory motifs are stretches of DNA that are recognized sequence specially by a transcription factor (TF) that is required to control the expression of the target gene. This TF can be an activator, enhancing the transcription of the target gene, or a repressor doing the opposite. Regulatory motifs specify and anchor the TFs in appropriate positions with respect to one another and to the basal transcription apparatus, these TFs, and other proteins that in turn bind to them, determine the rate of transcription and mediate the accurate activation and repression of the gene in developmental time and morphological space (Arnone & Davidson 1997).

Most regulatory motifs are 5 to 8 nucleotide (nt) long. Their presence is most often associated with the promoter region of the gene (i.e. the intergenic region located immediately upstream of the start of the gene). Recently it has been shown that they also occur at long distances upstream from the gene they target, furthermore, regulatory motifs sometimes occur in the un-translated region, the introns downstream (3') of the transcription unit and, rarely, within a coding exon, this diversity of positions is possible because DNA looping allows interaction between proteins associated with DNA and distant binding sites.

Known TFBSs are made publicly available through databases. Examples of such database are TRANSFAC (Matys et al., 2006), JASPAR (Bryne el al., 2008), REDFLY (Halfon et al., 2008), RegulonDB (Gama-Castro et al., 2008) and plantCARE (Lescot et al., 2002). Little is know about the amount of regulatory motifs present in mammalian genomes, but the number of such motifs is expected to be an order of magnitude higher than the number of protein coding genes, i.e. in the order of hundreds of thousands or more. The widely used TRANSFAC database contain 584 models for vertebrates TFBSs, this shows that our current knowledge of these DNA binding sites is severely limited. Although many methods have been developed to identify regulatory motifs, much more research is needed.

1.4 Motif representation

A review on motif representation is published by (Stormo, 2000). Four main ways are mostly used: Consensus Sequence (CS), Position Frequency Matrix (PFM), Position Weight Matrix (PWM) (or Position Specific Scoring Matrix (PSSM)) and Motif Logo (ML).

Consensus sequence: Each position is shown as one letter representing the most dominant base in that position. For example, the -10 region of the promoter would be represented by the consensus sequence TATAAT. However, it is very rare that this exact sequence is found in promoter regions. A better representation

would account for the mismatches or degeneracy of the motif. Thus, the IUPAC (International Union of Pure and Applied Chemistry) nucleic acid codes were employed in which two or more bases occurring at similar frequencies at the same position would be represented by a single letter. Using the same previous example, the -10 promoter region would be represented as TATRNT, allowing for an arginine or a guanine to be present at the 4th position. As much as this representation is an improvement to the 4-letter representation, it is still arbitrary and depends much on convention; for example, a single base is shown if it occurs in $> 50\%$ of the sites in some research articles, and in $> 60\%$ in others. Yet, this representation is still valid for motif detection tools depending on word enumeration as will be discussed later.

The significance of a particular site can be scored given the distribution of all occurrences of the consensus sequence using standard statistical procedures (e.g. Tompa, 1999).

Position Frequency Matrix (PFW): In this representation, the frequencies of each of the four DNA bases in the known sites for each of the positions is shown in a matrix. PFMs are more exact representations of the motif and allow for the use of probabilistic methods to search for new sites. However, it assumes a random distribution of the four bases in the genome, which is not the case as genomes are mostly biased in their GC content.

Position Weight Matrix (PWM) or Position Specific Scoring Matrix (PSSM): This is a matrix representation of the expected self-information of a particular base in a particular position

$$-f_{b,i} \log f_{b,i} \tag{1.1}$$

where $f_{b,i}$, i is the frequency of base b at position i. Pseudocounts have to be added to the frequencies to compensate for the limited observed data and the zero occurrences in the frequency matrix. When the distribution of single bases in the genome are taken into account, the formula becomes as follows

$$-f'_{b,i} \log_2 \frac{f'_{b,i}}{p_b} \tag{1.2}$$

Where $-f'_{b,i}$ is the frequency of base b at position i with psuedocounts added and p_b is the frequency of base b in the whole genome. Thus, a position's significance (weight) can be measured with this equation

$$I_{seq}(i) = \sum_b f'_{b,i} \log_2 \frac{f'_{b,i}}{p_b} \tag{1.3}$$

Which is also a measure of the relative entropy (Kullback-Liebler distance) of the binding site with respect to the background frequencies, and is also equivalent

to the log likelihood ratio. A PWM score of a complete motif is the sum of the log-likelihood scores of all its positions, and thus, it assumes independence between positions of a motif. A PWM is used to search for novel sites with a threshold typically based on the scores of the known sites.

Motif logo: This is graphical representation of the motif, where each position is represented by stacks of base letters, the height of which is scaled to the information content (IC) of the base frequency at that position, following this formula

$$I_i = 2 + \sum_b f_{b,i} \log_2 f_{b,i} \tag{1.4}$$

where I_i is the information content at position i, $f_{b,i}$, i is the frequency of base b at position i. IC indicates how well the base is conserved at each position, and takes a value between 0-2 bits, such that perfectly conserved positions contain 2 bits of information while bases that occur $> 50\%$ of the time contain one bit.

Limitations of the mentioned representations: Two main issues arise with respect to the use of the above motif representations to search for novel sites:

- Dependence on the number of known sites. The more sites the model is built on, the greater is its accuracy in predicting new sites. This is a major limitation that greatly biases the discovery of new sites, and cannot be overcome except with the laborious biological experiments.
- Interdependencies of bases within the motif are not accounted for. The significance of this is arguable. While some studies emphasize that interdependencies exist in at least some motifs (Bulyk, Johnson & Church, 2002; O'Flanagan, Paillard, Lavery & Sengupta, 2005), other studies show that accounting for those did not significantly improve the search results (Benos, Bulyk & Stormo, 2002). Several models were suggested to represent interdependencies, e.g. pairwise dependencies (Zhou & Liu, 2004) and Bayesian networks (Barash, Elidan, Friedman & Kaplan, 2003). As complex models maybe better representations of the reality, they come at a cost of needing more data to estimate the parameters, and running the risk of overfitting.

1.5 *Cis*-regulatory module

Transcription factors (TFs) are proteins either active or repress of genes by binding to short *cis*-regulatory elements called transcription factor binding sites (TFBSs) that lie in the vicinity of the target genes. TFBSs are often organized into clusters called *cis*-regulatory modules (CRMs), which typically span a few hundred nucleotides and contain several binding sites for about 2-10 transcription factors (TFs). CRM screening is a very important and difficult problem in computational

biology, with the availability of more and more biological information, the methods for CRM screening also experienced evolution. As to which method should we chose, we'd better first have an overview of the available methods, and also utilze what we have in hand to the upmost extent. In this review, we will first discuss the sequence based methods for CRM screening and then discuss some other features which can be or already be integrated into CRM screening methods to improve the prediction; lastly we summarize the available methods for assessing the performance of CRM screening methods.

In complex multicellular organisms, transcription factors (TFs) generally do not work in isolation, but together with other TFs, refer as *cis*-regulatory modules (CRMs). TFs that bind to DNA on these transcription factor binding sites (TFBSs) usually locate at the upstream of the transcription start site (TSS) of a gene. The presence of a CRM thus determines the transcriptional response of a specific gene. Coexpression might imply a similar mechanism of co-regulation, thus co-expressed genes can be searched for the presence of statistically over-represented CRMs. One challenge in molecular biology is to capture the CRMs. Thanks to the high throughput sequencing technologies, e.g. ChIP-chip experiments which allows for genomewide TFs screening. Nevertheless, ChIP-Seq experiment can only measure the binding specificity for single TF, and due to the limited availability of antibodies for certain TFs as well as the high expense for ChIP experiments, prediction of combination of TFBSs or CRMs still relies on CRM screening methods. The prediction of such CRM is very difficult while computational methods provide great hope, indeed computational biologists devoted considerable efforts to solve this problem in the past decade.

1.6 Traditional *cis*-regulatory module screening methods

If little knowledge is known about the TFs and their binding sites, such as in some understudied species, one is limited to the information contained within the DNA sequence. Methods have been developed which only use set of co-expressed or co-regulated sequences as input, referred to as *de novo* CRM screening methods (Zhou et al., 2004, Xie et al., 2008). Due to the computational limit, the set of sequences are required to contain fewer sequences (less than a hundred) and the length of the sequences should be shorter (only several hundreds nucleotides). In this thesis, we will not discuss *de novo* methods. With more and more TFs being studied and stored in public databases (Matys et al., 2006; Sandelin et al., 2004), some methods appeared, not only using sequences but also using already know motif models. Biologists want to know if the set of sequences are regulated by these already known TFs.

Table 1.1: Popular CRM Detection Methods

Algorithm	Year	Input	Parameters	Principle	Availability	Validation Data
Cister	2001	(1)DNA sequences	(1)Binding site detection threshold (2)Average distance between transcription binding sites (3)Average number of transcription factor binding sites (4)Average distance between transcription binding sites (5)Window size for local nucleotide frequency calculation (6)Pseudocount	HMM	Online	LSF (human) Muscle data
Ahab	2002	(1)DNA sequences (2)PWMs	(1)Window size (2)Window step size	Statistics	Request	Two synthetic data *Drosphila* embryo data *Drosphila* segmentation
Cluster-Buster	2003	(1)DNA sequences (2)PWMs	(1)Expected average distance between motifs (2)Window size for local nucleotide frequency calculation	HMM	Online	Muscle data
MSCAN	2003	(1)A DNA sequence (2)PWMs	(1)Significance threshold for TFBS (2)Window size (3)Maximum number of motif in a CRM	Statistics	Request	Muscle data Liver data
MCAST	2003	(1)DNA sequences (2)PWMs	(1)P-value cutoff for TFBS (2)Maximum gap length (3)Gap penalty	HMM	Request	Synthetic data *Drosphila* data Human LSF data
ModuleSearcher	2003	(1)DNA sequences (2)PWMs	(1)Number of motifs in a CRM (2)If overlap between TFs allowed (3)If multiple copy of TFs allowed (4)If overlap between TFs are allowed (5)Penalize "incomplete" CRM (6)Use Genetic or A* algorithm (7)Maximum number of iteration (A*) (8)Start with simple search solution (A*) (9)Probability of mutation (G) (10)Number of iteration (G) (11)Population size (G) (12)Number of surviors in each generation (G) (13)Number of top scoring module to return (G)	A*,Genetic Algo	Request	2 orthologous pairs
Stubb	2006	(1)DNA sequences, one or more species (2)PWMs	(1)Window length	HMM	Website	*Drosphila* segmentation
EEL	2006	(1)2 homologous sequences (2)PWMs	(1)Six parameters that weigh different aspects of Binding sites alignment score (2)Background model of "ACGT" (3)Cutoff for sequence and PWMs matching parameter	Statistics	Online	Drosophila eve enhancers Mouse embryonic data
CMA	2006	(1)DNA sequences	(1)Number of single PWMs (2)Distance between TFs (3)Size of CRM (4)Number of iterations of genetic algorithm (5)Population retain in each interation (6)Mutation level (7)If restrict FP/FN (8)Fitness function components	Genetic Algo	Website	Synthetic data T-cell specific genes in TRANSCompel db
ModuleMiner	2008	(1)DNA sequences (2)PWMs	(1)Select database (2)Ensembl IDs	Genetic Algo	Website	Multiple data
Compo	2008	(1)DNA sequences (2)PWMs	(1)If overlap allows (2)Number of TFs in CRM (3)Length of window (4)TP-factors (5)Background sequences	Itemset mining	Online	Muscle data Liver data *Drosphila* data
ModuleDigger	2009	(1)DNA sequences (2)PWMs (3)Background sequences	(1)Support (2)Number of TFs in the CRM (3)Number of CRM should output	Itemset mining	Online	ESC ChIP-chip data

Different CRM detection methods have been developed that differ from each other in the way they tackle the combinatorial search problem. Methods such as for instance ModuleSearcher (Aerts et al., 2003) and ModuleMiner (Van Loo et al., 2008) pose the CRM problem as an optimization problem (e.g. uses a genetic algorithm) with an explicit cost function to be optimized while Compo (Sandve et al., 2008) and ModuleDigger (Sun et al., 2009) rely on itemset mining to first enumerate all possible module combinations after which a statistical filtering strategy is applied to identify the most promising CRMs. Methods also differ in the way they define a module either in the cost function or during the enumeration (for itemset mining approaches). In all methods a CRM is defined as a set of motifs. However depending on the method the description can be more accurate such as e.g. the motifs should occur together within a predefined distance (Aerts et al., 2003; Frith et al., 2001; Frith et al., 2003; Sandve et al., 2008; Sharan et al., 2003; Sun et al., 2009; Van Loo et al., 2008) or the spacing between the motifs sites contributing to the CRMs should be of fixed size. A major distinction can be made between CRM methods that are based on the assumption that a set of coregulated genes should share a common CRM versus those that treat each sequence independently (further referred to as the single-sequence based methods). Cister or ClusterBuster are examples of the latter category: these methods search in a single sequence for potential CRMs that best match a predefined structure as imposed by the model parameters (here a hidden Markov model) using as input the probabilities of each segment matching individual motif models. Methods that do exploit the dependency between the sequences in an input set, in contrast assign a higher weight to CRMs that occur frequently and of which this frequency of occurrence is not likely given the background nucleotide distribution. For the purposes of this review, we shall assume that the motifs are represented as PWMs. Usually motif models (PWMs) from the same protein family are very similar. Be-fore CRM screening, we can first filter out very similar PWMs in different ways (Shobhit Gupta, 2007), e.g. filter PWMs with "Kullback-Leiber" distance below a certain value (Coessens et al., 2003), or group similar PWMs into one PWM.

While traditionally methods identify a CRM as a set of motifs that co-occur more frequently than expected based on the nucleotide background composition of the organism of interest, the more recent methods also assess the specificity of the CRM for the set of input sequences i.e. they compare to what extent a similar CRM occurs in a large set of se-quences randomly sampled from the genome using, a hypergeometric (Sharan et al., 2003), adopted binomial statistic (Sun et al., 2009) or a rank based strategy (Van Loo et al., 2008).

Interestingly, some methods use the frequency of the detected CRMs in the genome as an estimate for their specificity in the input sequences, e.g. CREME (Sharan et al., 2003), ModuleMiner (Van Loo et al., 2008) and ModuleDigger (Sun et al., 2009). By using background sequences these methods only select the CRMs that are more specific for the input sequences but not for the background sequences. Given the

input sequences and the background sequences, CREME calculates the probability of observing a single TFBS on all of the sequences, i.e. co-regulated sequences and background sequences based on hypergeometric distribution. Similarly but not identically, to calculate whether a certain found CRM is specific for the input sequences, ModuleDigger compares the number of sequences observed to contain this CRM in the background sequences. ModuleDigger uses a cumulative binomial distribution to calculate the enrichment score to see how specific this CRM is to the input sequences. ModuleMiner (Van Loo et al., 2008) adopted a leave-one-out cross validation (LOOCV) strategy. In each run, one gene was left out and ModuleMiner constructed a CRM using the remaining genes. This CRM was used to rank all genes in the genome and the position of the left-out gene was determined. Then ModuleMiner uses order statistics to assign a probability to the combination of ranks of the given co-expressed genes. Hence, the resulting p-value represents how well that CRM models the given set of co-expressed genes, comparing with the other genes in the genome. These strategies can increase the specificity of the results especially when the data is very noisy. The features and usages of discussed tools are outlined in Table 1.1.

1.7 Limitations of current CRM screening methods

1.7.1 Limitations of single transcription factor screening

TFBSs or motifs are typically short and degenerate, moreover, recent studies show that DNA sequence alone is an impoverished source of information for TFBSs prediction (Whitington et al., 2009) and that lower binding specificity but stable chromatin stability can also lead to TF binding (Ozsolak et al., 2007). With the availability of ChIP-Seq (Jothi et al., 2008) and ChIP-chip (Buck and Lieb, 2004) data for eukaryotic TFs, it indeed becomes increasingly clear that only in a few cases the physically bound sites correspond to the 'best conserved or highest scoring' sites obtained with a PWM screening (Whitington et al., 2009; Won et al., 2010). This is probably partially due to the fact that PWMs stored in public databases are biased towards sites discovered by their resemblance to the already stored motif model (circular reasoning) but also because other physical factors such as chromatin positioning determine the accessibility of a site (Whitington et al., 2009).

1.7.2 Limitations of combinatorial search

Because of the combinatorial large search space (many different motif combinations that can define a possible CRM) often methods are computationally restricted in the maximal size of the sequence set and/or the maximal number of TF binding

sites (hits of the individual motifs) they can handle. Most state-of-the-art CRM detection methods are typically applied on a dataset of a few sequences consisting of a few 100 bp and a PWM library of at most 50 TFs.

1.8 Possible epigenetic features for CRM screening

Epigenetic refers to heritable phenotypic changes that are causes by mechanisms other than the genetic mutations. Recent work has led to the realization that TFs may also be effective gene regulators in cases of low binding specificity of TFs on sequences but high chromatin stability and accessibility (Ozsolak et al., 2007). Eukaryotic cells exhibit diverse transcriptional profiles across different cell types and conditions and here it is the epigenetic micro-environment that dictates tissue-specific variation. The epigenome adjusts specific genes in our genome landscape in response to our rapidly changing environment.

1.8.1 Nucleosome occupancy feature

Chromatin is the complex of DNA and proteins in which the genetic material is pack-aged inside the cells of organisms with nuclei (Felsenfeld and Groudine, 2003). DNA in eukaryotes is highly packed into nucleosome arrays. The nucleosome is the fundamental unit of chromatin and it is composed of eight octamer of the four core histone proteins (H3, H4, H2A, H2B) around which 147 base pairs of DNA are wrapped. Neighboring nucleosomes are separated from each other by 10-50 bp long stretches of unwrapped linker DNA and typically around 75%-90% of the genome is wrapped in nucleosomes. TF-binding is reduced in nucleosomal DNA. Thus, nucleosomes and TFs compete for access to the DNA, which is a major mechanism by which nucleosomes influence transcriptional activity. AT-content is a major cis factor influencing nucleosome positioning. It is believed that AT-rich tracts deter nucleosomes because these sequences are unusually stiff, thereby resisting the sharp bending required for histone binding. For example, in yeast, nucleosome-depleted TFBSs are linked to high gene activity and low expression noise, whereas nucleosome-covered TFBSs are associated with low gene activity and high expression noise (Dai et al., 2009). For some species, the genomewide nucleosome positioning maps (Kaplan et al., 2009) are already available, e.g. yeast. But for most of the genomes, e.g. human and mouse, this information is not yet available, but several methods have been developed for predicting the nucleosome occupancy (Field et al., 2008; Gupta et al., 2008; Ioshikhes et al., 1996; Kaplan et al., 2009; Peckham et al., 2007; Tolstorukov et al., 2008; Xi et al., 2010).

1.8.2 Histone modification features

It has been observed that epigenetic marks such as the histone acetylation (HAc) can be associated with active promoters and open chromatin (VetteseDadey et al., 1996) and is of particular relevance to transcriptional regulation. The histone code refers to profiles of posttranslational modifications of histone proteins (e.g. acetylation, methylation, phosphorylation, uniquitylation, SUMoylation, and adensosine diphosphateribosylation). For example, the chromatin modification feature H3K4me3 (trimethylation of lysine 4 of histone H3) has long been regarded as a maker for open chromatin and actively transcribed genes (Tony, 2007). The genomewide distribution of this marker was recently mapped in several mouse and human tissues (Barski et al., 2007; Guenther et al., 2007; Guenther et al., 2007; Mikkelsen et al., 2007). Regulatory elements such as promoters and enhancers are associated with distinct chromatin features. Such chromatin features could be used to predict the regulatory elements (Ji et al., 2006; Valouev et al., 2008; Wang et al., 2009). These observations have stimulated the development of approaches that integrate multiple types of chromatin features to improve the accuracy of TFBSs prediction.

1.8.3 DNA methylation feature and CpG islands

DNA methylation is the most studied epigenetic mark and it's very common in bacteria, fungi, plants and animals. In eukaryotic organisms DNA methylation usually occurs only at the cytosine pyrimidine ring. In mammalian, DNA methylation usually occurs at the cytosine of a CpG dinucleotide. CpG dinucleotides constitute only 1% of the human genome and between 70%-80% of all CpGs are methylated. Unmethylated CpGs are grouped in clusters called "CpG islands" that are present in the 5' regulatory regions of many genes. DNA methylation may impact the transcription of genes in two ways. First, the methylation of DNA may itself physically prevent the binding of transcriptional proteins, thus blocking transcription. Second and likely more important, methylated DNA may be bound by proteins known as Methyl-CpG-binding domain proteins (MBDs). MBD proteins then re-cruit additional proteins to the locus, such as histone deacetylases and other chromatin remodeling proteins that can modify histones, thereby forming compact inactive chromatin which is termed 'silent chromatin'. In several types of cancer, CpG islands in the promoter of genes acquire abnormal hypermethyation resulting in heritable transcriptional silencing.

CpG islands on genomic sequences play crucial roles in transcriptional regulation. Generally, methylation related studies are focused on CpG islands (Zhang, 2007) and only the methylation at CpG islands is believed to have a biological significance. For example, highly methylated CpG islands in promoter regions suppress transcription, while lower-level methylated CpG islands favor transcription. Sequence with a

higher GC content tends to contain CpG islands and are thus more likely to be methylated. Furthermore, as was shown in previous genomewide studies (Mavrich et al., 2008; Yuan et al., 2005), variants of poly(dA:dT) sequences were found to be the most dominant nucleosome excluding DNA sequences, confirming that AT-rich (GC-impoverish) sequences have a very low propensity to form nucleosomes (Field et al., 2008). Thus when the experimental DNA methylation information is not available, the GC content feature of a genomic sequence or the fraction of GC bases in a sequence can be used to estimate the compression level of the chromatin structure. Data sources for these features are outlined in Table 1.2.

Features	Data source	Computational algorithms
Sequence conservation	UCSC (Fujita et al., 2011) Ensemble (Hubbard et al., 2002)	/
Nucleosome occupancy	Nucleosome occupancy Atlas Yeast(Lee et al., 2007) Kaplan et al., 2009 Yeast	Kaplan et al., 2009 NuPoP (Xi et al., 2010)
Histone modification	HHMD (Zhang et al., 2010) Human The National Human Genome Research Institute's Histone Database (Sullivanet al., 2000) ChromatinDB (O'Connor&Wyrick, 2007) Yeast ENCODE (Thomaset al., 2007) Cancer Genome Atlas (Boltonet al., 2010)	/
DNA methylation&CpG islands	ENCODE (Thomaset al., 2007) Cancer Genome Atlas (Boltonet al., 2010) MethPrimerDB (Pattynet al., 2006) MethyLogiX (Wanget al., 2008) MethDB (Negre and Grunau, 2006) PubMeth (Ongenaertet al., 2008) MeInfoText (Fanget al., 2008)	CpGislandsearcher (Takai&Jones, 2002) Methylator (Bhasinet al., 2005)

Table 1.2: Data sources for these features.

1.9 Achievements

1.9.1 Part I: ModuleDigger

We developed ModuleDigger, a *cis*-regulatory module detection framework based on itemset mining algorithm which is able to detect *cis*-regulatory module with larger data. Current available tools can handle limited size of data, and seldom check the specificity of a certain CRM for the input sequences with the random genome. By employing itemset mining algorithm, our framework makes it computationally tractable for larger data.

Our results show that our framework outperformed than available methods by using a ChIP-chip data as benchmark data. Different *cis*-regulatory module detection algorithms were applied to the dataset. The results show a qualitatively very different response of the algorithms with respect to parameters of the data such as noise, amount of data and interaction types. These results also prove that our algorithm is useful to provide more insights in the regulation activates of the set of co-expressed genes. The work has been published in the following paper:

Sun, H., De Bie, T., Storms, V., Fu, Q., Dhollander, T., Lemmens, K., Verstuyf, A., De Moor, B., Marchal, K. (2009). ModuleDigger: an itemset mining framework

for the detection of *cis*-regulatory modules. *BMC Bioinformatics*, 10(Suppl 1):S30; doi:10.1186/1471-2105-10-S1-S30.

1.9.2 Part II: CPModule

We proposed a method for detecting CRMs in a set of co-regulated sequences. Each CRM consists of a set of binding sites of TFs. We wish to find CRMs involving the same TFs in multiple sequences. Finding such a combination of transcription factors is inherently a combinatorial problem. We solve this problem by combining the principles of itemset mining and constraint programming. The constraints involve the putative binding sites of TFs, the number of sequences in which they co-occur and the proximity of the binding sites. Genomic background sequences are used to assess the significance of the CRMs. We experimentally validate our approach and compare it with state-of-the-art techniques. We also show on real ChIP-based experiments conducted by Chen et al., 2008 for five key TFs involved in self-renewal of mouse embryonic stem cells how our CRM detection flow can be used to prioritize true combinatorial interactions between the assayed TF and other TFs. The work has been published or under revision of the following paper:

Guns, T., Sun, H., Marchal, K., Nijssen S. (2010). *Cis*-regulatory Module Detection using Constraint Programming. *In Proceedings of IEEE International Conference on Bioinformatics and Biomedicine (BIBM2010), IEEE Computer Society*, BIBM.2010.12.18, 363-368.

Sun, H., Guns, T., Fierro, AC., Thorrez, L., Nijseen, S., Marchal, K. (2011). Unveiling combinatorial regulation through the combination of ChIP information and *in silico cis*-regulatory module detection. ***In revision***.

1.9.3 Part III: ViTraM

The problem of visualizing overlapping modules simultaneously is that the overlap in multiple dimensions complicates the choice of an appropriate layout. Therefore few tools exist that are capable of visualizing modules simultaneously. For instance, tool (Grothaus et al., 2008) for the visualization of multiple, overlapping biclusters in a two-dimensional gene-experiment matrix was developed, as each bicluster is represented in this layout-matrix as a contiguous submatrix, genes and experiments that belong to multiple overlapping biclusters will be duplicated to obtain an optimal layout of the biclusters. This duplication of genes and experiments, however, complicates the biological interpretation of the biclusters.

ViTraM simultaneously identifies multiple overlapping modules and an extension to ViTraM allows group both correlated and anticorrelated genes within a single module. The combination of ViTraM with a gene regulatory network construction

approach allows ViTraM to be easily extended to incorporate additional data sources, ultimately leading to the identification of regulatory modules with associated condition annotation, regulatory motifs, transcription factors and gene ontologies. The work has been published in the following paper and book chapter:

Sun, H., Lemmens, K., Van den Bulcke, T., Engelen, K., De Moor, B., Marchal, K. (2009). ViTraM: Visualization of Transcriptional Modules. *Bioinformatics*, 25(18):2450-2451; doi:10.1093/Bioinformatics/btp400.

Sun, H., Lemmens, K., Van den Bulcke, T., Engelen, K., De Moor, B., Marchal, K. (2009). Layout and Post-Processing of Transcriptional Modules. *In Proceedings of International Joint Conference on Bioinformatics, Systems Biology and Intelligent Computing (IJCBS2009)*, *IEEE Computer Society*, 10.1109/IJCBS.2009.95, 116-121.

Fu, Q., Lemmens, K., Thijs, I., Meysman, P., Sanchez, A., Sun, H., Fierro, C., Engelen, K., Marchal, K. (2010). Directed module detection in a large-scale expression compendium. In: Van Helden J., Toussaint A., Thieffry D. (Eds.), *Methods in Molecular Biology-Bacterial Molecular Networks*. New York: Springer New York.

I am also contributing to a web interface development MotifSuite, which offers a set of perfectly integrated well performing softwares for detecting (*de novo*), selecting, comparing and allocating regulatory motifs. The suite was tested on *E.coli* datasets with positive results. The work is in preparation for a journal paper:

Claeys M., Storms V., Sun, H., Marchal K. (2011). MotifSuite: work flow for regulatory motif detection with various motif assessment tools. ***In preparation***.

1.9.4 Summary

The sections relating to *cis*-regulatory module are partially took for the following review paper:
Sun, H., Storms, V., Meysman, P., Marchal, K. (2011). The past and future trends of *cis*-regulatory module detection, from DNA sequence based to multi-evidence based. ***In preparation***.

Chapter 2

Association rules mining algorithms

2.1 ARM algorithms

ARM (association rules mining) algorithms were initially developed in the database community to analyze market basket data. Basket data consists of information on transactions or sets of items that have been purchased together. Transactions are stored in a database. Analysis of these past purchases helps the management of a store to decide on products to put on sale, the design of coupons, the way to place merchandise on the shells to maximize profit, *etc.*

ARM algorithms are thus useful for mining large collections of data. Our in-house-build tools, ReMoDiscovery (Lemmens et al., 2006), DISTILLER (Lemmens et al., 2009), ModuleDigger (Sun et al., 2009), and CPModule (Guns et al., 2010; Sun et al., 2011 in revision) (collaborate with DTAI machine learning group, department of computer science, KULeuven) all make use ARM algorithms, a description of ARM algorithms is given below. We focus in particular on CHARM algorithm because DISTILLER and ModuleDigger, are both based on CHARM. In the last section of this chapter we discuss applications of ARM algorithms in the Bioinformatics domain.

Assume one has a database containing all genes together with the motifs that are present in the promoter regions of these genes. Given these data, ARM algorithms are able to find motifsets that occur across set of genes in a very efficient way. In the usual terminology of ARM algorithms, a gene is called a transaction, while the motif corresponds to an item. A set of motifs that shared by a number of

genes is an itemset. The number of common genes is the support of that itemset. An itemset is called frequent if its support exceeds a prespecified threshold: the support constraint. A frequent itemset is called maximal if it is not a subset of any other frequent itemset. A frequent itemset is called closed if there exists no proper superset with the same transaction as it.

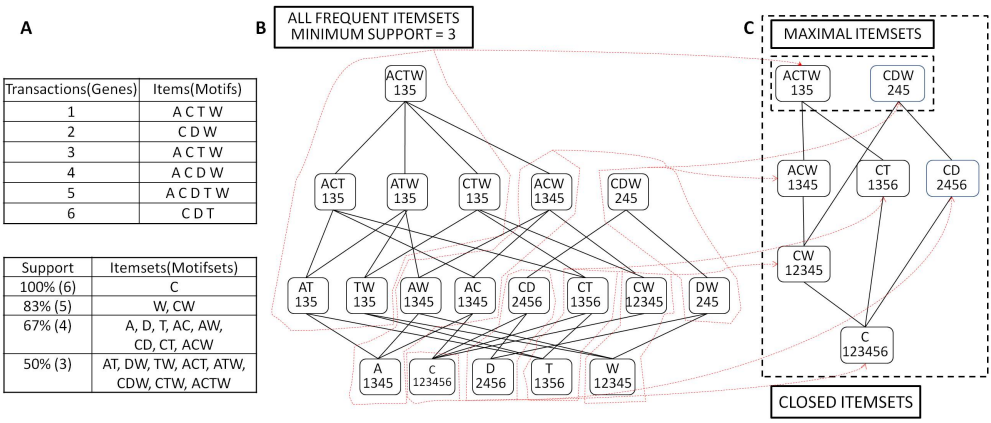
All possible itemsets can be represented in a lattice structure, i.e. all possible combinations of items in different itemsets of various sizes. In a naive way, all these combinations could be tested one by one to check whether they are frequent, closed or maximal. However for large databases this approach is computationally not feasible and we need to rely on efficient algorithms such as ARM algorithms. These algorithms start from a database of items and transactions and in a first step they search for frequent itemsets. In the second step they learn association rules from the frequent itemsets.

Since the association rules themselves are not very important for our research, our focus will be on the efficient identification of frequent, closed and maximal itemsets. These itemsets or sets of motifs that satisfy particular constraints or supports can be interpreted as *cis*-regulatory modules (or motifset, combinations of motifs). We will thus make use of the association rules mining algorithms to find *cis*-regulatory modules.

2.2 CHARM algorithm

The CHARM algorithm (Closed Association Rule Mining) is an efficient algorithm for identifying closed itemsets (Zaki & Hsiao, 2002). CHARM explores the itemset space and transaction space simultaneously over an IT or itemset-transaction tree search space. In this tree, a node consists of an (itemset $\times$ transaction) pair (Figure 2.1). CHARM searches this tree using a depth-first search strategy exploiting the notion of equivalence classes. In the IT-tree, each node is in fact a prefix based class. Two itemsets belong to the same class if they share a common k-length prefix, determined by an ordered list of k gene names. By construction, the children of a node all belong to the same equivalence class X since they all share the same prefix X (or the same geneset). In Figure 2.1, motif A and motif T, for instance, belong to one equivalence class [X]. Note that motif A and motif D does not belong to this class since itemset (motif A, motif D) is not frequent. So a class represents items with which the prefix can be extended to obtain a new frequent node. No subtree of an infrequent prefix has to be examined.

The frequent itemsets can readily be determined in the IT-tree framework: for a given node or prefix class the intersections of the transactions of all pairs of elements is determined and it is checked whether they meet the minimum support. A pass over the database to check the support of an itemset is not necessary anymore. Each



Transactions(Genes)	Items(Motifs)
1	A C T W
2	C D W
3	A C T W
4	A C D W
5	A C D T W
6	C D T

Support	Itemsets(Motifsets)
100% (6)	C
83% (5)	W, CW
67% (4)	A, D, T, AC, AW, CD, CT, ACW
50% (3)	AT, DW, TW, ACT, ATW, CDW, CTW, ACTW

Figure 2.1: **Example of a database of gene-motif combinations**. (A) and frequent (B), closed and maximal itemsets (C). Panel A shows a transaction base in which the motifs are the items and the genes are the transactions: gene 1, for instance, in its promoter region, we found motif A, motif C, motif T and motif W. In the lower part of panel A, the support of the itemsets, or the number of genes these combination of motifs, are shown. Itemset (motif A, motif C, motif T), for instance, has a support of three, meaning these three genes have these three motifs in common. Panel B shows the lattice of all possible itemsets. The black nodes indicate the frequent itemsets if a minimum support of two is required. The red dashed line indicates that itemsets have the same support and will result in the same closed itemset, indicated with the red arrow. Panel C shows the closed and maximal itemsets.

resulting frequent itemset is a class on its own that can be expanded recursively. The power of this class approach is that it breaks the original search space into independent subproblems. Any subtree rooted at node X can be treated as a new problem and only this subproblem has to fit in the memory.

CHARM makes use of four properties of the equivalence classes to skip levels in the IT-tree structure. Assuming two members, X_i and X_j, of the same equivalence class that are ordered such that $X_i < X_j$, the following properties apply:

- Rule 1: if X_i and X_j have the same support, then X_i can be replaced by X_i U X_j and element X_j does not need to be considered anymore.
- Rule 2: if the support of X_i is a subset of the support of X_j, then X_i can be replaced by $X_i \bigcup X_j$ since they have identical closures (i.e. they result in the same closed itemset), but element X_j cannot be removed.

- Rule 3: if the support of X_j is a subset of the support of X_i, then X_j can be replaced by $X_i \bigcup X_j$ since they have identical closures, but element X_i cannot be removed.
- Rule 4: if the support of X_i and X_j is different, then no element of the class can be eliminated since both X_i and X_j will lead to different closures.

By making use of these rules, parts of the IT-tree can be passed over and the IT-tree can be searched in very efficient way. Because rule 1 and 2 favor this skipping of levels, CHARM orders the items in increasing order of support such that there will be more occurrences of rule 1 and 2.

CHARM starts by listing the items (motifs) by increasing number of items. In the next step, we check the item-transaction pair with the minimum number of items, motif A, C, D, T, W will combined with the other items (or motifs), and only the combinations that are frequent will be kept. For example, motif A will be extended with motif W. Since the support of motif A and motif W is different, no elements can be discarded. The resulting frequent itemset is (motif A, W) with a support of four. Then this itemset (motif A, W) will be extended with motif C. Since the support of itemset A is part of the support of motif C, CHARM will replace motif A in the tree by (motif A, C), resulting in the itemsets (motif A, C), the same for itemset (moitf A, W) and itemset (motif, A, C, W). This search strategy drastically reduces the number of combinations that need to be tested. Using this strategy, the IT-tree can be searched in a very efficient way.

2.3 Application of ARM algorithms on Bioinformatics

Recently, ARM algorithms found their way to the field of Bioinformatics. Several applications make use of the Apriori algorithm (Chiu et al., 2006; Ivan et al., 2007; Morgan et al., 2007; Oyama et al., 2002; Sandve et al., 2008) or Apriori-based algorithms like (Artamonova et al., 2005; Artamonova et al., 2007; Becquet et al., 2002; Brazma et al., 1997; Carmona-Saez et al., 2006; Creighton & Hanash, 2003; Rodriguez et al., 2005; Fang et al., 2010). Some of those applications search for closed itemsets (Huang et al., 2007; Okada et al., 2007; Pham et al., 2004).

Although most of these Bioinformatics applications rely on existing itemset and rule mining algorithms, sometimes new algorithms are being developed to take into account specific properties of the biological problems at hand (Georgii et al., 2005; Lopez et al., 2008; Tamura & D'haeseleer, 2008). In almost all cases, the approach consists of four steps. In a first step, data is gathered and transformed into a matrix format. Usually, the entries in the matrix are binary. In a second step, the matrix is processed to obtain frequent itemsets. In a third step association rules can be derived from the frequent itemsets. Some of the approaches skip the third

step. In the latter case, the frequent itemsets themselves form the result. Because ARM methods tend to generate large amounts of itemsets or association rules, a final filtering or post-processing step is usually introduced to obtain biologically interesting itemsets or rules.

Diverse usages of ARM algorithms in the field of Bioinformatics have been applied. Lin et al., (2006), for example, make use of the database of the HIV Drug Resistance database and ARM algorithms to find relationships between mutations in the HIV protease gene and antiretroviral drug treatment. In another example, ARM algorithms were used to obtain sets of COGs (Clusters of Orthologous Groups of Proteins) associated with a phenotype (Tamura & D'haeseleer, 2008). ARM algorithms have also been used to study protein interactions or to annotate proteins. Oyama et al., 2002 reveal rules related to protein-protein interactions, while Ivan et al., 2007 study ligand-protein interactions. Both studies collect information on different properties of the proteins like functional category, protein domain information or residue composition.

Subsequently, rules are derived that provide information on which of these properties occur together very frequently in interactions. These rules could provide novel insight in the characteristics of the interactions. ARM algorithms have also been used for the annotation of proteins based on protein domain composition (Chiu et al., 2006) or protein sequence similarities (Rodriguez et al., 2005).

One of the earliest Bioinformatics applications of ARM approaches concerned the search for combinations of transcription factor binding sites in the upstream regions of yeast genes that occur more frequently than expected by chance (Brazma et al., 1997). Brazma et al., (1997) screened the promoter regions of yeast for the presence of motif instances. Subsequently they searched for the frequent sets of regulatory motifs and used these frequent itemsets to derive association rules, such as "if motif 1 and motif 2 are present, then motif 3 is also present". The order of occurrence of the motifs could not be taken into account by the approach of Brazma et al., 1997. Despite this shortcoming, this kind of research is still very useful to study combinatorial regulation. More recently, ARM was used to search for frequent combinations of regulatory motifs that are located close to each other in the DNA sequence of the human genome (Morgan et al., 2007). Another approach, developed by Doi et al., 2008, uses as input a set of userdefined genes and searches for significant combinations of regulatory motifs in their upstream regions. The search for motifs is performed via both *de novo* motif detection and screening with known motif matrices. Recently, ARM algorithm also has been applied for Biomarker discovery or differential coexpression detection (Fang et al., 2008). When searching for the combination of genes co-expressed in the case experiment, at the mean time, the corresponding genes are also checking for the expression situations in the control experiments. In this way, we can find set of differentially expressed genes in the case and control experiments, these genes might be the reason for the disease or abnormality.

In spite of the previously mentioned applications, ARM methods are not used very frequently for network inference although they offer many advantages:

- ARM algorithms search for all solutions in an exhaustive manner so in contrast to optimization-based methods, ARM algorithms do not suffer from problems with local optima.
- Because ARM methods find all solutions at once, they do not need to mask previous results to find the next result or to use multiple initializations to find all possible results.
- In contrast to other methods, overlapping solutions are possible and generated in a natural and straight-forward way. This is obviously an interesting property, for instance while searching for biclusters. Since a gene can be involved in more than one biological function, a gene can also belong to more than one bicluster.
- In principle, each data set from which hereditary properties can be derived, can be included in the analysis in a straight-forward way. These hereditary constraints allow the ARM algorithms to search efficiently by employing pruning steps. ARM methods are therefore extremely useful for data integration since many data sources can be included.

The reason why these methods have not been used more often can be traced back to two major challenges associated with the use of ARM algorithms for network inference.

A first major challenge when applying ARM algorithms is the large amount of generated frequent itemsets or rules. The use of maximal or closed itemsets (instead of all frequent itemsets) can partly solve this problem but still too many itemsets and rules remain. Therefore, the resulting itemsets and rules need to be filtered and analyzed in a computationally efficient and biologically meaningful way to remove redundancy (for instance, in the form of heavily overlapping sets) and separate interesting results from less interesting results. Tuzhilin & Adomavicius (2002) investigated strategies to tackle this problem. They suggest rule filtering in a context where the user is only interested in the rules concerning for instance a particular group of genes. Rule grouping can be applied when many similar rules are generated.

In this case, it would be useful for the biologist to analyze similar rules together to obtain a high-level overview of the inferred rule classes.

The definition of biological relevance may differ from application to application. The most appropriate filtering approach can therefore be very different in each ARM application. For instance, Artamonova et al., 2005, 2007 for instance employed ARM techniques for the discovery of incorrectly annotated proteins in protein

databases. They calculated the confidence of all derived association rules. The confidence of a rule X= >Y was defined as the ratio of the support of all itemsets containing X and Y to the support of all itemsets containing X. Artamonova et al., 2005, 2007 were especially interested in those association rules with a confidence close to one, since the exceptions on this rule might indicate that the corresponding proteins are wrongly annotated. Oyama et al., 2002 investigated properties of interacting proteins. Many association rules were derived, including rules of the form "if the protein has domain A then it also contains domain B".

These kinds of rules only say something about one protein, not about an interaction. Since Oyama et al., 2002 were only interested in those rules that describe a relationship between two interacting proteins, they had to apply a filtering procedure to obtain only those rules that contain information on two interacting proteins. In summary, it is a non-trivial challenge to define the biologically relevant information and a corresponding measure of interestingness for each specific application. In chapter 4 and chapter 5, we suggest a statistical significance enrichment calculation based on cumulative binomial distribution.

Chapter 3

ModuleDigger: *Cis*-regulatory module detection based on itemset mining

3.1 Introduction

In eukaryotic genomes transcriptional regulation is often mediated by the concerted interaction of several transcription factors and cofactors (Davidson, 2001). Each transcription factor recognizes its own binding site or regulatory motif. The combination of several transcription factor specific motifs is called a *cis*-regulatory module (CRM). The presence of a *cis*-regulatory module thus determines the transcriptional response of a specific gene. As coexpression might imply a similar mechanism of coregulation, coexpressed genes can be searched for the presence of statistically overrepresented CRMs. Some strategies have been developed to search *de novo* for the best transcription factor binding site combination, such as for instance CisModule (Zhou&Wong, 2004). The complex nature of the problem, however, still poses some restrictions on the applicability of these *de novo* algorithms. Most of the more pragmatic module detection methods are combinatorial search strategies that start from a set of binding sites for individual motifs. These binding sites are obtained by screening intergenic sequences with each TF-specific position weight matrix (PWM). Subsequently these methods search for the motif combination that is statistically most overrepresented in a set of genes of interest, as compared to the background (Aerts et al., 2004; Sharan et al., 2003). Although these algorithms can in principle be applied to sets of coexpressed genes, most of them do not explicitly assess the specificity of the overrepresented module for the

observed expression pattern in the coexpressed geneset. Exceptions are for instance CREME, which provides an extensive statistical framework and ModuleMiner (Van Loo et al., 2008), which does apply a leave one out strategy in combination with a genomewide ranking to define the modules most specific for the coexpressed geneset as compared to the remainder of the genome. The drawback of the latter method is that the underlying optimization procedure is computationally very intensive restricting its use to relatively small sets of genes and a small number of TFs.

3.2 Module detection framework

The analysis flow we used is outlined in Figure 3.1. Like other module detection methods, our method starts from an existing library of PWMs extracted from TRANSFAC (step 1). All intergenic sequences of a coexpressed or coregulated set of genes are screened with those PWMs to identify per PWM the p-value of the best hit in each sequence. The search for CRMs then boils down to searching through an exponentially large number of combinations of these individual binding sites (step 2). Traditional optimization based methods rely on heuristics to make this search computationally tractable; however, such methods come with no guarantee that a globally optimal solution will be found. In contrast, here we applied a strategy from itemset mining (see Methods). Itemset mining approaches exhaustively investigate all possibly interesting solutions (in this case, motif modules or CRMs), and hence do not suffer from local optima problems. They are able to do this despite the exponential number of combinations of binding sites by exploiting properties of the search space that allow for efficient pruning during the search. The output of our itemset mining algorithm is an exhaustive list of all possible motif modules (or potential CRMs). To filter the biologically most interesting CRM candidates from this list, we compute a score for each of the potential CRMs (see Methods). This score assess how specific this CRM is for the set of genes in which it occurs, and for the cluster of input genes as a whole. A CRM is considered significant for the genes in which it occurs if that geneset does not contain many other overrepresented CRMs, and it is considered specific for the whole cluster of input genes if the CRM is statistically more overrepresented in this cluster of genes than in the remainder of the genome. By iteratively applying this scoring system we can prioritize a list of non-redundant and most promising CRMs. The higher the rank of a CRM in this list, the higher its potential of being a biologically valid one (as it is the most specific for the genes in which it occurs and the most explanatory for the whole set of input genes). As such, our framework combines advantages associated with the efficiency of an itemset mining search strategy with those related to statistical scoring measures.

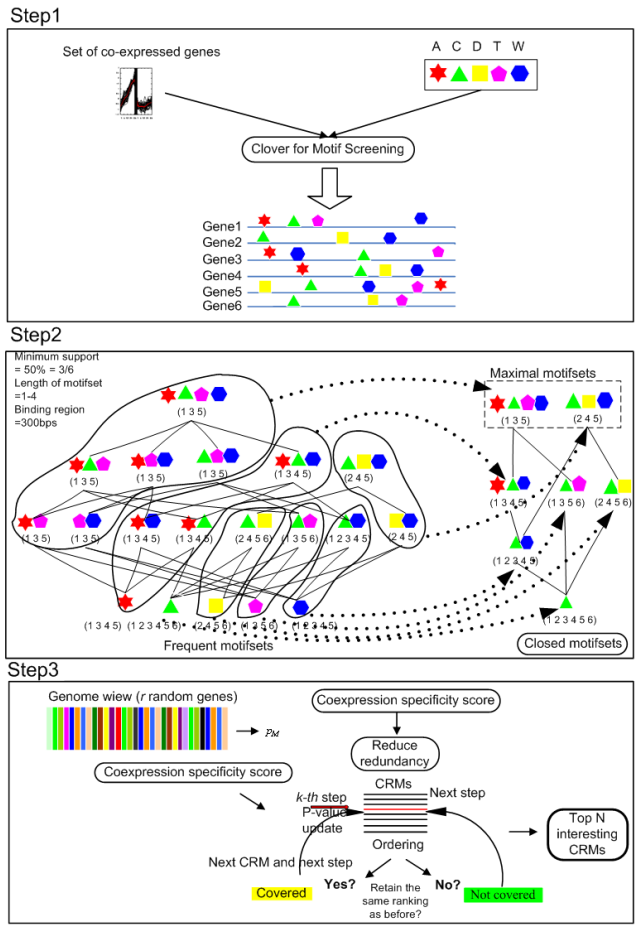


Figure 3.1: **Analysis flow**. The input consists of a set of coexpressed or coregulated genes. Step 1: Screening the intergenic sequences of these genes with a library of PWMs. Step 2: Apply our itemset mining strategy to find all the modules (closed motifsets) that occur in a minimal number of genes in the dataset (a minimum support defined by the user). Step 3: Determine the final rank of each module. An original ranking is assigned to each module based on the module specificity score. In the update step the rank of overlapping modules is reduced by iteratively assigning an updated score.

3.2.1 Enumerating all frequent closed CRMs

For the identification of modules, defined as combinations of individual motifs, we rely on itemset mining. Itemset mining searches for the combination of items (in our case the motifs) that are supported by a minimal number of transactions (in our case the genes). We used an implementation provided in the package MINI (Gallo et al., 2007) which is based on CHARM (Zaki&Hsiao, 2002). CHARM searches for closed sets using a dual itemset-tidset (motifset-geneset) search tree. A closed set is a set of motifs (or a potential module) that is frequent (i.e. simultaneously contained in the intergenic region of a minimal number of genes) and that can no longer be extended by additional motifs without decreasing the number of genes with all these motifs in their intergenic region.

CHARM is designed to efficiently limit the number of combinations to be tested if different itemsets (or motifsets) are related to each other by a valid "subset" relation, meaning an itemset can only satisfy all constraints if all of its subsets do. A consequence is that we can search for modules by starting with very small motifsets (containing just one motif), gradually expanding them, and stopping (or pruning) the search once a motifset is reached which does not meet a lower bound on the number of genes that contain that motifset. This pruning step results in a massive speed-up, making the method applicable to large data sets. Implementing the subset relation for the motif data is straightforward as the motif matrix is a binary matrix: a target gene has a motif instance for a regulator if the corresponding gene-regulator entry in the motif matrix is equal to one. In our set up an itemset was called valid if it contained at least two genes.

CHARM outputs all possible closed motifsets (or equivalently closed CRMs). This list is exhaustive and still contains many redundant (i.e. partially overlapping modules) modules as well as modules that are not biologically interesting because they are not specifically associated with the set of genes in our benchmark set.

3.2.2 Assigning a rank to each CRM

To assess the statistical significance of the selected modules we adapted the filtering strategy described in MINI. The scoring scheme developed here depends on one score outlined below: the module specificity score.

3.2.3 Module specificity score

We formulate a null hypothesis under which we assume that the motifs are the independent random variables, meaning that each motif has its own specific probability of occurrence in any given gene, independent of the presence of the

other motifs in the gene. The probability of each individual motif m is derived from its frequency of occurrence: $f_m = \frac{c_m}{N_g}$ where c_m corresponds to the number of intergenic sequences in the genome that contain at least one hit of the motif, and N_g is the total number of background genes considered The independence assumption implies that the probability of finding a particular module (being a set of motifs $m \in M$ n a gene equals $p_M = \prod_{m \in M} f_m$, the product of the individual probabilities of each of the single motifs. Using these module probabilities, the probability of finding a particular motifset M in a set of s genes out of the total cluster of n genes by chance can be calculated by the binomial distribution:

$$p_c^M = \binom{n}{s} p_M^s (1 - p_M)^{n-s} \tag{3.1}$$

The probability of finding a motifset M in at least t of the n genes is calculated by means of the cumulative binomial distribution function, as

$$p_c^M = \sum_t^n \binom{n}{t} p_M^t (1 - p_M)^{n-t}, p_M = \prod_{m \in M} f_m \tag{3.2}$$

Stronger deviations from the null hypothesis assuming motif independence are revealed by smaller values of ${P_c}^M$, which may in turn reveal an association between the genes containing the motifs in M.

3.2.4 Iterative p-value updating of the module specificity score

We have noted that the set of closed CRMs is already a reduced representation of the set of all frequent CRMs across the set of input genes. Additionally, the module specificity score from Equation 3.2 allows us to rank CRMs in order of decreasing significance (i.e. in order of increasing p-value). However, in practice this list will still contain many partially overlapping modules. For instance, consider two CRMs which occur in almost the same genes and of which the first is composed of two motifs (M1 and M2) while the second module consists of three motifs, partially overlapping with the motifs of the first module (M1, M2, M3). It is not uncommon that both such highly redundant CRMs are highly ranked in the list of CRMs after sorting it in order of decreasing significance. To avoid the output being overwhelmed by a large number of highly redundant CRMs, we need to correct for redundancy between CRMs, and we do this by means of an iterative procedure that in each iteration selects the next most interesting CRM, conditioned on the CRMs already selected so far.

To start, the list of closed CRMs is sorted according to the module specificity scores calculated as in Equation 3.2. The CRM on top of the list is then selected as the most interesting CRM and removed from the list. To select the subsequent CRMs, an iterative procedure is applied. In each iteration, the p-values of all CRMs still

in the list are adjusted, ensuring that the CRM with smallest adjusted p-value remains on top of the list. This p-value adjustment (described in detail below) is designed to penalize CRMs that overlap with already selected CRMs, in order to make sure that selected CRMs are as non-redundant as possible with the ones that have previously been selected. Subsequently, the top-ranked CRM is selected and removed from the list as well.

To explain the iteration more in detail, let us assume that k CRMs have already been selected, with gene sets G_i for i=1,...,k (i.e. we are now in iteration k). The set of genes of all already selected CRMs is denoted by $\hat{G_k} = \bigcup_{i=1:k} G_i$. Let M be a motifset of a CRM in the list of which the p-value (Equation 3.2) will need to be adjusted. We will discuss how the module specificity score (Equation 3.1) can be adjusted; then to adapt the module specificity score from Equation 3.3.

All we will do is adapting the way in which p_M is computed in Equation 3.2: the probability that a random gene's motifset contains all motifs from M. As noted earlier, all motifs from M can simultaneously be among a gene's motifs by pure chance, assuming independence of the motifs. However, after a few iterations, it may also be attributable to associations already identified in previous iterations by already selected CRMs. In particular, for any gene g, all motifs that have been part of the motifset of an already selected CRM containing g in its geneset, have already been associated with gene g. Let us denote this set of motifs for a gene g by $M_k(g)$. Then, we adapt p_M in the following way (where again N_g is the number of background genes):

$$p_M{}' = \frac{N_g - \left|\hat{G_k}\right|}{N_g} \left(\prod_{m \in M} f_m \right) + \frac{1}{N_g} \sum_{g \in G_{k-1}} \left(\prod_{m \in M \setminus M_k(g)} f_m \right) \tag{3.3}$$

Here the first term captures the probability that all motifs from M are associated in a random gene, assuming that the motifs occur independently of each other. At iteration k, such genes are estimated to occur with a prior probability of $\frac{N_g - \left|\hat{G_k}\right|}{N_g}$, and the probability that they do contain all motifs in M is estimated as $\prod_{m \in M} f_m$. The second term captures the probability that a gene belonging to an already selected CRM contains all motifs from M. This probability is possibly larger than under the independence model, estimated by multiplying just those f_m for motifs that were not part of the already selected CRMs containing the gene g, i.e. for motifs m that do not belong to $M_k(g)$.

Note that this adjusted value of $p_M{}'$ reduces to the initial definition of p_M in Equation 3.2 if no CRMs have already been covered (i.e. if $k = 0$). Then the first factor in the first term is equal to one, and the second term is equal to 0. However, for larger values of k it can only increase in value. As a result, Equation 3.2 can only increase. This means that we can avoid having to adjust all p-values for all CRMs still in the list, and still be able to select the CRM that is most significant

after adjustment. We can do this by starting with the CRM at the top of the list, adjusting the p-value, and reinserting the CRM with adjusted p-value in the list in order to maintain the correct order. If the new position after reinsertion is still on top (i.e. its adjusted p-value is smaller than all p-values for all other CRMs, whether already adjusted or not), this means that it would remain on top even after adjusting the p-values of the lower-ranked CRMs. Hence, it can be selected as the next CRM in the output and removed from the list, thus ending the iteration.

The pseudocode of the entire iterative algorithm is given below. The set R of all closed CRMs is sorted according to its module specificity score. The most highly ranked CRM is then selected and removed from R (steps 1-4).

Algorithm (R)
1: for each R do
2:
3: sort R in ascending order of these p-values
4: select the top-ranked CRM from R and remove it from R
5: k :=1
6: repeat until a sufficient number of CRMs are selected:
7: CRM:= the top ranked CRM in R
8: adjust the p-value of CRM
9: insert CRM in R to keep R sorted in order of increasing p-value
10: if CRM remains top-ranked after reinsertion:
11: select CRM and remove it from R
12: k:=k+1

Then the iterative updating and selection of CRMs starts. If after the updating of its p-value the CRM remains top-ranked in R, then that CRM is considered to be interesting and is selected (steps 7-12). The iteration counter k is incremented and the next iteration starts. The iteration can be stopped as soon as enough CRMs have been selected.

3.3 Results

3.3.1 Dataset

From the UCSC database (human assembly of NCBI 35) (Kent et al., 2002) we could retrieve a match for 333 gene names out of the 353 names originally listed as being cobound by OCT4, SOX2 and NANOG. We retrieved the corresponding 1000 bp intergenic sequences of these 333 genes from UCSC. Only those sequences were retained for which the binding of OCT4, SOX2 and NANOG was located in

the 1000 bp proximal promoter region. This resulted in 116 intergenic sequences known to bind OCT4, SOX2 and NANOG.

The application of chromatin immunoprecipitation combined with DNA microarray techniques (ChIP-chip) in eukaryotes allows the genomewide mapping of the physical interaction between a TF and its target gene. Our test set was derived from a genomewide ChIP-chip analysis performed by Boyer et al., 2005. It consists of 116 genes that co-bind three core TFs, OCT4, SOX2, NANOG (involved in plurypotency and self-renewal) in their 1000 bp proximal promoter region. Moreover, the three TFs bind in each other's close proximity turning them in a true case example of a CRM. The advantage of this dataset over previous ones is that it is much larger (the muscle dataset, for instance contains 12 genes), which allows us to fully exploit the potential of our method.

Note that ModuleDigger will normally be applied to sets of genes that are coexpressed, as identified for example by microarray data. Here the set of genes is selected based on ChIP-chip data instead. While this may be unusual in practical applications, knowing exactly which regulators bind the intergenic regions of the set of genes selected, allows us to better assess the performance of our method.

3.3.2 Running parameters for ModuleDigger

Potential binding sites for individual motifs were identified by screening the intergenic sequences of each of the genes of the benchmark data set by motif models described in TRANSFAC. Screening was performed with the method of Hertzberg et al., 2005. The advantage of this method is that it converts screening scores to p-values by using a randomization strategy. Using p-values instead of raw scores allows comparing motif hits obtained with motif models that are different in length. A motif matrix is compiled by discretizing the screening results with a one indicating that the particular TF contains at least one hit of the corresponding PWM within the upstream region of the gene and a zero that it does not contain a hit. A threshold on the Hertzberg screening p-value of 0.4 was chosen.

To estimate for each TF the number of occurrences of its binding site on a genomewide level (module specificity score see below), we selected 5000 random sequences with a length equal to the length of our benchmark sequences (1000 bp). Frequencies of genomewide occurrence were derived by converting the screening results to a corresponding random motif matrix, discretizing this matrix with a similar Hertzberg p-value threshold as for the test data and counting for each TF the number of occurrences (ones). The minimum support parameter of ModuleDigger, specifying the required minimum number of genes in a CRM, was set to two. For the tests outlined in Table 3.1, we set the parameters such that all CRMs contained exactly three motifs. For the tests outlined in Table 3.2 (shown in next page), we choose parameter settings such that all CRMs contain two or three motifs.

Number of TFs included												
Test set	10 TFs			20 TFs			30 TFs			40 TFs		
Runs	RANK	S	FP	RANK	S	FP	RANK	S	FP	RANK	S	FP
1	6	y	1	18	y	3	41	y	29	15	y	12
2	12	y	2	9	y	2	35	y	3	153	y	2
3	12	y	2	11	y	3	45	y	39	55	y	1
4	11	y	3	14	y	14	71	y	66	54	y	9
5	3	y	1	12	y	1	15	n	18	138	y	52
6	1	y	0	8	y	3	38	y	1	48 y	23	
7	1	y	0	4	y	3	52	y	3	141	y	49
8	2	y	0	6	y	3	58	y	1	147	y	3
9	4	y	2	12	y	11	32	y	1	155	y	52
10	1	y	0	5	y	3	45	y	39	158	y	81
Average	6.3	/	1.1	10	/	4.5	43.2	/	20	108	/	28.4
Median	5	/	1	10	/	3	43	/	10.5	139	/	17.5
Std	4.7	/	1.1	4.3	/	4	15.2	/	22.6	56	/	28.0

Table 3.1: Running ModuleDigger in the presence of a gradual increase in noise. For each specified number of TFs, 10 different runs were performed which differed in the PWMs randomly selected from TRANSFAC. Average, Median and Std: average, median and standard deviation of the rank of the biologically valid module (OCT4, SOX2 and NANOG). RANK: the rank the valid module received in our output; Score of valid module versus random modules (S): assesses whether the score of the true module is higher than the score of an equally ranked module in a randomized dataset (y =yes, n=no). Number of false positives (FP): the number of modules in the randomized dataset that contained a score higher than the score of the valid module in our benchmark dataset.

3.3.3 Benchmarking ModuleDigger

To test ModuleDigger we ran it on the 1000 bp proximal promoter regions of all 116 genes. As mentioned above, ModuleDigger uses a two step approach: it first exhaustively enumerates all CRMs that occur in the benchmark geneset and subsequently assigns a rank to all CRMs that is proportional to the specificity of the module for the geneset in which it occurs and for the set of input genes as a whole. For benchmarking our method we considered only the module consisting of the three TFs OCT4, SOX2 and NANOG, as a valid module. All other modules were considered biologically invalid. Note that this is a conservative assumption, which may result in an overestimation of the number of modules considered to be biologically invalid as the genes of our dataset may contain other yet uncharacterized CRMs. The performance of the algorithm is assessed by the average rank the valid module receives after running the algorithm. In our test we started off with the simple case in which we only used ten TFs as input (the three true TFs together with seven randomly sampled TFs) (see Methods for Running parameters). The complexity of the problem was increased by gradually including more randomly selected TFs (20, 30 or 40 TFs). To asses the statistical significance of the ranked modules, we used a strategy described by Tusher el al. We compared the score that the valid module received with the score of a module that received a rank similar to the one of the valid module, in a randomized version of the dataset (see Methods, Figure 3.2). We can then conclude that if the score of the biologically

Table A

Method	Clover			Module Searcher(A*)			ModuleDigger		
Running time	1.6min			0.5min			10s		
	NM	RR	Sn	NM	RR	Sn	NM	RR	Sn
OCT4,SOX2,NANOG	0	0	0	2.4	10	1.1%	6	10	27.6%
OCT4, SOX2	3.9	10	45.3%	0	0	0	2	10	49.1%
OCT4, NANOG	0	0	0	0	0	0	3	10	42.2%
SOX2, NANOG	0	0	0	2.4	10	0.9%	9	10	28.4%

Table B

Method	Clover			Module Searcher(A*)			ModuleDigger		
Running time	4min			0.5min			20s		
	NM	RR	Sn	NM	RR	Sn	NM	RR	Sn
OCT4,SOX2,NANOG	0	0	0	0	0	0	5	10	28%
OCT4, SOX2	6.8	10	45.3%	21	5	2.8%	18	10	49%
OCT4, NANOG	0	0	0	0	0	0	23	10	42%
SOX2, NANOG	0	0	0	21	4	0.5%	20	10	9%

Table 3.2: Comparison between methods. A) For all algorithms we used as input the benchmark set, the PWMs of OCT4, SOX2, NANOG and one random PWM. NM: number of modules present in the output for those runs where the RR=1 (average over runs where RR=1). SRR (summation of recovery rate): number of runs for which the output contained a module corresponding to the valid modules (OCT4, SOX2, NANOG or combinations thereof). Sn: number of genes containing the valid module in the output (average of runs for which RR was 1). B) Similar as A but using OCT4, SOX2, NANOG in combination with 7 randomly selected PWMs.

valid module is higher than the score of an equally ranked module in more than $> 90\%$ of the randomized datasets, it was successfully assigned a significantly high rank by ModuleDigger. We also assessed the number of false positive modules that should be expected to be discovered by ModuleDigger, by counting the number of modules in the randomized dataset that contained a score higher than the score of the true module in our benchmark dataset. For testing the noise sensitivity of our method we applied ModuleDigger on the benchmark dataset of 116 genes and gradually increased the number of TFs that composed the input search space. Each combination consisted of the experimentally verified TFs together with a number of noisy TFs randomly sampled from TRANSFAC. Each test was run 10 times. In each run we recorded the rank and the score of the biologically valid module, consisting of three TFs OCT4, SOX2 and NANOG. The significance of the ranking was assessed based on order statistics as described by Tusher el al. The idea behind this approach is represented in Figure 3.3. In a randomized set we expect the scores to be neutral, not reflecting any true signal. A randomized set is composed by searching for modules in the 116 sequences using a TF set as input which does not contain the true TFs known to be present in the data. Different modules obtained in randomized and in the real set are ranked according to their score and plotted against a baseline. The baseline is constructed by making 10 randomized sets, ranking their modules and averaging the scores of the equally ranked modules. The baseline thus consists of average scores of randomized sets. When comparing a random set with the baseline we expect all values to be close to

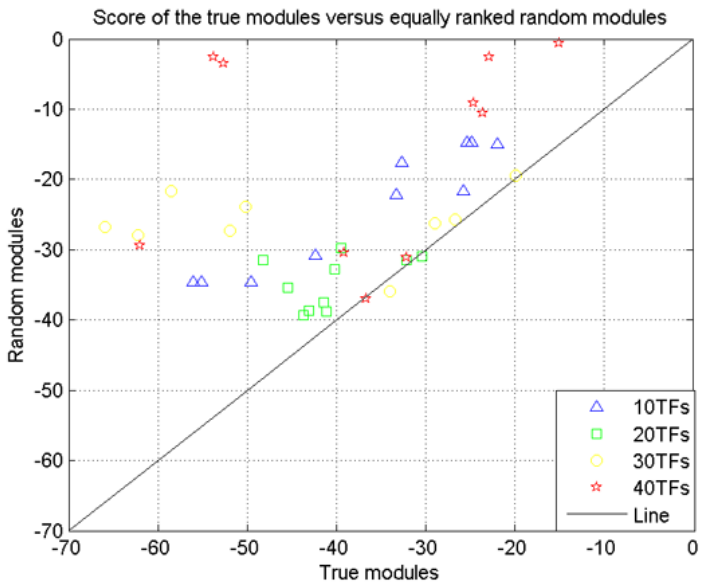


Figure 3.2: **Modules scores of the true modules versus equally ranked random modules**. For each true module its score (log value, the lowest one is the best) is plotted versus the score (log value) of the equally ranked module in the randomized dataset. Different symbols correspond to the different datasets of increasing complexity (using respectively 10, 20, 30 and 40 TFs as input).

the diagonal of the first quadrant (Figure 3.3a). When plotting the modules found in the non-randomized dataset against the baseline, it is clear that the highest ranked modules have a score which is consistently better than the score of the equally ranked modules in the random set, reflecting the true signals in the real data set (Figure 3.3b).

3.3.4 Running parameters for other tools

We only included module detection methods for which the command line version was available in order to be able to optimize parameter settings for our dataset. ModuleSeacher was obtained from the author (Aerts et al., 2003). For ModuleSearcher we used both the genetic and A* algorithm as optimization strategy. As input we used binding site predictions obtained by screening the intergenic sequences using MotifScanner (Coessens et al., 2003) with prior set as 0.2, and a 3th order background model (Thijs et al., 2002). For all parameter settings we

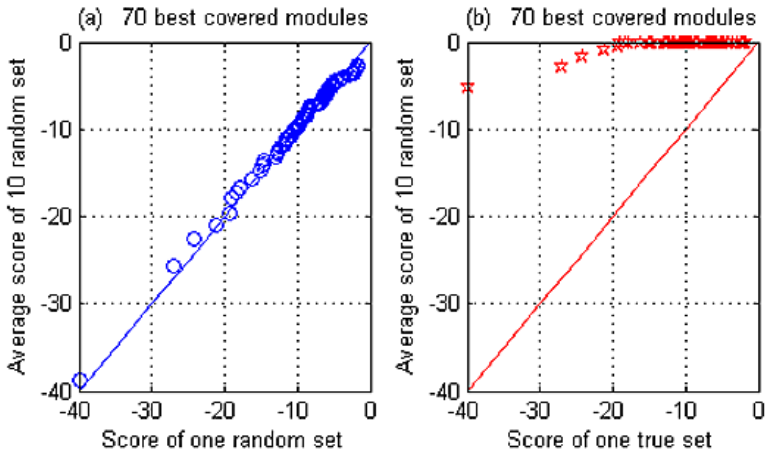


Figure 3.3: **Modules scores of the true modules versus equally ranked random modules**. Plot of the scores of the 70 best ranked modules versus a baseline for a) a random set and b) the true sets. The baseline consists of the average scores of the 70 best ranked modules in 10 different randomizations. For the "true sets" we used as input a set of 10 TFs amongst which OCT4, SOX2 and NANOG were present, while the random sets use as an input a set of TFs without the OCT4, SOX2 and NANOG. The random sets are thus not expected to contain any true modules. Panel a) all selected modules are random and reflect scores of false positives. They are distributed on the diagonal of the first quadrant. Panel b) the scores of the highly ranked modules in the true dataset score consistently higher than the equally ranked modules in the random sets.

used default settings except for the maximum number of motifs and for the length of the region within which a module should be contained. The maximum number of motifs was set to two or three motifs. For the length of the promoter region that should contain the module we used both 200 bp (default value) and 1000 bp. Clover was downloaded from the website of the paper. The input consisted of the intergenic sequences, the PWMs, and the human background model (default background model as suggested by the author). The p-value threshold for a motif to be called significant was set to 0.05. We compiled potential modules from the output of Clover by making all combinations of at least two TFs from the TF that were found significantly enriched in our benchmark dataset. Then we checked whether the true modules were among the collection of potential modules. For CisModule the number of motifs to search for was set to one, two or three and the length of the module was set to 200 bp and 1000 bp separately, we ran it for 1000 iterations, and set the motif length from 12 bp to 15 bp.

3.3.5 Comparing with other tools

For comparison with the other methods, we used for all methods the 116 intergenic sequences. Most of the previously developed module detection tools require a data reduction prior to their usage. This data reduction is usually based on preselecting regions conserved between species and the TFBS located within them. In most of the described analyses this results in sets of 500 bp in length per sequence and an input of about 20 TFs that can make up the module. For our benchmarking, we mimicked data reduction by only including for each gene the proximal promoter region and by providing a restricted number of TFs of which we knew they were present amongst the experimentally verified modules. In the comparative analysis we started with a first analysis containing four TFs (three experimentally verified ones and one sampled randomly from 584 TFs present in TRANSFAC). Each test was repeated ten times, each time randomly sampling another TF. We repeated all analysis using ten TFs as input. We assessed the results by calculating the sum of the recovery rate (SRR), the number of modules in the output (NM) and the sensitivity. The recovery rate (RR) equals one if the true module was among the results of a specific test. The SRR thus corresponds to the number of tests that contain the true modules. If the module was recovered (i.e. if RR=1), we also computed the NM, or the number of modules that were identified (for Modulesearcher this equals the total number of modules in the output which contain at least two motifs; for Clover this equals the total number of combinations one can make with TFs called significant; for ModuleDigger this equals the number of modules ranked higher than the true module). The sensitivity is defined as the number of genes out of the 116 in which the module was detected. All values reported are averages over the 10 test runs.

3.4 Conclusion

Our itemset mining methodology detects CRMs by taking as input a set of genes, assuming that at least a subset of these are coregulated, and searches for a recurrent pattern of TFBS. Our method differs from previous approaches in that it first enumerates all the possible combinations of the TFBSs in the input, and subsequently applies an iterative ranking step, statistically overrepresented in the complete set of input genes, and not overlapping with higher ranked CRMs, are prioritized by assigning them an overall module specificity score. The advantage of using an itemset mining approach instead of an optimization-based strategy is clear from the comparison with other module detection methods. First, the algorithm is much faster and can easily be applied to larger datasets (containing more genes and more TFs). Secondly, because it exhaustively explores all solutions it does not risk to get stuck in a local optimum.

Chapter 4

CPModule: Unveiling combinatorial regulation in mouse embryonic stem cell

4.1 Introduction

In eukaryotes transcriptional regulation is mediated by the concerted action of different transcription factors (TFs) (Davidson, 2001). Searching for *cis*-acting regulatory modules (CRM) or combinations of motifs that often co-occur in a set of coregulated sequences helps in unraveling the mode of combinatorial regulation. CRM detection is traditionally being applied on a set of intergenic regions located upstream of coregulated genes identified by e.g. microarray experiments (here referred to a gene centered methods). However, as in higher eukaryotes regulatory intergenic regions can be prohibitively large for successful CRM detection, the search for combinatorial regulation is often limited to the proximal promoter region. However, biologically relevant CRMs in eukaryotes can occur in enhancers or binding regions distantly located from the proximal promoter, reducing the success of gene centered CRM detection. Nowadays with chromatin-immunoprecipitation (ChIP) based techniques becoming increasingly popular for the genomewide identification of TF binding sites, it is feasible to locate at least for an assayed TF the approximate binding regions on its target genes. Such datasets therefore offer a new interesting application for CRM detection by allowing computationally predicting with which other TFs the ChIP-assayed TF potentially interacts. ChIP information thus not only allows reducing largely the regions in which the motif sites of the assayed TF should be located (typically 500 bp instead of thousands of bp), but also limits

the number of potentially relevant CRM solutions to those that at least contain a motif for the assayed TF (here referred to as a query-based search).

So far many different methods have been developed for CRM detection. Except for the de novo methods (Zhou & Wong, 2004; Xie et al., 2008), most CRM detection method rely on a motif screening step, in which all potential sites of TFs with known motifs are located in the intergenic sequences of interest. Subsequently a combinatorial search is performed to identify sets of binding sites that occur frequently in the input set. Different CRM detection methods have been developed that differ from each other in the way they tackle the combinatorial search problem. Methods such as, for instance, ModuleSearcher (Aerts et al., 2004) and ModuleMiner (Van Loo et al., 2008) pose the CRM problem as an optimization problem (e.g. uses a genetic algorithm) with an explicit cost function to be optimized, while Compo (Sandve et al., 2008) and ModuleDigger (Sun et al., 2009) rely on itemset mining to first enumerate all possible module combinations after which a statistical filtering strategy is applied to identify the most promising CRMs. Methods also differ in the way they define a module either in the cost function or during the enumeration (for itemset mining approaches). In all methods a CRM is defined as a set of motifs. However depending on the method the description can be more accurate such as e.g. the motifs should occur together within a predefined distance or the spacing between the motifs sites contributing to the CRMs should be of fixed size. A major distinction can be made between CRM methods that are based on the assumption that a set of coregulated genes should share a common CRM versus those that treat each sequence independently (further referred to as the single-sequence based methods). Cister (Frithet al., 2001) or ClusterBuster (Frithet al., 2003) are examples of the latter category: these methods search in a single sequence for potential CRMs that best match a predefined structure as imposed by the model parameters (here a hidden Markov model) using as input the probabilities of each segment matching individual motif models. Methods that do exploit the dependency between the sequences in an input set, in contrast assign a higher weight to CRMs that occur frequently and of which this frequency of occurrence is not likely given the background nucleotide distribution. While traditionally methods identify a CRM as a set of motifs that co-occur more frequently than expected based on the nucleotide background composition of the organism of interest, the more recently developed methods (CREME (Sharan et al., 2003;), ModuleMiner and ModuleDigger) also assess the specificity of the CRM for the set of input sequences i.e. they compare to what extent a similar CRM occurs in a large set of sequences randomly sampled from the genome using respectively a hypergeometric, a rank-based or a adopted binomial statistic strategy.

From all methods mentioned above those that are developed to work on sets of coregulated genes are most suitable for the detection of CRMs in ChIP identified binding sequences. However, because of the combinatorial complexity of CRM detection all previously mentioned methods are restricted in the sizes of the input

set they can handle: they are usually applied on a few input sequences only (of maximally a few 1000 bp) and allow searching for a combination of a few TFs only (preferentially using a prescreening with models of TFs that are involved in the process of interest). However, as the goal is to identify with which other TFs the assayed TF interacts, it is very hard to define in advance with which TF models the prescreening needs to be performed and ideally one wants to include all possible TFs in the analysis. In addition sequence sets obtained from ChIP-information are typically large i.e. a few 100 of sequence regions that correspond to the best scoring binding peaks for which there is no guarantee that all those genes should be coregulated by the same CRM. Applying CRM detection to ChIP-defined sequence regions therefore is still not trivial.

Therefore, we propose in this study an analysis flow that allows performing CRM detection on ChIP-defined regions by combining a powerful combinatorial search algorithm with a strategy to reduce the search space in a biologically motivated way. The latter is done by constraining the number of possible motif sites during the screening step and the number of valid motif combinations during the combinatorial search. We demonstrate using synthetic data that our CRM detection method has a performance comparable to that of state-of-the-art CRM techniques and show on real ChIP-based experiments conducted by Chen et al. 2008 for five key TFs involved in self-renewal of mouse embryonic stem cells how our CRM detection analysis flow can be used to predict combinatorial regulation of the assayed TF with other TFs with a documented PWM in TRANSFAC.

4.2 Results

4.2.1 Analysis flow

The complete analysis flow for CRM detection is depicted in Figure 4.1 and consists of three major steps: in a first step input sequences corresponding to ChIP-bound regions are scored with all PWMs (position weight matrices) of TRANSFAC using Clover (Frith et al., 2004). In our study we combined standard PWM screening with a filtering strategy based on epigenetic features. This filtering is meant to improve the trade off during the screening step between obtaining a high sensitivity in recovering as many true sites as possible while reducing the number of false positive binding sites. For the second step, the actual combinatorial search, we rely on a constrained-based itemset mining framework. For the third step, the calculation of a genomewide enrichment score was performed for each found CRM. All steps are elaborated below.

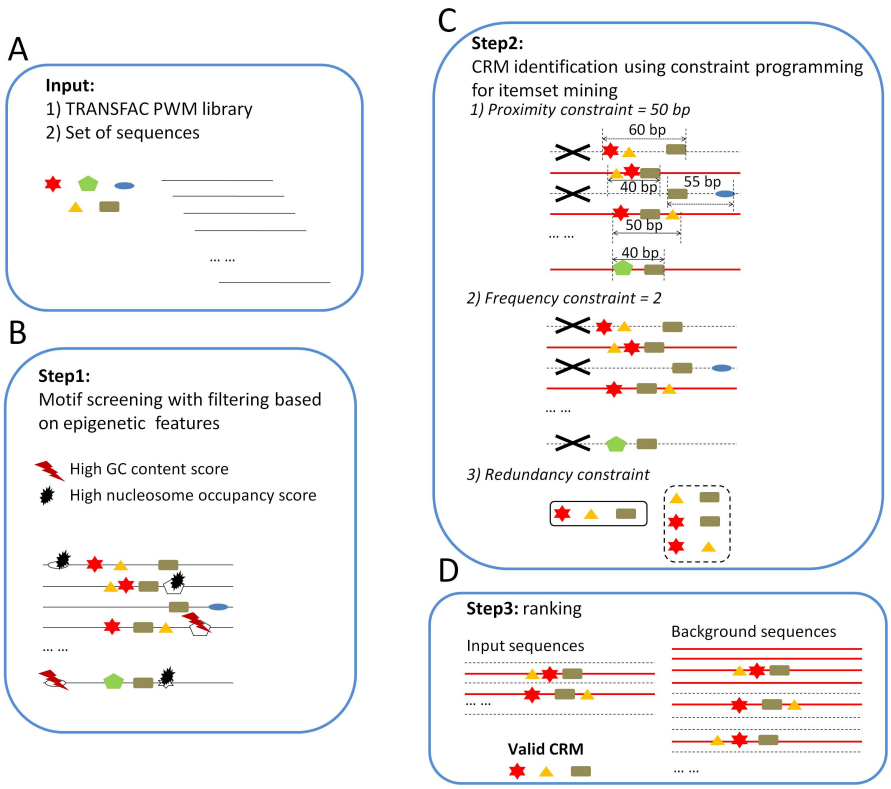


Figure 4.1: **Analysis flow**. Panel A: The input consists of a library of PWMs and a set of sequences. Panel B: In a first step, prior to the actual CRM detection a screening with public motif databases is performed. Here we combine standard PWM screening with filtering based on epigenetic features. Only regions containing a motif site that display a low GC content and a low nucleosome occupancy are withheld, the motif sites display a high GC content and a high nucleosome occupancy will be filtered out (indicated as the transparent shapes in Panel B). Panel C: The second step consists of the actual combinatorial search. Here we use a constrained-based itemset mining approach to enumerate all valid CRMs i.e. combinations of motifs 1) of which the motif sites contributing to the CRM occur in each others proximity (user defined) 2) that occur frequent in the input set (i.e. in all sequences displayed in red), 3) that are non-redundant. Panel D: Valid CRMs are finally ranked based on their specificity for the background sequences.

4.2.2 Motif screening with filtering based on epigenetic signals

Regarding the threshold on the GC content, we choose 50% as for this threshold we obtain on average 2.5 sites per screened TF per sequence (Figure 4.2 Panel B) with a reduction in sensitivity of the assayed TF to minimally 50% (For TF KLF4) (Figure 4.2 Panel A). In addition 50% to make a distinction between GC rich and GC poor regions corresponds to the definition of CpG island which are often associated with DNA methylated regions (24,25). For NuOS Figure 4.2 shows that the most significant filtering occurs after lowering the threshold from 1 to 0.9 (Figure 4.2 Panel D). It is at this transition that also the most significant drop in recovering the sites of the assayed TFs (i.e. sensitivity) occurs (Figure 4.2 Panel C). Further reducing NuOS thresholds reduces the average number of sites per screened TF from 1.7 to 1.2, resulting in an additional removal of about 25850 (0.5sites x 100seq x 517pwm) sites without compromising too much the sensitivity (Figure 4.2 Panel C). That is why eventually we choose 0.1 as a filtering threshold for NuOS. In the following, when filtering was performed on the screening results obtained

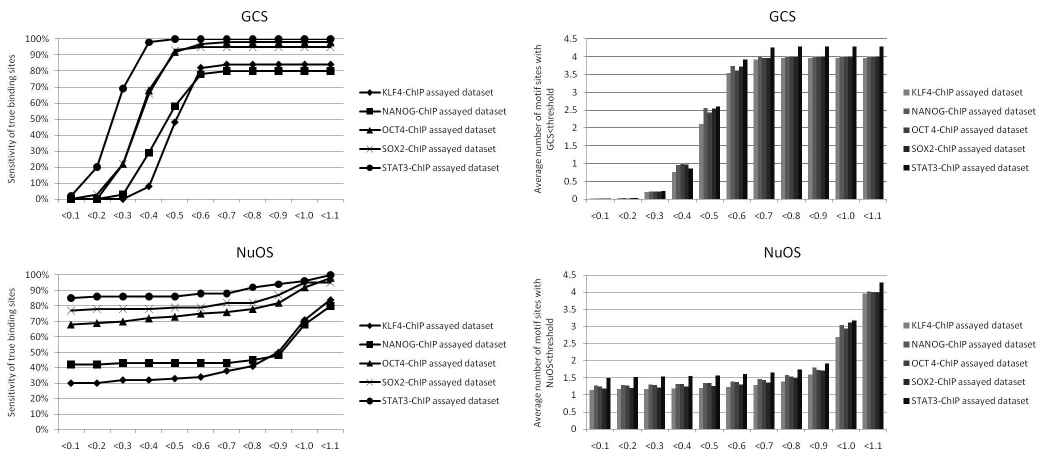


Figure 4.2: **Effect of the filtering thresholds on motif prediction results**. Effect of the filtering thresholds on detecting true sites (sensitivity): The percentage of binding peak regions in which a motif site of the assayed TF could be detected after filtering non-stringent screening results with respectively different values of GC content scores (Panel A) and nucleosome occupancy scores (Panel C). Effect of the filtering thresholds on the number of false positive detection rate: as estimated by the average number of remaining motif sites per sequence and per TF for each of the ChIP-assayed datasets after filtering the low stringent screening results with respectively different values of GC content scores (Panel B) and nucleosome occupancy scores (Panel D).

for one particular TF, only predicted motif sites with a low GC content and low nucleosome occupancy (lower than 0.1) were retained for further analysis (Figure 4.1: step 1). Next we illustrate how the combination of both filtering procedures with different screening thresholds affects the trade off between sensitivity and false discovery rate of the detected motifs. The sensitivity of the screening/filtering procedure was again estimated using the ChIP-Seq data of Chen et al. 2008 as golden standard while the false discovery rate was assessed as the average number of motif sites predicted per sequence and per screened TF. Figure 4.3 Panel A shows the sensitivity of retrieving the true binding sites for each of the assayed TFs after applying different combinations of screening/filtering. The sensitivity is expressed as the percentage of binding peak regions for each of the assayed TFs in which a corresponding motif site could be detected. As can be expected, a stringent screening results in a rather low sensitivity for most of the binding sites of the assayed TFs (sensitivity less than 50%). Lowering the screening stringency largely increases this sensitivity (at least 80% of the binding peaks for respectively KLF4 (84%), NANOG (80%), OCT4 (98%), SOX2 (95%) and STAT3 (100%) were found to contain a binding site for their respective TFs. This increased sensitivity comes at the expense of also predicting potentially many more potentially false positive sites as on average a low stringency screening as applied here results per sequence in 4 motif sites per TF (Figure 4.3 Panel B). Compared to a stringent screening, combining the non-stringent screening with epigenetic filtering largely increases the sensitivity for most of the assayed TFs while maintaining the number of predicted sites per TF we screened within a reasonable range (1 per screened TF and per sequence which still results in on average about 517 sites per sequence (double strand)). Using a too stringent screening threshold will result in a sensitivity that is too low for successful CRM detection (see below).

Previous studies showed, the GC-content also has effect on nucleosome positioning (Dhami et al., 2010; Andersson et al., 2009). Interestingly, this has been observed from Figure 4.2 and Figure 4.3 Panel A, which shows that the effect of using filtering based on both GC content and nucleosome occupancy can be complementary and that the degree of complementarity depends on the particular TF. For instance, for KLF4, the sensitivity in retrieving sites in true KLF4 binding peaks by applying only GC content (Figure 4.2 Panel A) is 48%. Well the sensitivity in retrieving sites in true KLF4 binding peaks by applying only the nucleosome screening (Figure 4.2 Panel C) is 30%. When using a combined screening of nucleosome occupancy and GC content (Figure 4.3 Panel A of the grey bar), the sensitivity is 15%, indicating that there are 73% of the kept sites (kept by either type of filtering) are overlapped. Similarly, NANOG has 74% overlap of kept sites; NANOG has 96% overlap of kept sites; SOX2 has 99% overlap of the kept sites; STAT3 has 100% overlap of the kept sites. Thus based on the results from Figure 4.2 and Figure 4.3 Panel A, actually the GC content filtering plays a minor role in the epigenetic filtering comparing with the nucleosome occupancy filtering, in other words, it's by utilizing nucleosome occupancy filtering, we reduced most of potential binding sites.

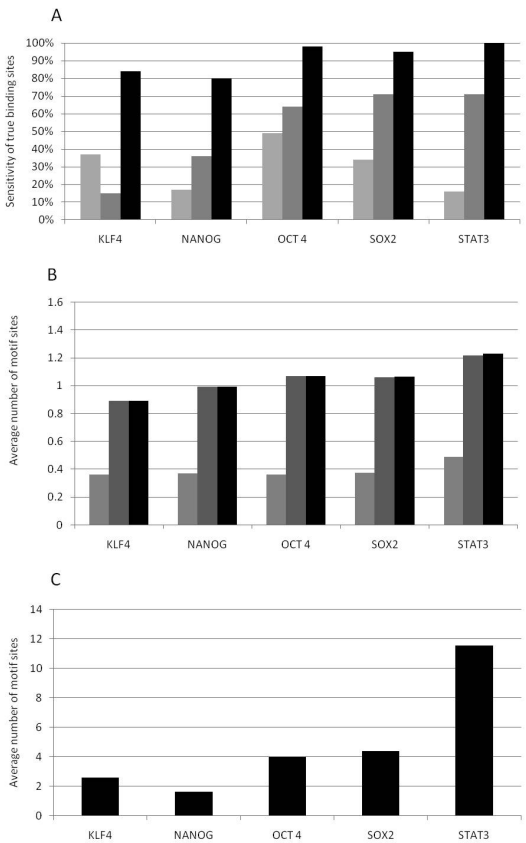


Figure 4.3: **Effect of combining epigenetic filtering with different screening thresholds on motif prediction results**. Panel A: Effect on the sensitivity of the assayed TF of respectively a stringent screening (motif screening threshold of 6.0), indicated by the light grey bar; a non-stringent screening with filtering for each TF (threshold of 3.0), indicated by the dark grey bar; and a non-stringent screening without filtering (threshold of 3.0), indicated by the black bar. Sensitivity is assessed by the percentage of binding peak regions in which a motif site of the assayed TF could be detected. Panel B: Effect on the average number of remaining motif sites per sequence and per TF for each of the ChIP-assayed datasets, for respectively a stringent screening (motif screening threshold of 6.0), indicated by the light grey bar; a non-stringent screening with filtering for each TF (threshold of 3.0), indicated by the dark grey bar; and a non-stringent screening without filtering (threshold of 3.0), indicated by the black bar.

4.2.3 CPModule: CRM detection based on constraint programming for itemset mining

For CRM detection we adopted a strategy based on constraint programming for itemset mining (De Raedt et al., 2008). This approach combines the advantages of itemset mining in being able to solve combinatorially hard problems with the flexibility of easily imposing constraints that allow reducing the complexity of the search space. CPModule enumerates all possible CRMs that meet the following biologically motivated constraints (Figure 4.1 panel C): a certain CRM should occur in a minimal number of sequences (frequency constraint (support)) and its composing motifs should occur within a maximal genomic distance from each other (proximity constraint). The first constraint allows tuning the degree of overrepresentation that we expect in a set of intergenic sequences, while the second constraint reflects that sites of combinatorially acting TFs occur in each others neighborhood. A last constraint (redundancy constraint) reduces the level of redundancy amongst the valid CRMs: if a CRM occurring in a set of sequences is completely contained within a larger CRM that occurs in exactly the same sequence set, only the larger CRM is withheld. After enumerating all possible, valid non-redundant CRMs, a genomewide enrichment score is assigned to each of them based on their expected occurrence in a set of background sequences. Predicted CRMs are ranked based on this enrichment score.

To test the performance of CPModule we applied it on the synthetic data constructed by Xie et al. 2008. This dataset contains intergenic sequences in which 'true motif sites' are inserted. As these true sites closely resemble the TRANSFAC PWMs from which they were sampled), that can easily be picked up by a first screening step using a stringent threshold. As this is a synthetic dataset, we did not apply the epigenetic filtering step. To evaluate CPModule we compared it against a number of well performing CRM tools, namely Cister, Cluster-Buster, ModuleSearcher and Compo. The performance of CRM detection was assessed by comparing the best scoring solution of each algorithm with the true solution, by using respectively the motif correlation coefficient (mCC) and the nucleotide correlation coefficient (nCC) (see materials and methods). The first score assesses to what extent a predicted CRM is composed of the true motifs while the second one also assesses to what extent a true motif was recovered by predicting the correct sites.

Table 4.1 shows for each of the algorithms their motif and nucleotide level correlation coefficients (CC), when using as input the synthetic data prescreened with the 516 TRANSFAC PWMs (see materials and methods). Table 4.1 shows that both Cister and Cluster-Buster were able to find a solution, albeit of mediocre quality. Because they operate on each sequence individually, a rather large number of different motif predictions per sequence are obtained, most of which are false positives. ModuleSearcher ran into memory problems, even when being allocated

2GB of RAM. Compo which also uses a strategy based on itemset mining was still running after 2 days without any results, probably because it doesn't address the redundancy problem, which is inevitable when dealing with large numbers of motifs. To better compare the algorithms we ran them with an increasing number of PWMs.

	Cister	Cluster-Buster	ModuleSearcher	Compo	CPModule
mCC	0.16	0.05	/	-	0.57
nCC	0.23	0.23	/	-	0.55

Table 4.1: Motif and nucleotide level correlation coefficients (CC) for different algorithms obtained using all 516 TRANSFAC PWMs. "/" indicates termination by lack of memory, "-" indicates that the algorithm was still running after 2 days.

We create different PWM sets by starting from the 3 PWMs of the inserted TFs, and sampling a number of additional PWMs from the set of 513 remaining PWMs. We independently sampled 10 sets for every sample size. Figure 4.4 shows the motif and nucleotide level correlation coefficients (CC) of the different algorithms when run with an increasing number of sampled PWMs. The single-sequence based tools Cister and ClusterBuster perform rather poor at motif level, but outperform their competitors at nucleotide level, at least when given a dataset prescreened with a few PWMs only (low noise scenario). This confirms what was also previously observed i.e. a single sequence approaches perform well when searching for CRMs that are composed of a few motifs only. In addition it shows that their performance is quite sensitive to the presence of noisy sites in the input. In the presence of a high noise level, CMR methods that exploit the dependency between sequences become more competitive as they use the frequency of occurrence of predicted CRMs to constrain the search space. Amongst the methods that exploit multiple sequence information, CPModule often performs best, closely followed by ModuleSearcher. This better performance is most pronounced at the nucleotide level. In our hands Compo exhibited a constant behavior independent of the noise level, but clearly underperformed compared to the other tools, mainly at the nucleotide level. We observed that independent of the PWM set used to perform the prescreening, it only returned CRMs being composed of one of the inserted motifs only. For this reason, we suspect that the problem was not that much in the CRM detection but in its build in screening method that was set too stringent. However, in our hands, changing the parameters of the algorithm did not change this behavior.

These results show that CPModule is definitely competitive with other CRM tools in effectively searching CRMs in large sequence sets, even in the presence of many motif sites that do not contribute to the actual CRM (i.e. when screening with complete TRANSFAC (16)). Because we aim at using the proximity constraint to reduce the search space we also assess the sensitivity of our method towards this constraint. Figure 4.5 shows the quality of the results obtained with CPModule on the same dataset using different values of the proximity constraint as measured by

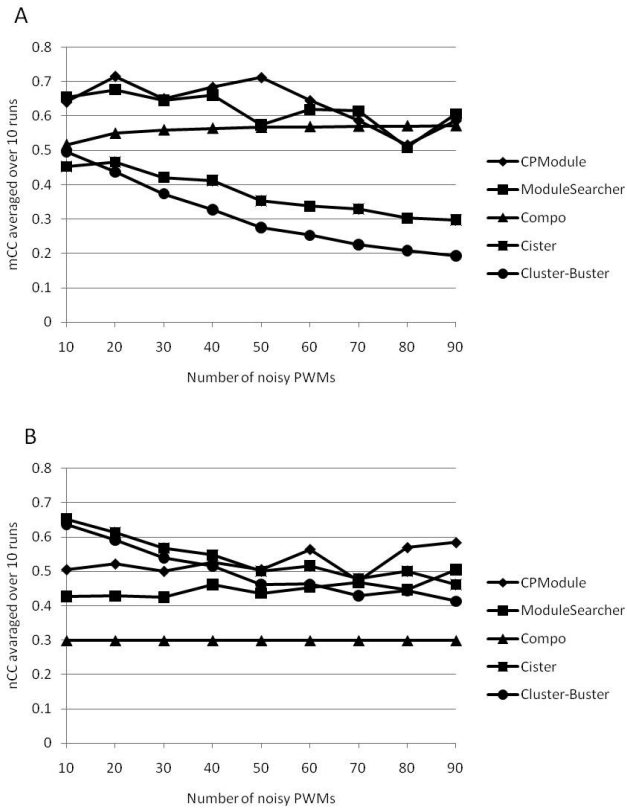


Figure 4.4: **Effect of noise on the performance of CRM detection algorithms**. All CRM detection algorithms were run on the synthetic dataset of Xie et al. 2008. Prescreening was performed with the PWMs used to generate the synthetic data in combination with an additional of PWMs sampled from TRANSFAC (the number of noisy PWMs added to the true PWMs is indicated on the X-axis). Panel A: mCC motif correlation coefficient, Panel B: nCC nucleotide correlation coefficient.

the motif and nucleotide level correlation coefficient (CC). In the dataset, sites of motifs composing the CRMs in each of the sequences were simulated to be located within a distance of 164 bp from each other. The proximity constraint defines the maximal region on the DNA within which motifs contributing to the CRM can be located: setting a too strict proximity constraint results in low nucleotide level correlation coefficients, and even lower motif level correlation coefficients. Probably the number of motifs contributing to the CRM is underestimated as some

contributing motifs might be located in a region outside the allowed proximity constraint. At a distance within which most of the inserted CRMs occurred (164 bp), our method achieves its highest motif and nucleotide level correlation coefficients. In addition, within a reasonable range around this optimal setting (i.e. between 140 and 200 bp) the method is not too sensitive towards the exact choice of the proximity constraint (with again the nucleotide level score being more sensitive than the motif level score towards the effect of missing motif sites when underestimating the proximity constraint and towards false positive sites when overestimating the proximity constraint). For completeness we compared the

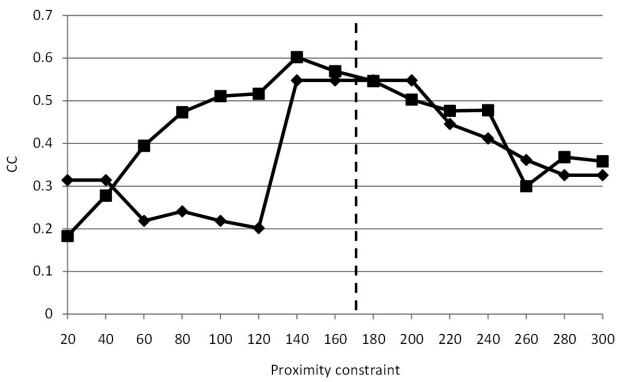


Figure 4.5: **Effect of the proximity constraint on the quality of results**. Motif correlation coefficient (the black diamonds) and nucleotide correlation coefficient (the black squares) of the results obtained by applying CPModule on the synthetic data for different proximity constraint values. The true proximity as applied when generating the synthetic dataset is 164 bp.

results of CPModule with that of other related well performing CRM tools, namely Cister, Cluster-Buster, ModuleSearcher and Compo. Table 4.1 shows for each of the algorithms their motif and nucleotide level correlation coefficients (CC) (see materials and methods), when using as input the synthetic data prescreened with the 516 non-redundant TRANSFAC PWMs (see materials and methods).

4.2.4 Detecting CRMs involved in mouse embryonic stem cell

For the analysis on real data we relied on the publicly available ChIP-Seq experiments conducted by Chen et al. 2008. The data consists of ChIP-Seq experiments for five key TFs involved in self-renewal of mouse embryonic stem cells, namely KLF4, NANOG, OCT4, SOX2 and STAT3. In this analysis we demonstrate the potential of our proposed flow: starting from the data of a single ChIP-Seq

experiment, we will use CRM detection to discover, in silico, the other TFs with which the assayed one constitute a regulatory module. As we know from Chen et al. 2008 and literature that combinatorial interactions exist amongst at least some of these five TFs (Figure 4.6), we used the binding peaks of the TFs other than the one for which we use the data as input to verify our CRM predictions (cross validation set up).

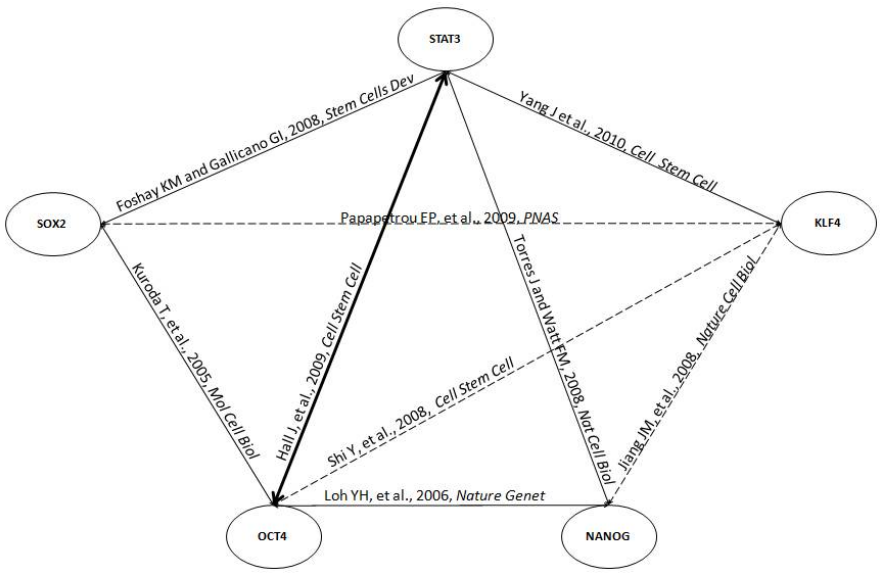


Figure 4.6: **Known combinatorial regulation of the five assayed TFs**. Network representing combinatorial interactions between the five transcription factors (KLF4, SOX2, OCT4, NANOG and STAT3) that are involved in embryonic stem cell development. Edges indicate that a combinatorial interaction between the indicated TFs exists as reported in literature (with a combinatorial interaction referring to the fact that at least subsets of genes contain binding sites for both TFs in each others neighbourhood). The thin black lines indicated detected interactions using one ChIP-Seq data from these two interacted TFs, and the thick black line indicated the detected interaction between these two TFs using respectively their corresponding ChIP-Seq data. The dash thin black line indicated the undetected interactions in our analysis.

We started from the top 100 binding peaks identified for one ChIP-Seq-assayed TF (as we assume that those represent the most reliable binding sites). As it was recently shown that the sites of the assayed TF do not exactly coincide with their binding peaks, but can be located as far as 250 bp from the actual peak, we will use a sequence region of 500 bp centered around each binding peak as input

sequences. The sequences surrounding these peaks were screened with a large number of position weight matrices (PWMs) from TRANSFAC (16). To recover as many as possible binding sites for the assayed TF, while not too drastically increasing the total number of detected sites, we applied a non-stringent screening threshold in combination with filtering for all binding sites except for the ones corresponding to the ChIP-Seq-assayed TFs (see above and Figure 4.2 Panel A), following the reasoning that for TFs where we have experimental evidence of binding, this evidence overrules the results of the filtering. This screening/filtering combination results in a recovery rate of respectively 84%, 80%, 98%, 95% and 100% of the peak regions of KLF4, NANOG, OCT4, SOX2 and STAT3 while keeping the total number of predicted sites for the other TFs within a reasonable range (with about 2, 2, 4, 4, and 11 sites per peak region for respectively KLF4, SOX2, OCT4, NANOG and STAT3 (Figure 4.3 Panel C) and on average 1 site for every other TF (Figure 4.3 Panel B)).

Subsequently we used combinatorial module detection to search for combinations of motifs that occur frequently amongst the 100 best scoring peak regions for each of the TFs. To further limit the search space we used the proximity constraint to restrict the maximal region in which we consider binding sites to co-occur. As it is difficult to know in advance the best value for the proximity constraint, we started from 150 bp and step wisely (50 bp) extended this value until we detect the first valid CRM that contains the motif for the ChIP-Seq-assayed TF (with a maximal bound of 400 bp).

As such we search preferentially for CRMs being composed of the ChIP-assayed TF (query-based mode) together with other TFs, co-occurring in the smallest possible distance (most stringent proximity constraint), as we assume the latter ones are the most likely candidates to reflect true combinatorial regulation. For each potential CRM its probability to occur with the same frequency in a random set of 5000 background sequences was used to rank the different predicted CRMs. We also apply the epigenetic filtering on the background sequences, and the background sequences have the same properties as the input sequences, i.e. each of the background sequences at least contain one TFBS for the assayed TF.

To validate our results, we tested whether our method recovered any of the previously described benchmark CRMs involving respectively KLF4, NANOG, OCT4, SOX2 and STAT3 (see Figure 4.6). If recovered these CRMs are displayed in Table 4.2 together with their respective support and rank. The support indicates how many of the 100 peak regions contained the module. The larger this value the more sequences in the input set will be found coregulated by this module. CRM detection using constrained-based itemset mining allows enumerating all possible CRMs that meet the preset constraints and scoring them based on their statistical significance. This allows viewing the rank of a true solution amongst all total solutions. The rank displayed in Table 4.2 thus indicates whether a module could be detected as a top scoring one or not. Both its rank relative to the total number

of valid CRMs and relative to the valid CRMs containing the ChIP-Seq-assayed TF are displayed (the latter one is referred to as the rank in the query-based setting). As an additional cross validation we tested for each assayed TF for which we could predict a true previously documented CRM, to what extent the sites contributing to motifs in that CRM fell within identified binding peak regions of their corresponding TFs (with a peak region being defined as a region of 250 bp around the peak center). For instance, when starting from the ChIP-seq data of SOX2 we predicted a previously described CRM containing SOX2-OCT4. This retrieved module was ranked first amongst the 22 potential CRMs that contained OCT4. OCT4 and SOX2 co-occurred in 63% of the SOX2 ChIP-Seq identified regions (support) within a distance of 150 bp and the identified sites for OCT4 fell within the identified OCT4 ChIP-Seq regions in 79% of the cases. Although Table 4.2 displays for each assayed TF the results with smallest possible proximity constraint, we also tested the effect of further increasing the proximity constraint in retrieving benchmark CRMs. Within some boundaries further extending the bound in which the motifs of a module could co-occur resulted in the retrieving the same benchmark CRMs albeit at a lower rank as indeed increasing the proximity constraint increases the search space and the number of valid CRMs. We, however, never detected a benchmark module at a higher proximity constraint that was not found with the lowest possible proximity constraint. Table 4.2 thus shows which of the previously described CRMs involved in self-renewal could be recovered by CPModule on the input ChIP-Seq data together with an estimate of their quality. These results show that CPModule is able to retrieve 6 out of 9 of the previously described CRMs involving KLF4, NANOG, OCT4, SOX2 and STAT3 (Figure 4.6). On overall the quality of these CRMs is considerable: the CRMs were detected in at least 60% of the sequences and in the cross validation set up the reported CRMs were validated in at least 10% of the cases by the ChIP-Seq data of the cognate validation sets. However, these true CRMs were not always the best ranked solutions. Mainly the CRMs involving STAT3 sites rank poorly. This is probably due to the low specificity of the screening results obtained with the STAT family of TFs: after screening and epigenetic filtering we still obtain on average 11 STAT sites per sequence (data not shown), indicating that even after filtering STAT3 sites are frequently occurring in the genome. Such high genomic frequency deteriorates the specificity of CRMs containing STAT3 sites for the set of input sequences and decreases their rank. Without ChIP-Seq data these CRMs would never be considered. When comparing the rank of the obtained solutions relative to the total number of all possible CRMs with their rank in a query-based setting (only taking into account the CRMs containing the ChIP-Seq-assayed TF), the added value of performing CRM detection in a query-based setting using ChIP-Seq derived information becomes clear. Using chip-derived information to delineate an approximate binding region for at least one TF, increases the absolute rank of the true CRMs considerably compared to using standard gene centered CRM detection.

ChIP -Seq- assayed TF	CRM	Rank	Support	Cross Validation	Proximity Constraint	Number of solutions
KLF4	KLF4,STAT4	143/2	60%	40.00%	300	3/147
NANOG	NANOG,OCT1	6846/4	61%	70.49%	300	17/6868
	NANOG,STAT3	14017/10	60%	25.00%	350	26/14033
OCT4	OCT4,STAT1[XFD2,STAT4,STAT6]	5/5	63%	11.10%	150	613/5068
SOX2	SOX2,OCT4	430/1	63%	79.40%	150	22/14180
	SOX2,STAT3[CDXA,PAX2,STAT5A]	61807/24	60%	23.33%	250	189/117006
STAT3	STAT3,OCT4[STAT1,STAT6,STAT5A]	1/1	61%	24.59%	150	20/1366

Table 4.2: **CRMs obtained with CPModule in combination with epigenetic filtering (non-stringent screening with filtering for all TFs except the assayed one).** The set of sequences corresponding to the 100 top scoring peak region of the assayed TF were screened with a set of 516 non-redundant TRANSFAC motifs using a non-stringent screening threshold. Epigenetic filtering was applied on all motif sites except on the ones of the assayed TF. **ChIP-Seq-assayed TF**: TF from which the top 100 binding peaks were used to perform the analysis. CRM: obtained CRMs that correspond to previously well described modules for the assayed TF; [between brackets are indicated other TFs that were predicted to belong to the same CRM, but that have not previously been described to interact with the assayed TF]. **Rank**: rank this CRM received in all of the solutions/the rank this CRM received in the query-based setting. **Support**: the percentage of sequences from the input set in which this CRM occurs (should be higher than the frequency constraint). **Cross validation**: we started from the ChIP-Seq data of one TF and tried to predict using CRM detection with which other TFs the assayed TF interacts. We verified whether the motif sites contributing to the predicted CRMs fell within the binding peaks of the other ChIP-Seq-assayed TFs. **Proximity constraint (bp)**: the proximity constraint at which the displayed CRM was found. **Total number of solutions**: the total number of valid CRMs/the number of solutions containing the motif for the ChIP-Seq-assayed TF.

Besides the results on the benchmark set, we also displayed for all assayed TFs their top 3 ranking CRMs (Table 4.5). Note that those CRMs score better than the benchmark CRMs and based on literature evidence some might be true interactions involved ESC biology.

Several CRMs involving NANOG, OCT4, KLF4, STAT3 or SOX2 were predicted by CPModule based on the ChIP-Seq data and 517 Transfac PWMs. Together, these consisted of 19 additional TFs (Table S1). Functional analysis of 19 TFs was performed by Ingenuity Pathway analysis and significant functions in which at least 10 of these TFs are involved are: gene expression, cellular growth and proliferation, cell death, cancer, cellular development and tissue morphology. Besides gene expression (which is a logical finding, since this is a general property of TFs), we thus find all functions related to cell ESC growth, death and differentiation. For a handful of the CRMs, literature evidence of binding of the TFs supports the

predictions (See literature support part in the following sections). Additionally, further literature analysis clearly points towards a role for most of the retrieved TFs in embryonic stem cell biology (See literature support part in the following sections).

Several CRMs involving NANOG, OCT4, KLF4, STAT3 or SOX2 were predicted by CPModule based on the ChIP-Seq data and 517 Transfac PWMs. Together, these consisted of 19 additional TFs (Table S1). Functional analysis of 20 TFs was performed by Ingenuity Pathway analysis and significant functions in which at least 10 of these TFs are involved are: gene expression, cellular growth and proliferation, cell death, cancer, cellular development and tissue morphology. Besides gene expression (which is a logical finding, since this is a general property of TFs), we thus find all functions related to cell ESC growth, death and differentiation. For a handful of the CRMs, literature evidence of binding of the TFs supports the predictions (See literature support part in the following sections). Additionally, further literature analysis clearly points towards a role for most of the retrieved TFs in embryonic stem cell biology (See literature support part in the following sections).

With optimized constraints, we are able to retrieve 6 out of 9 CRMs (Figure 4.6) from our benchmark set, of documented CRMs comprising the binding sites of the assayed TFs. By using CPModule, we predicted several novel putative CRMs involved in transcriptional regulation of embryonic stem cells (See literature support part in the following sections). Literature evidence supports a role for the retrieved TFs in embryonic stem cell biology (See literature support part in the following sections).

4.2.5 Effect of the screening procedures on the final modules

As already indicated the effect of the motif screening largely influences the quality of the final CRM detection: this is illustrated by comparing the results for CRM detection using a set up similar as described above, but by using as input motifs lists obtained with different screening threshold-filtering combinations. For instance Table 4.3 shows the results obtained using a stringent screening and no filtering applied. It shows that despite the more stringent screening the overall results deteriorate: less benchmark CRMs are detected and the ones that are found have a much lower rank than with the relaxed screening in combination with filtering displayed in Table 4.2. This is mainly due to the fact that increasing the screening threshold lowers the recovery rate of the binding sites of the assayed TF to such extent (30% at most) that only part of the input sequences can by definition contain a module with the assayed TF. To compensate for this we had to run CPModule with a lower frequency constraint. The frequency constraint was gradually lowered from its maximum until results were obtained. This resulted in the CRMs described

in Table 4.3, except OCT4 obtained with a frequency threshold of 20%, all other obtained with a frequency threshold of 10% instead of 60% applied in Table 4.2. This lower frequency constraint results in many more valid CRMs containing the motif sites for the ChIP-Seq-assayed TF, albeit all with a lower statistical significance than when they would have occurred in more sequences (higher support that can be obtained at a higher frequency constraint). Therefore, increasing the stringency of the screening requires the use of less stringent module criteria during the combinatorial search and thus enlarges the search space instead of obtaining the desired reduction in search space. To illustrate the effect of the low recovery

ChIP-Seq-assayed TF	CRM	Rank	Support	Proximity Constraint	Number of solutions
KLF4	KLF4,STAT3,[SP1]	279/22591	11%	150	24004/377
NANOG	/	/	/	/	25997/-
OCT4	OCT4, STAT6	7/7	10%	150	1056/7
SOX2	SOX2, OCT4	42/33922	10%	150	39219/67
	SOX2, STAT	55/37032	12%	150	39219/67
STAT3	/	/	/	/	37337/-

Table 4.3: **CRMs obtained with CPModule without filtering (using stringent screening).** Legend as in Table 4.2. Based on the recovery rate of the ChIP-Seq-assayed TFs. "/" indicates no CRM containing the motif for the ChIP-Seq-assayed TF was found.

rate of the binding sites of the ChIP-Seq-assayed TF itself after performing the stringent screening on the results, we display in Table 4.4 the results obtained by running CPModule on an input set screened with a stringent threshold for all TFs except the assayed one. For the latter one we applied the same screening strategy as in Table 4.2 (low stringency screening without filtering). These results show that improving the recovery rate of the binding sites of the assayed TF during the screening step helps in increasing the support of at least some of the CRMs. For those CRMs also the rank increased drastically compared to the situation where they could only be detected with a low frequency constraint. Table 4.2 shows that results further improve by using also for the other TFs a non-stringent screening in combination with an epigenetic filtering: more of the benchmark CRMs were recovered at a higher rank and with a higher support (results here were obtained with a frequency constraint of 60%). Allowing for more possible sites comes however at the expense of also increasing the search space and the total number of CRMs that need to be enumerated (indicated by the total number of solutions in Table 4.2). This not only makes it increasingly difficult to find the true CRMs amongst the highest ranked ones, but also makes the problem combinatorially prohibitive. Indeed when using the full list of motifs obtained after a non-stringent screening without any filtering applied, the problem became computationally intractable.

ChIP-Seq-assayed TF	CRM	Rank	Support	Proximity Constraint	Number of solutions
KLF4	/	/	/	/	-/-
NANOG	/	/	/	/	-/-
OCT4	OCT4, STAT	2/2	59%	150	5/5
SOX2	SOX2, SMAD	1/1	52%	14482	13/13
STAT3	/	/	/	/	-/-

Table 4.4: **CRMs obtained with CPModule without filtering (using non-stringent screening for the assayed TF and stringent screening for the other TFs).** Legend as in Table 4.2. Based on the recovery rate of the ChIP-Seq-assayed TFs, we use a maximal support of 50%. "/" indicates no CRM containing the motif for the ChIP-Seq-assayed TF was found.

4.2.6 Literature supports for detected CRMs/TFs in mouse embryonic stem cell (ESC) biology

Known combinatorial regulation of the five assayed TFs

Among these five TFs, we found 9 existing interactions involved in ESC biology between them in literature. With optimized constraints, we are able to retrieve 6 out of 9 CRMs (Figure 5.6) from our benchmark set, of documented CRMs comprising the binding sites of the assayed TFs.

Literature support for the CRMs listed in Table 4.5

By using CPModule, we predicted several novel putative CRMs involved in transcriptional regulation of embryonic stem cells (Table 4.5). Literature evidence supports a role for the retrieved TFs in embryonic stem cell biology. Besides the results on the benchmark set, we also displayed for all assayed TFs their top 3 ranking CRMs (Table 4.5). Note that those CRMs score better than the benchmark CRMs and based on literature evidence some might be true interactions involved ESC biology.

KLF4-TBP: We could not find direct literature support for the interaction between KLF4 and TBP, but TBP is a general TATA box-binding protein (Bertolino&Singh, 2002), making the interaction plausible.

NANOG-TTF1: Recent studies in mouse models have demonstrated that SOX2 regulates airway epithelium differentiation and that SOX2 and thyroid transcription factor TTF1 are modulated in concert during the course of tracheal and esophageal development (Que et al., 2007). As NANOG belongs to the same regulatory network as SOX2, at least during embryonic stem cell development, the interaction of NANOG with TFs that are also interaction partners of SOX2 is possible.

Assayed TF	Proximity constraint	Query-based rank	Support	CRM	Reference
KLF4	300	1	60%	KLF4, TBF	More details see below
	300	2	60%	KLF4, STAT4	(Bourillot&Savatier, 2010)
	300	3	61%	KLF4, CAP	Not available or no comment
NANOG	300	1	62%	NANOG, TTF1	More details see below
	350	2	61%	NANOG, BRCA	More details see below
	350	3	63%	NANOG, FAC1	More details see below
	350	1	61%	NANOG, HOXA3	More details see below
	350	2	63%	NANOG, TTF1	More details see below
	350	3	60%	NANOG, HELIOS	More details see below
OCT4	150	1	60%	OCT4, XFD2, ELF1, HMGIY	(John et al.,1995; Leger et al.,1995)
	150	2	63%	OCT4, XFD2, CDXA, HMGIY	(John et al.,1995; Leger et al.,1995)
	150	3	60%	OCT4, PAX2, XFD2, CDXA, HMGIY	(John et al.,1995; Sun et al.,2008;Gupta et al.,2006)
SOX2	150	1	63%	SOX2, OCT	(Kuroda et al., 2005)
	250	1	60%	SOX2, CDXA, AR	Not available or no comment
	250	1	62%	SOX2, CDXA, CAP	Not available or no comment
	250	1	60%	SOX2, OCT4, CDXA, LEF1	(Kuroda et al., 2005)
	250	1	60%	SOX2, OCT4, PAX2, LEF1	(Kuroda et al., 2005)
	250	1	60%	SOX2, OCT4, PAX2, SRY	(Kuroda et al., 2005)
STAT3	150	1	61%	STAT3, OCT4, STAT1, STAT5A, STAT6	(Hall et al., 2009)
	150	2	61%	STAT3, OCT4, STAT6, STAT5A	(Hall et al., 2009)
	150	3	60%	STAT3, OCT4, STAT5A, STAT6	(Hall et al., 2009)

Table 4.5: **Top 3 ranked CRM composed of the assayed TF for the five assayed TFs respectively**. **Assayed TF:** TF from which the top 100 binding peaks were used to perform the analysis. **Proximity constraint (bp):** the proximity constraint at which the displayed CRM was found. **Rank: All:** the rank this CRM received in all of the solutions. **Rank:** Query-based: the rank this CRM received amongst all valid CRMs that contain the assayed TF. **Support:** the percentage of sequences from the input set in which this CRM occurs. CRM: motifs contribute to the detected CRM. **Reference:** indicates whether direct literature support was available for the retrieved CRM.

NANOG-BRCA1: Roles of BRCA1 in both homologous recombination and nonhomologous end joining DNA repair have been shown (Shafee et al., 2008). Such function of BRCA1 might also play a role during the self-renewal process to repair DNA damage.

NANOG-FAC1: The putative transcriptional regulator FAC1 is expressed in embryonic and extraembryonic tissues of the early mouse conceptus. Study showed FAC1 is essential for trophoblast differentiation during early mouse development (Goller et al., 2008). Thus there might be an interaction between NANOG and FAC1.

NANOG-HOXA3: As we known, HOXA3 is involved in wound repair (Mace et al., 2009). so it might interact with NANOG in the self-renewal process.

SOX2-CDXA: Binding of homeobox domain from CDX1 protein and SOX2 protein was shown to occur in a system of purified components (Beland et al., 2004). Although we identified a module with CDXA, CDXA and CDX1 belong to the same family and have very similar motif models.

STAT3, STAT6, STAT1: Binding of human STAT3 protein and human STAT6 protein occurs (2-hybrid assay) (Ravasi et al., 2010). STAT1 and STAT3 can form heterodimers (John et al., 1995; Levy et al., 2002). Note however that with the STAT motif models it is difficult to make the distinction between the different STAT members.

Literature support for the involvement of the retrieved TFs in functions related to ESC biology

Several CRMs involving NANOG, OCT4, KLF4, STAT3 or SOX2 were predicted by CPModule based on the ChIP-Seq data and 517 TRANSFAC PWMs. Together, these consisted of 20 additional TFs (AR, BPTF(FAC1), BRCA1, CAP1, CDX1, ELF1, FOXI1(XFD2), HMGA1, HOXA3, IKZF2(HELIOS), LEF1, PAX2, SRY, STAT1, STAT4, STAT6, STAT5A, TBP, TTF1, LEF1). Functional analysis of 20 TFs was performed by Ingenuity Pathway analysis and significant functions in which at least 10 of these TFs are involved are: gene expression, cellular growth and proliferation, cell death, cancer, cellular development and tissue morphology. Besides gene expression (which is a logical finding, since this is a general property of TFs), we thus find all functions related to cell ESC growth, death and differentiation. For a handful of the CRMs, literature evidence of binding of the TFs supports the predictions. Additionally, further literature analysis clearly points towards a role for most of the retrieved TFs in embryonic stem cell biology. For the 20 transcription factors in the list of predicted CRMs that could be mapped to Ingenuity Pathways, we searched for known functions involving at least half of the TFs in the set.

Gene Expression (AR, BPTF, BRCA1, CDX1, ELF1, FOXI1, HMGA1, LEF1, PAX2, SRY, STAT1, STAT4, STAT6, STAT5A, TBP)

Cellular Growth and Proliferation (AR, BRCA1, CDX1, ELF1, HMGA1, HOXA3, IKZF2, LEF1, PAX2, SRY, STAT1, STAT4, STAT6, STAT5A)

Cell Death (AR, BRCA1, CDX1, HMGA1, HOXA3, IKZF2, LEF1, PAX2, STAT1, STAT4, STAT6, STAT5A, TBP, TTF1)

Cancer (AR, BRCA1, CDX1, HMGA1, HOXA3, LEF1, PAX2, STAT1, STAT6, STAT5A)

Cellular Development (AR, BRCA1, CDX1, HMGA1, LEF1, PAX2, SRY, STAT1, STAT4, STAT6, STAT5A)

Tissue Morphology (AR, BRCA1, CDX1, FOXI1, HOXA3, LEF1, STAT1, STAT4, STAT6, STAT5A)

Based on this search it appears that all of the TFs that were found in our detected CRMs have a function related to cell ESC growth, death and differentiation. Extra

information on how the retrieved TFs can modulate ESC biology (by a function in cellular growth, differentiation or morphogenesis) was retrieved from literature:

Human **TBP** protein increases anchorage-independent growth of cells (Johnson et al., 2003).

STAT4 activation is involved in differentiation of type 1 helper T cells (Farrar et al., 2002).

CAP1 has a role in apoptosis (Wang et al., 2008).

TTF1 is involved in lung morphogenesis (Hosgor et al., 2002).

HELIOS is expressed in the earliest hematopoietic sites of the embryo (Kelley et al., 1998).

HMGA1 affects embryonic stem cell lymphohematopoietic differentiation (Battista et al., 2003).

FOXI1 genetic and biochemical data suggest a central role in embryonic development for genes encoding forkhead proteins (Pierrou et al.,1994).

ELF1 plays an important and non-redundant role in the development and function of NKT cells (Pierrou et al., 1994). Homozygous knockout of ELF1 in mice affects development of heart, brain, liver and gastrointestinal tract (Choi et al., 2010).

CDX1 is involved in axial patterning and intestinal cell differentiation (Tang et al., 2003; Beck et al., 2010).

AR is required for male embryonic sexual differentiation (Part et al., 2009).

LEF1 regulates lineage differentiation of multipotent stem cells in skin (Holdcraft & Braun et al., 2004). Mouse LEF1 is involved in differentiation of paraxial mesoderm and morphogenesis of embryonic limb (Merril et al., 2001).

PAX2 is involved in nephric lineage specification (Galceran et al., 1999) and urogenital development (Bouchard et al., 2002).

SRY is the master switch in mammalian sex determination (Torres et al., 1995).

The JAK1-**STAT1-STAT3** pathway promotes proliferation and prevents premature differentiation of myoblasts (Kashimada & Koopman, 2010).

STAT5A is required for embryonic thymocyte production, TCRgamma gene transcription, and Peyer's patch development (Sun et al., 2007). STAT5A promotes multilineage hematolymphoid development in vivo through effects on early hematopoietic progenitor cells (Kang et al., 2004).

STAT6 protein is necessary for development of T-helper cell (Snow et al., 2002).

4.3 Conclusion

Our results illustrate that using ChIP-Seq information together with combinatorial CRM detection is able to prioritize true combinatorial interactions between the assayed TF and any other TF. The success of our approach stems from combining ChIP-Seq information to not only determine a set of coregulated genes, but to also delineate the region in which at least the assayed TF binds with a powerful combinatorial approach that allows detecting combinations of the binding site of the assayed TF with any other known TF for which a PWM has been reported. In contrast to gene centered methods, ChIP information allows reducing largely the regions in which the motif of the assayed TF should be located (typically 500 bp instead of thousands of bp). However, as we have no clue about the combination of TFs with which the assayed one will co-occur nor in which sequences the CRMs will possibly occur, CRM detection in ChIP-Seq defined regions still boils down to a combinatorial search problem. This combinatorial problem is solved using CPModule, a novel approach of CRM detection with a performance that is competitive to that of other state-of-art tools, but that in contrast to previous tools can handle much larger datasets (such as 100 sequences in combination with a library of 517 PWMs). The advantage of CPModule is that it builds upon a constrained-based itemset mining framework CP4IM (26): this offers the advantage of flexibly adding relevant constraints and a straight forward application of existing itemset mining principles. This allowed us to use CPModule in a query-based setting, searching for CRMs only that contained our motif of interest, i.e. the motif of the assayed TF and that meet other biologically relevant constraints that help us to prioritize the most likely biologically true CRMs, such as encompassing a restricted region (proximity constraint) or occurring in a high number of sequences (frequency constraint (support)). Note that the use of CPModule is not restricted to the application described in this paper, but can be used for CRM detection in general.

Our results also showed that the quality of the screening input largely affects the outcome of the combinatorial search. A too dense screening obtained by a non-stringent screening threshold results in too many motif combinations that make the problem intractable or in case an output is obtained decreases the prediction power (too many false positive but valid combinations are possible). Just increasing the stringency of the screening seems not to be an option as then many true sites and thus also true CRMs seem to be missing. Using a lower screening threshold in combination with a filtering procedure based on epigenetic features seemed to provide a good trade off between recovering true sites while still keeping the number of false positives within a reasonable range.

4.4 Materials and Methods

4.4.1 Datasets

The synthetic dataset retrieved from Xie et al. 2008 consists of 22 genomic sequences each 1000 base pairs in length. In 20 sequences, sites from the TRANSFAC position weight matrices (PWMs) of respectively OCT4, SOX2 and FOXD3 are inserted in a region of at most 164 bp (so the module encompasses maximally 164 bp). Each site is inserted three times per sequence. Two sequences have no sites inserted. The real-life dataset was derived from genomewide chromatin immunoprecipitation data obtained with DNA sequencing (ChIP-Seq) for the TFs KLF4, NANOG, OCT4, SOX2 and STAT3 as described by Chen et al. 2008. The input set consisted of 100 sequences, each corresponding to 500bp centered around one of the top-100 ChIP-binding peaks of the assayed TF. Binding peaks were taken from the GEO file GSE11431 (Barrett et al., 2009).

4.4.2 Step 1: Motif screening based on epigenetic features

Motif screening

PWMs for screening were derived from TRANSFAC. Each PWM from TRANSFAC were used as query PWM that was compared to all the other PWMs, using the program MotifComparison (Coessens et al., 2003) that implements the Kullback-Leiber distance between matrices. Very similar PWMs with a distance lower than 0.1 to the query PWM will be removed. This resulted in a final list of 516 PWMs used to perform screening on the synthetic data. Because it was not available in TRANSFAC, we added to the non-redundant TRANSFAC list a KLF4 PWM for the analysis of the sequence set derived from the ChIP-Seq data. This PWM was derived by Whitington et al. 2009 using de novo motif detection on a set of coregulated genes involved in the development of mouse ES cells. Coregulation was derived from a ChIP-chip experiment by Jiang et al., 2008 (Jiang el al. 2008) independent from the ones used in this study.

Motif screening was performed with Clover. After screening each subsequence of length W a log ratio of the motif being generated by the PWM versus it being generated by a background is obtained. The default threshold of 6.0 was used to define a stringent screening while a threshold of 3.0 corresponded to a non-stringent screening.

Filtering based on epigenetic features

GC content score: The GC content related effects are cell type dependent. As we are looking at stem cells, where many CpG-rich promoters are silenced by bivalent histone modifications (Bernstein, et al., 2006), thus making the sites unaccessible for TFs. Based on this, the GC content of a genomic sequence was used to estimate the bivalent histone modification level. For each potential motif site we calculated its GC content score as the fraction of G or Cs within a window of 50 bp centered around the motif site, as is done in (Xi et al., 2010; Ramsey et al., 2010). Only predicted motif site located within a low GC content region, in our case 50% were retained.

Nucleosome occupancy score: for each potential motif site we calculated its nucleosome occupancy score. To this end we first assigned a nucleosome occupancy probability to each base pair position of the potential motif site using the prediction model "NuPoP" of Xi et al. 2010. The final nucleosome occupancy score (NuOS) for a potential motif site was then calculated as the geometric mean of the nucleosome occupancy probabilities at all positions of the potential motif site (Ramsey SAet al., 2010). Only predicted motif site located within a low probability of nucleosome occupancy region, in our case 10% were retained.

The "NuPoP" model didn't predict the cell-type specific but the species specific nucleosome occupancy. Different cell types from the same organism can exhibit quite different linker DNA length distributions (Van Holde, 1998). As most of the nucleosome occupancy prediction models are supervised methods, a useful future refinement would utilize high quality nucleosome maps for the given cell type, when such data become available.

4.4.3 Step 2: CRM detection based on constrained-based itemset mining

Combinatorial search

The combinatorial problem is solved by constrait programming for itemset mining using the generic framework (CP4IM). In this framework, a problem is formalized in a *model*, while the problem is solved using a generic solver. The *model* is a specification of a problem in terms of constraints, given by the user. It can be formalized as a triplet (V, D, C) consisting of variables V, a domain $D(v)$ of possible values for each variable $v \in V$, and a set of constraints C. Each constraint is defined on a set of variables. A solution to the model is an assignment of one value to each variable that satisfies all the constraints. The search is done by a generic solver that uses the constraints to enumerate only valid solutions.

The *search* strategy taken by a CP solver is based on depth-first backtracking search. It is an alternation of branching, in which a variable is assigned a value from its domain, and propagation. Propagation is the process of using a constraint to remove values from the domain of variables that would violate it. Every constraint has a corresponding propagator that does its propagation. As a simple example: for constraint $x + y \geq 2$ and initial domains $D(x) = D(y) = \{0, 1\}$, the corresponding propagator would remove value 0 from x and y their domains. The advantage of using a generic framework is that additional constraints can be added in a modular and straightforward way by adding propagators, preventing the reimplementation of the itemset mining strategy from scratch. For more details on the implementation of the propagators we refer to (Guns et al., 2010).

The core of CPModule is a combinatorial search strategy that uses as input the motif sites located in the input sequences by motif screening. It enumerates all possible motif sets, where a motif set is defined as a subset of all screened motifs $\mathcal{M}$. Valid motif sets (CRMs) are defined as those motif sets which 1) occur frequently in the input set (frequency constraint) of sequences $\mathcal{S}2$) , 2) when determining the occurrences, only sequences are considered in which the contributing motif sites (MS) appear in each others proximity (proximity constraint) and 3) motif sets must be non-redundant (redundancy constraint).

The result of the screening and filtering step is for each motif M and sequence S a set of motif set (MS) intervals $MS(M, S) = \{(l, r)1 \leq 1 < r \leq |S|)\}$, M has a site at (l, r); here (l, r) is an interval between positions l and r on the sequence. If there is a region in the sequence in which each of the motifs has at least one motif site, we say that they are in each others proximity. The maximal distance θ of that region is specified by the user and controls the level of proximity. More formally, a set of motif $\mathcal{M} = \{M_1, \ldots, M_n\}$ is a potential CRM in a sequence S if and only if its set of hit regions (HR) is not empty, where the set of hit regions (HR) is defined as follows:

$$HR(\mathcal{M}, S) = \{(l, l+\theta) \mid 1 \leq l \leq |S|, \forall \in \mathcal{M} : \exists (l', r') \in MS(M, S) : l \leq l' < r' \leq l+\theta\}.$$

Given a set of sequence $\mathcal{S}$, the subset of sequences in which a set of motifs $\mathcal{M}$ forms a potential CRM is denoted by $\varphi(\mathcal{M}, \mathcal{S})$.

We propose a CSP (Constraint Satisfaction Problems) formulation in which there is a Boolean variable $\widetilde{M_i}$ for every motif, indicating whether this motif is part of the motif set. If a certain $\widetilde{M_i} = 1$, then we say that the motif is in the motif set; otherwise the motif is not in the set. Furthermore, we have a Boolean variable $\widetilde{S_j}$ for every genomic sequence, indicating whether the motif set is a potential CRM in a sequence, i.e. whether $S_j \in \varphi(\mathcal{M})$. Lastly, we define a boolean variable $\widetilde{seqM_{ij}}$ for every motif i and every sequence j. The variables $\widetilde{seqM_{ij}}$ indicate whether motif M_i is in the proximity of the motifs in motif set $\mathcal{M}$ on sequence j (we will define this more formally below).

We propose a constraint programming model for the CRM detection problem, consisting of 3 different constraints, the following constraints are imposed on these variables:

Proximity Constraint: The essential constraint is the proximity constraint, which will couple the $\widetilde{seqM}_{ij}$ variables to the variables representing motifs. Formally, we define the $\widetilde{seqM}_{ij}$ variables as follows for every motif on every genomic sequence:

$$\forall ij : \widetilde{seqM}_{ij} = 1 \Leftrightarrow (\exists (l, r) \in HR(\mathcal{M}, S_j) \\ \exists (l', r') \in MH(M_i, S_j) : l \leq l' < r' \leq r). \quad (4.1)$$

In other words, if in a particular genomic sequence a particular motif is within a hit region of the motif set, this motif's variable for that sequence must be 1. Observe that $\widetilde{seqM}_{ij} = 1$ will hold for all motifs in the motif set $\mathcal{M}$, for all sequences that are in $\varphi(\mathcal{M})$; however, there may be additional motifs that have hits in the proximity of regions in $HR(\mathcal{M}, S_j)$.

Frequency Constraint: The constraint that imposes a minimum size on $\varphi(\mathcal{M})$ is easily formalized as:

$$\textstyle\sum_j \widetilde{S}_j \geq \mathit{min_frequency}. \quad (4.2)$$

Here the $\widetilde{S}_j$ variables are defined as follows in terms of the $\widetilde{seqM}_{ij}$ variables, such to ensure that only sequences are counted in which all selected motif occur within each other's proximity:

$$\forall j : \widetilde{S_j} = 1 \Leftrightarrow (\forall i : \widetilde{M_i} = 1 \vee \widetilde{seqM}_{ij} = 1) \quad (4.3)$$

Redundancy Constraint: Exhaustive search is likely to consider a large number of solutions, some of which can be considered redundant with respect to each other. This is partly due to the non-sparsity of the data (data typically consists of multiple binding sites for most motif and sequence combination). For instance, if a motif set consisting of 5 motifs $\{a, b, c, d, e\}$ meets the proximity and frequency constraints, then any of its subsets $\{a, b, c, d\}, \{a, b, c, e\}, \ldots, \{b, c, e\}, \ldots, \{e\}$ will also contribute to CRMs that meet the same constraints and hence will be reported as a solution. Many of those subsets contribute to CRMs that occur in exactly the same sequences and often contain exactly the same binding regions as those identified for the larger superset. Tests on small datasets indicated that up to 80% of the solutions, and hence computation time, is spent on enumerating redundant solutions. To avoid these solutions, we imposed that a solution has to be maximally specific given the sequences that it covers. More formally, we require that

$$\forall i : (\forall j : \widetilde{S_j} = 1 \Rightarrow \widetilde{seqM}_{ij} = 1) \Rightarrow \widetilde{M_i} = 1 \quad (4.4)$$

i.e. if a motif is within the proximity of the selected motifs on all selected sequences, then this motif should be selected as well.

These constraints are modeled in a state-of-the-art constraint programming system, which effectively searches for solutions.

Genome-wide enrichment score calculation and ranking

To assess the significance of each of the potential CRMs found in the previous step, we determine their statistical significance. We want to find motif sets which are very specific to our target genomic sequences, but not to a background model (Gallo et al., 2007). The background model consists of a large number of intergenic sequences sampled from intergenic regions of the mouse genome, as downloaded from the UCSC database.

We use these background sequences to calculate an enrichment score (p-value), and rank the potential CRMs accordingly.

To calculate the enrichment score, we adapt the strategy proposed in MINI (Gallo et al., 2007). We compare the number of observed sequences that contain the motif set, $|\varphi(\mathcal{M}, \mathcal{S})|$, with the expected number of sequences. The latter is estimated by counting the number of background sequences containing the motif set, $|\varphi(\mathcal{M}, \mathcal{S}_{background})|$, where we use the exact same screening and filtering strategy as on the input sequences. For the application on the ChIP-Seq data we retain for the final background set only those sequences that contain at least one motif of the ChIP-assayed TF (Query-based mode). From this, we calculate a genome-wide enrichment score (p-value) by means of a cumulative binomial distribution:

$$p - value(\mathcal{M}) = \sum_{i=|\varphi(\mathcal{M},\mathcal{S})|}^{|\mathcal{S}|} \binom{|\mathcal{S}|}{i} p^i (1-p)^{|\mathcal{S}|-i}; \tag{4.5}$$

where $p = |\varphi(\mathcal{M}, \mathcal{S}_{background})|/|\mathcal{S}_{background}|$; $\mathcal{S}$ is the set of target sequences; $\mathcal{S}_{background}$ the set of background sequences.

Note that in this ranking, for two motif sets $\mathcal{M}_1 \subseteq \mathcal{M}_2$, with $\varphi(\mathcal{M}_1, \mathcal{S}) = \varphi(\mathcal{M}_2, \mathcal{S})$, motif set $\mathcal{M}_2$ will never score worse than $\mathcal{M}_1$ and is hence of more interest. Avoiding redundancies in step 2 hence makes the ranking more useful.

4.4.4 Motif level correlation coefficient (mCC) and nucleotide level correlation coefficient (nCC)

As in (Klepper et al., 2008), we evaluated the performance of the different CRM tools on the synthetic data at both the motif level and nucleotide level. At the

motif level, a predicted motif for a sequence is a true positive (TP) if that motif was indeed inserted in that sequence, otherwise it is a false positive (FP). If a motif was not predicted, but was inserted in that sequence, it is counted as a false negative (FN), otherwise as a true negative (TN). As the motif-level evaluation does not take the predicted binding sites into account, we also evaluate a solution at the nucleotide level: for every nucleotide we verify whether it was predicted to be part of the CRM and whether it should have been predicted or not, again resulting in TP, FP, FN, and TN counts. These counts are aggregated over all sequences to obtain the total counts of this solution. Ideally, a solution score should be good at both the motif and nucleotide level. The value of correlation coefficient (CC) lies in the range of -1 to 1. A score of +1 indicates that a prediction corresponds to the correct answer. Random predictions will generally result in CC values close to zero.

$$CC = \frac{TP \times TN - FN \times FP}{\sqrt{(TP + FN)(TN + FP)(TP + FP)(TN + FN)}} \tag{4.6}$$

4.4.5 Running parameters

For the tests on the synthetic data: Clover was used as motif screening tool with default thresholds unless mentioned otherwise and without the randomization options. For CPModule tool we use a proximity constraint of 165 bp (as this was the distance used to simulate the data) and a frequency constraint of 60%. For the other tools, we used the best parameter values according to the characteristics of the data (length of the sequences, the distance between two insertion sites, and the maximum size of CRM) listed in Table 4.6 and default values otherwise.

Algorithms	Parameter	Setting
Cister	Average distance between motifs	20
Cluster-Buster	Gap	20
	Residue abundance range	1000
ModuleSearcher	Search algorithm	Genetic algorithm
	Average number of motifs	6
	Maximum CRM size	165
	Multiple copies of TF	No
	Incomplete CRM penalty	No
	GA iteration	300
Compo	Forbid overlaps	Yes
	Number of motifs	between 1 and 20
	Distance window	165
	Tp-factors	2,3 and 4
	Background sequences	Same as CPModule

Table 4.6: **Non-default settings for alternative algorithms**.

Chapter 5

ViTraM: VIsualize TRAnscriptional gene Module network

5.1 Motivation

Organisms are able to adapt their cellular machinery to changing environmental conditions. This complex cellular behavior is mediated by the underlying regulatory network. When the regulation takes place at the level of mRNA regulation, we talk about the transcriptional network. Previous studies have unveiled the modular and hierarchic organization of the transcriptional network (Hartwell et al., 1999; Guelzim et al., 2002; Tanay et al., 2004). Indeed, biological processes consist of pathways that mainly act on their own although communication exists between these pathways. Therefore one might expect that the distinct biological processes are organized in discrete and separable modules.

Biclustering tools form one type of transcriptional module detection tools. These algorithms make use of microarray compendia to reveal the modularity of the transcriptional network. A bicluster (or module) is defined as a group of genes that show a similar expression profile in a subset of experiments. Genes within a bicluster usually belong to the same pathway or have a related biological function. Other transcriptional module detection tools (Tanay et al., 2004; Lemmens et al., 2006; Lemmens et al., 2009; Bar-Joseph et al., 2003; Segal et al., 2003; Xu et al., 2004) go one step beyond. Not only do they search for the modules, but they also identify the regulatory program responsible for the observed coexpression behavior

of the genes in the module.

Usually, many overlapping modules are identified by module detection tools. Indeed, genes can be involved in multiple pathways. In addition, multiple pathways can be triggered in one particular environmental cue. Having a visual overview of how these modules overlap, gives insight in the structure of the biological system. Biclustering software usually includes the possibility to visualize the retrieved modules one at the time, but rarely simultaneously.

For instance, BiVisu (Cheng et al., 2007), BicAt (Barkow et al., 2006) and Expander (Shamir et al., 2005) allow visualizing the genes and experiments of one module by means of an expression profile or a heatmap. The problem with visualizing overlapping modules simultaneously is that the overlap in multiple dimensions complicates the choice of an appropriate layout. Therefore few tools exist that are capable of visualizing modules simultaneously.

In Grothaus et al., 2006 for instance, a tool for the visualization of multiple, overlapping biclusters in a two dimensional gene-experiment matrix was developed. As each bicluster is represented in this layout-matrix as a contiguous submatrix, genes and experiments that belong to multiple overlapping biclusters will be duplicated to obtain an optimal layout of the biclusters. This duplication of genes and experiments, however, complicates the biological interpretation of the biclusters. The recently developed tool BicOverlapper (Santamaria et al., 2008) displays overlapping biclusters by means of a graph-based representation. The nodes in the graph represent respectively experiments and genes of the data set. An edge between two nodes indicates that the connected nodes are part of the same bicluster. A bicluster is thus represented as an undirected, fully connected subgraph. The nodes are positioned in the display based on their bicluster assignment: nodes of the same bicluster will be placed close to each other while nodes belonging to different biclusters will be positioned at a larger distance. Nodes that are in common between multiple biclusters will be placed in between those biclusters.

5.2 Introduction

Revealing the complete regulatory network underlying the cell's behavior is one of the major challenges in current research. Recently, there is a growing interest in the modular description of regulatory networks. Biclustering algorithms form one type of algorithm for revealing the modularity of the network. A bicluster or a module is defined as a group of genes that show a similar expression profile in a subset of experiments. Genes within a bicluster usually belong to the same pathway or have a related biological function. Other transcriptional module detection tools go one step beyond. Not only do they search for the modules, but they also identify the

regulatory program responsible for the observed coexpression behavior of the genes in the module.

Usually, many overlapping modules are identified by module detection tools. Having a visual overview of how these modules overlap, gives insight in the structure of the biological system. Biclustering software usually includes the possibility to visualize the retrieved modules one at the time, but rarely simultaneously. The problem with visualizing overlapping modules simultaneously is that the overlap in multiple dimensions complicates the choice of an appropriate layout. Therefore few tools exist that are capable of visualizing modules simultaneously.

In this study we developed ViTraM (Sun et al., 2009; Bollen master thesis, 2006) that allows for a dynamic visualization of overlapping transcriptional modules in a 2D gene-experiment matrix. Multiple methods are included for obtaining the optimal layout of the overlapping modules. In addition to the previously developed tools for visualizing multiple modules, ViTraM also allows to display additional information on the regulatory program of the modules. The regulatory program consists of the transcription factors and their corresponding motifs. A first way of obtaining information on the regulatory program is by using the information from curated databases. This information can be used to further analyze modules inferred by biclustering algorithms. Secondly, information on the regulatory program can also be the outcome of a module inference tool itself. Both types of information on the regulatory program can be included by ViTraM. By visualizing not only the modules, but also the regulatory program, ViTraM can provide more insight into the modules and makes the biological interpretation of the identified modules less complicated for biologists.

The XMLCreator will use the input and output data of DISTILLER (Lemmens et al., 2009), together with the additional data to create the XML file and expression data file that are required for visualization by ViTraM. Although the XMLCreator currently only includes the possibility to derive the XML file from the output and input of the module detection tool DISTILLER, more algorithms will be included in the future.

5.3 XMLCreator

XMLCreator is used as the interface to formate the module result in an XML format which can be used as input for ViTraM.

5.3.1 Requirements for installation of XMLCreator

Developed in JAVA, the XMLCreator is platform independent and is expected to work under other operating systems (Windows, Linux, Mac) that support the JRE (1.5 or higher) and with sufficient memory depending on the size of the input data.

5.3.2 Installation of the XMLCreator

The software can be downloaded from the download section on:
http://homes.esat.kuleuven.be/ kmarchal/ViTraM/Index.html

After downloading the package, please follow these steps:

- Unzip the downloaded file
- Open the unzipped folder
- Depending on the OS:

Windows or Mac:

- Double click on the file XMLCreator.jar to run the software.
- Or open a command line window, and execute the command "java -jar -Xms64m -Xmx256M XMLCreator.jar" in the folder in which the files of the XMLCreator are stored.

Linux:

- Run the XMLCreator in a terminal with command "java -jar -Xms64m -Xmx256m XMLCreator.jar".

If everything is OK, the XMLCreator should start right now and the following window appears (Figure 5.1). By choosing DISTILLER in the previous step, the following window will appear (Figure 5.2). When the data is created, a pop-window will be shown.

5.3.3 Data for XMLCreator

The following data files are required for visualization by ViTraM and thus for generating the XML file that is required by ViTraM:

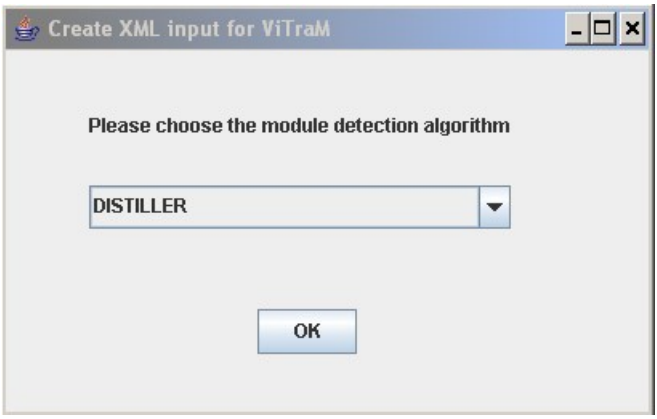


Figure 5.1: The interface of XMLCreator.

- DISTILLER/Query-driven Biclustering (QDB) output data: A file that contains information on the modules, i.e. which genes are co-expressed in which experiments. If not only expression data is available but also regulator and/or motif data, DISTILLER will not only derive modules, but also information on the regulation of the module. Information on which regulators and/or motifs are in control of which modules will then also be present in the output file.
- Expression data: The expression data contains the array names in the first row, while the first column consists of the gene names. This is the expression data that was used as input for DISTILLER.

If regulator and/or motif data were used as input for DISTILLER, these data should also be provided to the XMLCreator. The order of the gene names must be the same as the ones in the expression data.

- Motif data: The motif data consists of motif names in the first row and gene names in the first column.
- Regulator data: The first row of the regulator data are the regulator names, whereas the first column contains the gene names.

Finally the user can also include additional data sets like information on the gene function or on the experiments. The latter files are however not required if not available.

- Gene function data: The gene function file consists of binary data indicating whether a gene is for instance member of a particular gene ontology category.

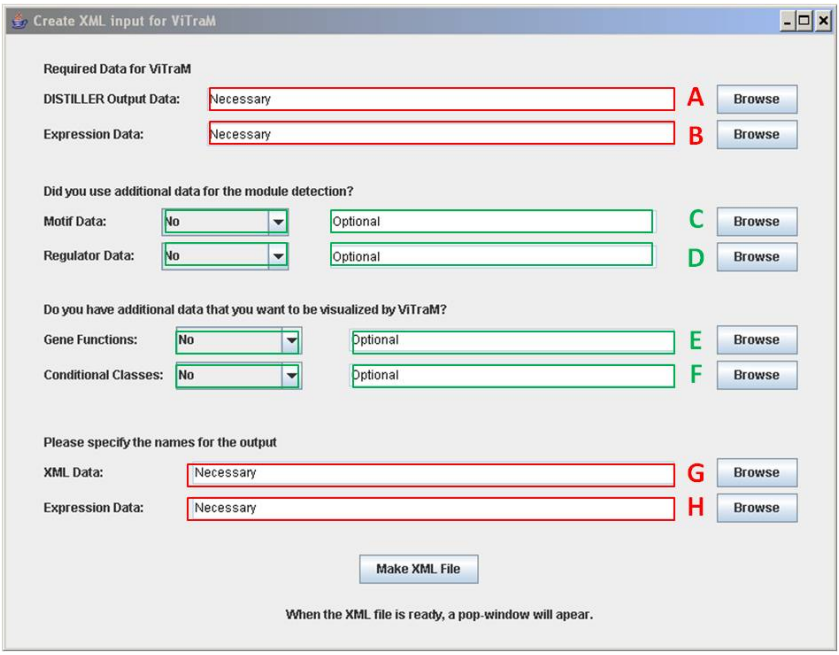


Figure 5.2: **The XMLCreator software**. The output (A) from DISTILLER and the expression data (B) used as input for DISTILLER are required for creating the XML file (G) and corresponding expression file (H) for ViTraM. There are also optional files (C, D, E, F) that can be specified for creating the output files.

The first row contains the gene names, whereas the first column contains the functional categories.

- Conditional classes: A conditional class gives information on the major cue that was measured during the experiment. A similar file as for the gene function data can be derived for the conditional classes.

In addition, the user should indicate which data file was used as input by DISTILLER. It is possible to include only one data, such that the user should indicate ”Data source 1”. If both data sources were included in the analysis, the user has to indicate which data source is the first data source and which one is the second in the input of DISTILLER. Otherwise the output of DISTILLER can not be interpreted well. The gene, motif, regulator and experiment names that are used in these data sets will be displayed in the visualization by ViTraM and gene names should be used consistently in all files.

5.4 ViTraM

In this study we developed a tool that allows for a dynamic visualization of overlapping transcriptional modules in a 2D gene-experiment matrix. Multiple methods are included for obtaining the optimal layout of the overlapping modules. In addition to the previously developed tools for visualizing multiple modules, our tool also allows to display additional information on the regulatory program of the modules. The regulatory program consists of the transcription factors and their corresponding motifs. A first way of obtaining information on the regulatory program is by using the information from curated databases or other data sources. This information can be used to further analyze modules inferred by biclustering algorithms. Secondly, information on the regulatory program can also be the outcome of a module inference tool itself. Both types of information on the regulatory program can be included by our visualization tool (Figure 5.3). By visualizing not only the modules, but also the regulatory program, our tool can provide more insight into the modules and makes the biological interpretation of the identified modules more accessible to biologists.

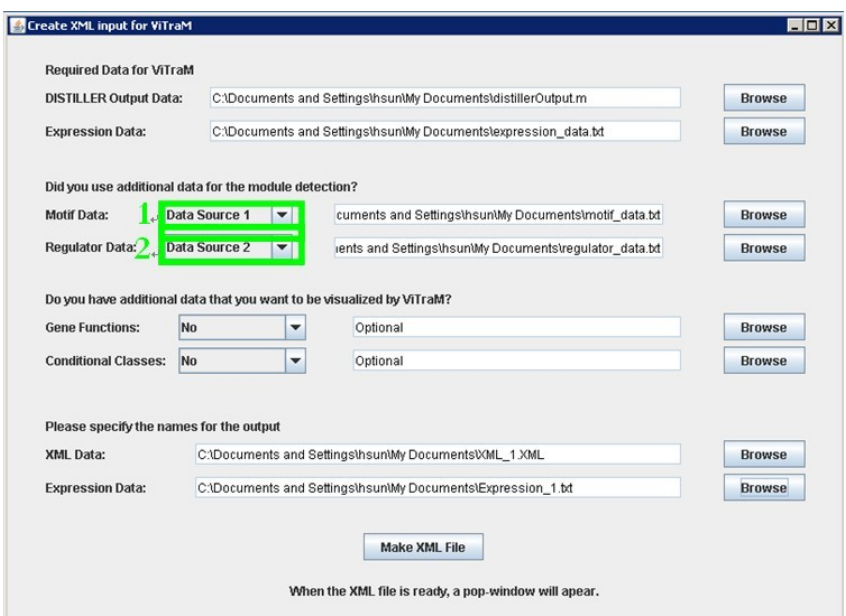


Figure 5.3: When you have two data sources, e.g. regulator and motif, we should indicate which is data source 1 (1) and which is data source 2 (2) to make a clear XML file.

5.4.1 Inputs for ViTraM

ViTraM requires two input files:

- Module file (minimal input)
- Expression data file

Module data

The module file should be loaded first. This module file is an XML file that contains the minimal information to visualize the modules. Optionally the module properties and the gene and experiment properties can be included (see below). The expression values of the genes per experiment are given in the expression data file. The structure of the 'module file' is shown in Figure 5.4. This file contains the following information:

- General information: The used xml version and/or the algorithm that was used for retrieving the modules or the date can be mentioned here.
- Gene property information: This part requires minimally for each gene a unique gene name and id. The id corresponds to the row number in the expression file (see below), such that each gene can unambiguously be linked to its expression values in the expression matrix (one to one relation). The gene ids should start from 1. Optionally, additional gene properties can be added (but this is not strictly required). These optional properties include membership to a particular gene ontology class or the presence of a transcription factor binding site (motif) in or the binding of a transcription factor to the gene's upstream region. Additional gene properties are indicated as follows: a binary value is used to indicate whether a gene belongs to a particular gene ontology functional class (0 = absent, 1 = present); the interaction of a transcription factor with a gene is indicated by a binary value or a score (depending on the source of information, ChIP-chip data, motif screening, ...).
- The experiment property information: Again, for each experiment, a name and id are strictly required. The experiment id refers to the corresponding column in the expression data file (see below) and start from 1. Optionally, the user can add experiment properties, such as a classification of the experiments according to the cue that was measured in that experiment. A value 'one' for a particular conditional functional class indicates that an experiment belongs to this class.
- Gene properties list: If additional gene properties, such as membership of a gene to an ontology class, a regulator or a regulatory motif, were assigned to the genes

and mentioned in the "gene property information", a list of these properties should be provided in this section.

- Gene-experiment information: Contain as a minimal level of information the actual composition of the regulatory modules. A regulatory module is defined here as a set of co-expressed genes and the experiments under which they are co-expressed. This part of the file should thus minimally contain the ids of the genes and experiments of which each module is composed. Depending on the inference algorithm that was used, a regulatory program can have been assigned to the module, such as the set of regulators or regulatory motifs that regulate the module. This information is called the additional "module properties". Such additional module properties can be included in the XML file, but are not required.

Expression data

The microarray expression data should be loaded in case the user wants to plot the modules' expression profiles or heatmaps. The expression file is a tab-delimited file. Gene names (ids) or experiment names (ids) are mentioned in the file in the first column and first row, respectively.

5.4.2 Structure of ViTraM

ViTraM is a java-based tool for visualizing multiple overlapping modules together with additional information on their regulatory program. The minimal information required by ViTraM to visualize modules are the genes and experiments composing the modules (Figure 5.5). Additional properties of the genes, experiments and modules are optional. A gene property includes membership to a particular gene ontology class or the presence of a transcription factor binding site (motif) in or the binding of a transcription factor to its upstream region. An experiment property includes the membership of an experiment to a particular conditional class, which gives information on the major cue that was measured during the experiment. A module property is only available when the module inference tool was capable of identifying the regulatory program of the module. This module property consists of the list of transcription factors or transcription factor binding sites that were assigned by the inference algorithm to the module. Note that the latter information on the regulatory program is different from the gene properties which are not inferred, but derived from curated databases.

5.4.3 Layout algorithm

By using microarray data, module inference algorithms usually retrieve multiple modules that overlap in both genes and experiments. When loaded into ViTraM, these modules initially will be displayed according to their order in the input module file. This initial non-optimal ordering will result in modules being split up in the ModuleImageDisplay. Indeed by changing the order in which the genes and experiments are displayed, the first modules can be shown completely without being split up. Gradually adding more modules overlapping with the first displayed one reduces the flexibility of reordering, causing the last added module to be split up again. When visualizing multiple modules, it is therefore essential that the modules are placed in such a way that an optimal overview of the results can be obtained.

To improve the visualization of overlapping modules, ViTraM includes two different ordering algorithms, one based on the "overlap index" and a second one based on the "order score". If the user is specifically interested in a module that should not be split up, a user-defined ordering of the modules can also be imposed.

The overlap index is defined as the number of modules a particular module overlaps with. In order to get the optimal layout of the modules, in which as few modules as possible are split up, the module that shows overlap with the largest number of modules, i.e. the module with the largest overlap index, is displayed first. All modules overlapping with this module will be added subsequently. Next, the module with the largest "overlap index" amongst the remaining modules will be selected and placed in the ModuleImageDisplay, and again all modules that overlap with this module are positioned in the layout. This procedure will be repeated until all modules are displayed in the layout.

The second layout algorithm is based on the following "Order score" S:

$$S = \log(modulesize) \times \log(overlaparea) \tag{5.1}$$

The module size is determined by the product of the number of genes and number of experiments in a module. The overlap area is the area (number of genes $\times$ number of experiments) that a module has in common with other modules. Each module will be assigned an order score S. The module with the highest score, a large module showing much overlap with other modules will be positioned first.

Subsequently the remaining modules are placed in the ModuleImageDisplay in decreasing order of their score.

Based on either the overlap index or the order score, the optimal order in which the modules are placed in the ModuleImageDisplay is determined.

Constraint ordering of genes/experiments

Once an initial set of modules is placed in the most optimal way, the experiments (genes) can still be rearranged in order to better group the next module. In what follows we will describe the experiment grouping (the gene grouping is analogous) by means of an illustrative example. Figure 5.6 shows a situation in which Module1 and Module2 are already positioned in the canvas. Although the grouping is OK for placing both Modules1 and Module2, it is still suboptimal for placing a subsequent module (i.e. Module3). In this case Module3 shares experiments with both Module1 and Module2, but is still highly fragmented. By regrouping the conditions of Module1 and Module2 slightly, we can reduce the splitting. However, we do not have full freedom in regrouping all experiments as then Module1 and Module2 will be split up again. So when rearranging experiments in order to optimally place Module3 we have to take into account that the placing of Module1 and Module2 already posed constraints. In this situation the experiments (genes) can be subdivided in four groups: the set of experiments present in only Module1, the set of experiments present in only Module2, the set of experiments shared by Module1 and Module2, the set of experiments that are not present in any module. The constraints posed by placing Module1 and Module2 in advance allow only for rearranging experiments within one subdivision, but not between subdivisions. Experiments within one subdivision that overlap with Module3 can be grouped together reducing the number of splits required for placing Module3 (Panel 3). If more modules are present, more subdivisions will exist according to which the experiments will be grouped. When grouping the experiments (genes) into the most optimal way, we designed an iterative procedure that brings these constraints into account one by one. Note that this procedure for the constrained reordering of experiments is an inherent part of finding the optimal ordering of modules based on the "Overlap Index" and "Score Function": each time a novel module is added the constrained gene/experiment reordering procedure is applied.

5.4.4 Dynamic visualization

ViTraM visualizes the modules in a 2D display, called the ModuleImageDisplay, in which the rows represent the genes and the columns the experiments. Each regulatory module is represented in this display, as a transparent colored rectangle. General information on the currently displayed modules such as their gene, experiment, motif or regulator content is shown by ViTraM.

In addition to this display, two other displays the, GenePropsImageDisplay and ExpPropsImageDisplay, show respectively the gene properties and the experiment properties. Both displays are dynamically linked to the ModuleImageDisplay, meaning that if the order of genes or experiments changes in the ModuleImageDisplay, their order will also change in the other two displays.

The properties that are displayed in the GenePropsImageDisplay are the gene properties (i.e. membership to a gene ontology class, presence of transcription factor binding site or binding of a transcription factor). The rows represent the genes whereas the columns of this 2D display represent the gene properties. For the properties "regulator" and "motif", a color gradient indicates the values of the score for a particular property and gene combination. This score can be derived from, for instance, a motif screening in the case of the motifs or from the results of a ChIP-chip experiment in case of regulator binding. For the gene ontology membership, binary values are available: the gene either belongs or doesn't belong to the functional class. These gene properties can be included as additional information for the analysis of the regulatory modules, but are not required.

The ExpPropImageDisplay displays the experiment properties. In this image the rows represent the different experiments of the experiments and the columns show the conditional categories to which the different experiments can be assigned.

Selection of modules, genes and experiments in the ModuleImageDisplay

Although all modules resulting from a biclustering or module detection method can be visualized simultaneously by ViTraM, the user might also want to zoom in on a specific subset of modules. A subselection of modules can be made from the complete set of modules or from the currently displayed modules. For the selected modules, the layout can also be optimized using the ordering algorithms mentioned above. ViTraM provides several module selection criteria:

1. Selecting all modules that overlap with one module of interest. Overlapping modules can be defined based on overlap in experiments, overlap in genes or overlap in both genes and experiments.
2. Selecting all modules to which the same regulator or the same motif has been assigned by a module detection algorithm.
3. A user-defined selection of modules.

In addition to these module selection criteria, ViTraM also includes the possibility to further filter the output. In contrast to the selection procedures, the filtering options will always function on the currently displayed modules which allows for sequential filtering according to several criteria. The filtering techniques include:

- Filtering of genes: genes can be filtered based on their gene properties. If a gene does not satisfy the user-defined criteria, it will not be visualized.
- Filtering of experiments: experiments can be filtered based on their experiment properties. This allows the user to only visualize those experiments that measure the same cue.

- Filtering of modules. Several criteria are provided for filtering the modules, such as the number of genes or experiments contained within a module, the module size (genes × experiments) or the presence of a particular gene/experiment in the module. In addition, based on the motif/regulator score ViTraM allows selecting those modules for which a motif/ regulator is present in all genes of the module.

Filtering of gene properties in the GenePropsDisplay

Sorting the gene properties helps with the biological interpretation of the modules visualized in the ModuleImageDisplay. Motifs can, for instance, be ordered according to their score for the genes in the currently displayed modules in order to investigate the modules' regulatory program. We have included two options for ordering the gene properties in the GenePropsDisplay:

- Based on the score of the regulators/motifs. The higher the scores of a particular regulator/motif are for the genes in the currently displayed modules, the higher these regulators/motifs will get ranked in the list of gene properties.
- Based on the assignment of regulators/motifs to the modules. It is possible to only show those regulators/motifs that were assigned by a module detection algorithm to the currently displayed modules.

Additional visualizations

The general display consisting of the ModuleImageDisplay, GenePropsImageDisplay and ExpPropsImageDisplay, is used to display in detail a selection of modules, their genes and experiments together with their properties. Depending on the modules' size and number, the ModuleImageDisplay can usually only display a partial view of the selected modules in one window. Interactively navigating through the ModuleImageDisplay allows to see the rest of the selected modules into detail. The OverviewDisplay, given in a separate window, provides a less detailed but total overview of all currently displayed modules and allows the user to keep track of which part of the module selection is currently displayed in the ModuleImageDisplay.

ViTraM also provides two ways of viewing the expression values of the genes in the modules. First, as a heatmap of the expression values in the ModuleImageDisplay (low expression values are colored green, while high expression values are colored red). Secondly, by means of the average expression profile of the genes in a module in a separate window.

5.4.5 Export images

Once an optimal layout of the modules is obtained, the resulting image can be saved as a figure in SVG (Scalable Vector Graphics) format. SVG is a vector graphics formate that does not lose the resolution when zooming in.

5.4.6 Overview on some of the functionalities of ViTraM

All modules obtained from the DISTILLER module detection tool were loaded by ViTraM. Overview on some of the functionalities of ViTraM. (A) The ViTraM software consists of several displays. The ModuleDisplay shows the modules in a 2D matrix in which the rows represent the genes and the columns the experiments. The GenePropertiesDisplay shows the genes and their properties, whereas the ExperimentPropertiesDisplay shows the experiments present in the modules and their properties. (B) An overview of all modules can be seen in the OverviewDisplay. Visualization of a subset of modules is possible, for instance, selecting all modules to which a module inference tool (here DISTILLER) assigned a particular motif (here CRP). After applying one of ViTraM's ordering methods, all modules can be displayed in a coherent way. (C) The GenePropertiesDisplay shows the scores for the binding of a regulator or the presence of a motif by a color gradient in which green is the lowest value and red the highest value. After sorting the gene properties based on these scores, it is clear that the CRP motif assigned by the module inference tool indeed has a high score for both modules. (D) The ExperimentPropertiesDisplay displays the conditional categories to which an experiment belongs. A selection of experiments that should be visualized can be made, for instance, only experiments that measure the influence of the carbon source can be visualized.

5.5 Case study

5.5.1 General introduction of DISTILLER

DISTILLER (Lemmens et al., 2009) is a module detection tool that identifies sets of genes that are co-expressed in a set of conditions (or modules) and the regulatory program of the genes in these modules. The regulatory program consists of regulators and/or their corresponding regulatory motifs. Although in the original publication, only expression and regulatory motif data were used, the data integration framework is very flexible for adding more data sources, for instance ChIP-chip data. The input data for DISTILLER thus consists of expression data, usually in combination with another data source like motif screening data or

regulator binding data. DISTILLER also requires that all data sources consist of the same number of genes and that genes are ordered in the same way in all data sets.

5.5.2 Visualizing gene regulatory network constructed by DISTILLER

To demonstrate that ViTraM can assist a user in analyzing the output of a module detection tool, it was applied on the results (around 100 modules) obtained by DISTILLER (Lemmens et al., 2009). DISTILLER is a data integration tool that uses expression data and regulatory motif data to identify modules together with their regulatory program. When applied to *E. coli* data, overlapping regulatory modules were obtained together with (a) motif(s) assigned by DISTILLER to each separate module. In addition to the motifs assigned by DISTILLER, we screened all genes in the dataset for the presence of known regulatory binding sites according to RegulonDB. The regulatory modules together with their assigned motifs (module properties), and the additional motif information obtained by motif screening (gene properties) were used as input for ViTraM.

To be able to fully exploit all possibilities of ViTraM, additional information for these modules was included. The following gene and experiment properties were added to the input file: the functional classes to which module genes belonged and the conditional classes of which module experiments are part of. Expression data for the obtained modules were available in a separate expression data file. When loading the modules and the additional information into ViTraM, all modules are initially displayed according to their order in the input file, resulting in a scattered representation of the modules. By using one of the ordering algorithms of ViTraM, a more optimal layout of the modules is obtained from which the overlap structure of the different modules becomes clearer. Subsequently, we selected modules to which the module detection tool has assigned the motif CRP (Figure 5.7). In the GenePropsImageDisplay, the scores of motifs for the genes in the CRP modules are shown. When sorting the motifs in this display according to their scores, it is clear that in addition to the algorithmically assigned CRP motif, also the ArcA motif is important for at least one module and the FNR motif for a second module, as both motifs had high scores for all genes in their respective modules. The ExpPropsImageDisplay shows that many experiments in which the genes of these two modules were co-expressed belong to either the conditional category ”carbon-source” or the ”anaerobiosis_aerobiosis”. These findings are consistent with the known functions of the assigned regulator CRP. The catabolite repressor is known to be active during glucose starvation and known to interact with the regulators ArcA and FNR in response to oxygen. All modules obtained from the DISTILLER module detection tool were loaded. Subsequently modules to which the motif CRP was assigned by DISTILLER were selected by using

one of the module selection techniques. This resulted in the selection of two modules. (A) The ModuleImageDisplay allows to investigate the two selected modules into more detail: which genes and experiments are present in one or both modules. Because the modules contain many experiments, the second module is not completely visible in the ModuleImageDisplay. Interactively navigating through the ModuleImageDisplay is, however, possible and allows to see all modules into detail. (B) An overview of the modules can be seen in the OverviewDisplay. Initially, one of the modules is split up in five parts. After applying one of our ordering methods, all modules can now be displayed in a coherent way. (C) More information on the currently displayed CRP modules, such as the genes, experiments, motifs and regulators assigned to the modules, can be obtained from the information panel. (D) The ExpPropsImageDisplay shows the experiments present in the modules and their properties. A selection of experiments that measure for instance the influence of the carbon source (category carbon_source) was made. (E) The GenePropsImageDisplay shows the gene properties. Scores for the binding of a regulator or the presence of a motif are indicated by a color gradient in which green is the lowest value and red the highest value. The cross indicates those scores that satisfy a pre-defined threshold. In this example we choose the threshold for the motif scores to be in between 0.999 and 1. The gene properties can be sorted based on the motif scores. After sorting it is clear that the CRP motif has indeed a high score for both modules. For the first (upper displayed module) the ArcA motif scores are high, whereas for the second module (lower displayed module) the FNR scores are high.

For other applications of ViTraM, we refer to the application of ViTraM on the results obtained by Biclustering tool (see PhD thesis of Riet De Smet Ensemble Methods for Bacterial Network Inference (2010)) and the application of ViTraM on the results obtained by Probic (Loots et al., 2011) (Ongoing work).

5.6 Conclusion

ViTraM is a user-friendly software tool developed for the visualization and analysis of transcriptional modules and their regulatory program. The previous example shows how ViTraM allows studying in detail a subset of modules and their properties while maintaining an overview on how the different modules are related to each other. This interactive exploration of results can help biologists in the interpretation of the many modules that are present in the output of module inference tools.



Figure 5.4: **The structure of the module file**. First some general information can be listed in the module file. Subsequently, the genes and possibly their properties are listed, followed by the experiments and their properties. Then the different gene properties such as the regulators or motifs are listed. Finally the actual module information is given. Modules minimally consist of genes and experiments, but also additional module properties like a regulator or motif are possible.

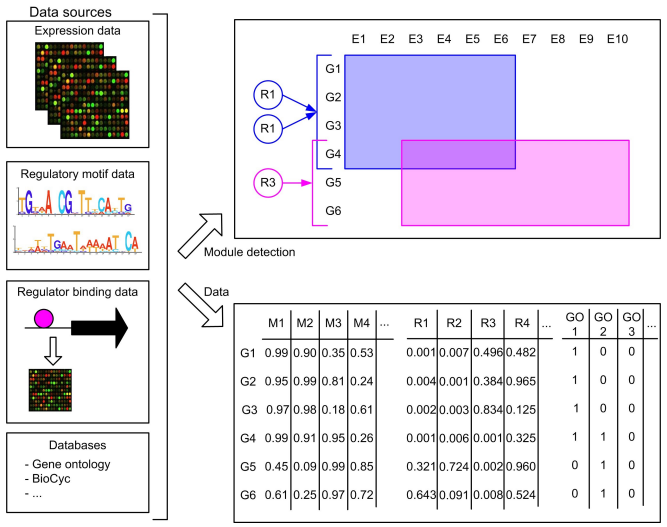


Figure 5.5: **Example of input for ViTraM**. Module detection tools can make use of several types of data, such as expression data, regulatory motif data, regulator binding data, databases, *etc.* for the identification of regulatory modules and their program. This example shows two modules (blue and purple) that consist of a set of genes that are co-expressed in a set of experiments. The blue module, for instance, consists of genes G1, G2, G3 and G4 and experiments E1, E2, E3, E4, E5 and E6. This is the minimum information that is required for ViTraM to visualize the modules. Additional information of the regulatory program can also be given as input to ViTraM and visualized. This information can be inferred by the module detection tool. In that case it represents a module property. The regulators 1 and 2 (R1 and R2), for instance, were assigned to the blue module by a module inference tool and are thus examples of properties of this module. In addition to module properties, additional information can be derived from curated databases or other datasources: in that case it represents a gene property. In the example, the presence of a motif (M1-M4) is indicated by a score ranging from 0 to 1. The higher the score the more likely the presence of the motif. Similarly the physical binding of a regulator (R1-R4) to a gene as derived from ChIP-chip data is indicated by its p-value. Membership to a gene ontology functional class (GO1-GO3) is indicated by a binary value major cue that was measured during the experiment. A module property is only available when the module inference tool was capable of identifying the regulatory program of the module. This module property consists of the list of transcription factors or transcription factor binding sites that were assigned by the inference algorithm to the module. Note that the latter information on the regulatory program is different from the gene properties which are not inferred, but derived from curated databases.

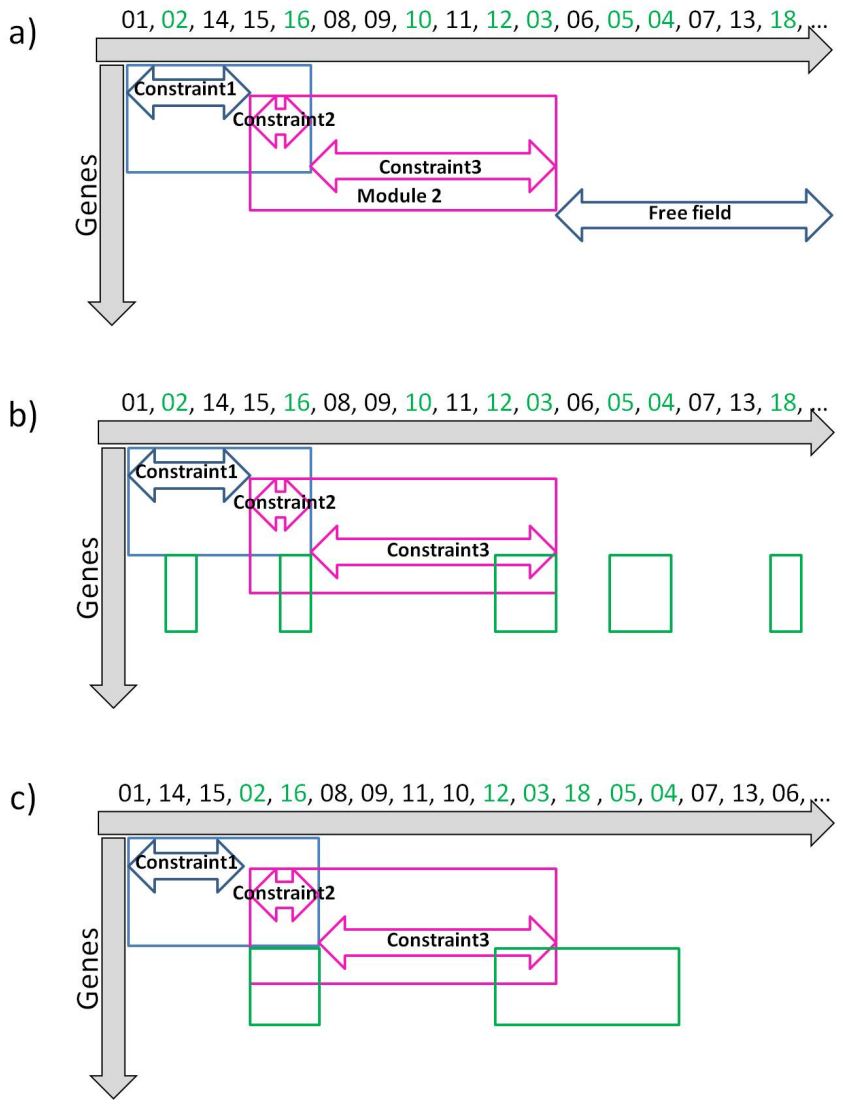


Figure 5.6: **The "constraint reordering of genes/experiments"**. Panel a; Module1 and Module2 are placed in an optimal way. Panel b: Adding Module3 using the current order of experiments will lead to a high fragmentation of Module3. Panel c: the fragmentation of module3 can be reduced by reordering the experiments, while taking into account the constraints posed by the optimal placing of Module1 and Module2. Experiments can only be freely reordered within one subdivision. Therefore experiments within each subdivision that overlap with Module3 (in this case first 2 and 16, then 12, 3, 18, 5 and 4 will be grouped together. As such, the number of splits needed to place Module3 is reduced.

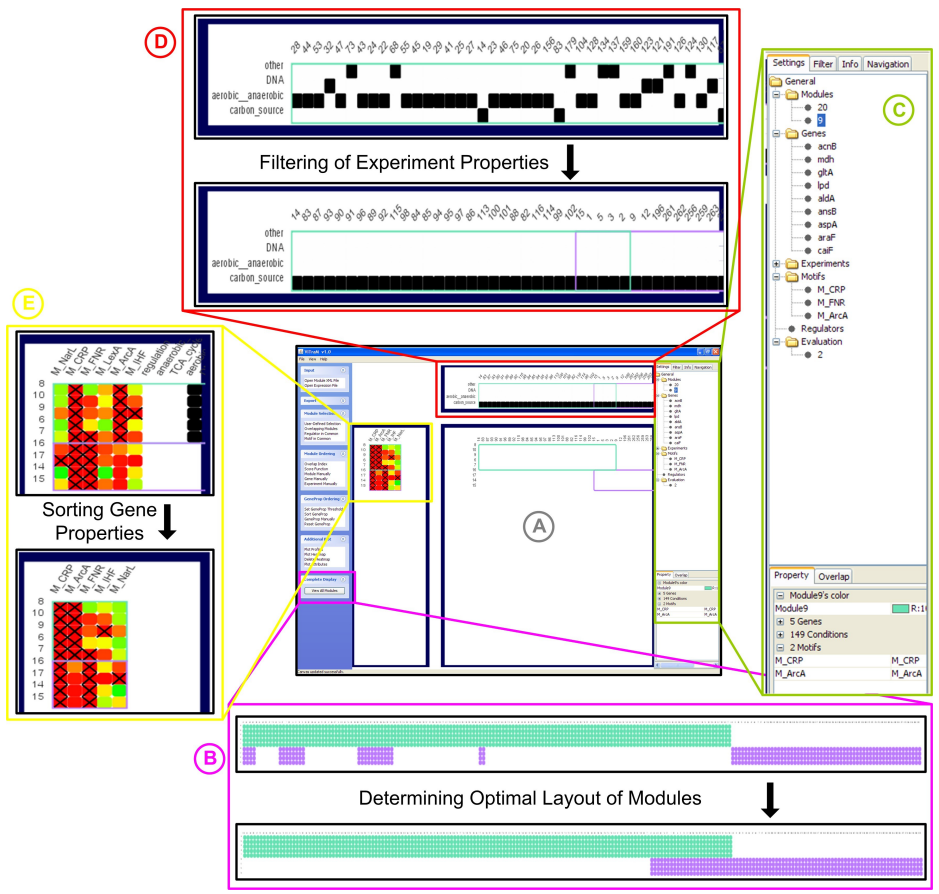


Figure 5.7: Case study with data from DISTILLER.

Chapter 6

Conclusions and Perspectives

6.1 Conclusions

Chapter 3 illustrated a tool ModuleDigger for *cis*-regulatory module detection based on itemset mining algorithm, which can handle larger dataset. By comparing with other state-of-art methods using a ChIP-chip data on human embryonic stem cell, we showed better performance of our algorithm.

Chapter 4 described a constraint programming for itemset ming based algorithm CPModule for *cis*-regulatory module detection. The algorithm simultaneously rank the multiple potentially overlapping modules containing similar set of motifs using a statistic model. This leads to the robust identification of *cis*-regulatory module under various distances and noises. This constraint programming based algorithm is built on a generic framework which makes it easy to extend. We demonstrated using a literature existing synthetic data that our CRM detection method has a performance comparable to that of state-of-the-art CRM techniques and showed on ChIP-Seq datasets how our CRM detection analysis flow unravel combinatorial regulation of the gene set of interest.

Chapter 5 presented a tool ViTraM designed for transcriptional regulatory network visualization. The ordering algorithm provided in the tool is based on the properties of the modules. The visualization algorithm in the tool is based on scalable visualization graphics (SVG), which is integrated with JAVA by using Batik project (http://xmlgraphics.apache.org/batik/). We also provide an interface which can format results from gene network construction algorithms to a XML formated structure, and ViTraM allows for visualizing the transcriptional network based on the properties of genes/conditions/modules.

6.2 Perspectives

Biomarkers In The Age of Omics

A biomarker is a measurable indicator of a specific biological state, particularly one relevant to the risk of contraction, the presence or the stage of disease (Rifai et al., 2006). Limitations to biomarker discovery are not only technical or bioinformatic but conceptual as well. The total number of cancer patients in the United States projected to increase by 55% at 2020, the need for an effective early detection methods and prevention programs becomes more crucial to ameliorate the situation of rising statistics (Roukos 2009; Warren et al., 2008). Therefore, accurate predictive biomarkers and/or profiling techniques for early detection can play an important role in affecting patients' survival and provide the proper treatment. Transcriptional profiling and DNA methylation studies have shown strong potential for biomarker discovery in cancer (Ramaswamy & Perou, 2003).

Recent studies of complex disease have revealed powerful insights into how genetic and epigenetic factors may underlie etiopathogenesis (Bell et al., 2010; Christine & Aminoff, 2004; Franks & Ling, 2010). Epigenetic mechanisms play important roles during normal development, aging, and a variety of disease conditions. Numerous studies have implicated aberrant methylation in the etiology of common human disease, including cancer, diabetes and schizophrenia (Egger et al., 2004; Parizel et al., 2003). Hypermethylation of CpG islands located in the promoter regions of tumor suppressor genes is firmly established as the most frequent mechanism for gene activation in cancers (Esteller & Herman, 2002; Herman & Baylin, 2003).

Prioritizing biomarker genes by differential coexpression analysis using itemset mining

Linking expression variation to patient information translates into finding gene sets that are differentially expressed between a group of patients (or disease affected tissues) and a control group (normal tissue) (Emilsson et al., 2008). We are not only interested in searching for individual genes that are differentially expressed between the patient and control group, but also grouped sets of genes that are mutually coexpressed in one group and no longer coexpressed or differently coexpressed in the other group (searching for genesets rather than individual biomarkers increases the statistical power of the predictions and facilitates the interpretability i.e. pathway based disease classification). A differential coexpression pattern hints at the disruption of a regulatory mechanism that might be at the cause of the observed phenotype (pathway based biomarker identification). Where most of the existing approaches search for differential coexpression over the full-space of samples (all patients versus all controls), we can search for differential coexpression patterns that do only cover a partial subset of the patients. This allows us to compensate for the presence of an unknown heterogeneity in the subject population or uncharacterized disease causes defining further subclasses in the patient groups

than the ones that were initially described.

Itemset mining strategies are very useful for solving combinatorially explosive problems. They allow enumerating all possible solutions in a concise way by using very fast search strategy. Because association rules mining (ARM) methods tend to generate a large number of itemsets or association rules, a final filtering or postprocessing step is needed to obtain biologically interesting itemsets or rules. However, as no explicit score is assigned to the solutions, it is not clear which modules are the 'most interesting' to select. Also, the output usually contains partially overlapping and redundant solutions. In our previous work we already developed an efficient itemset mining framework that uses a probabilistic filtering step (Lemmens et al., 2009). We applied it for the integration of several omics data (Lemmens et al., 2009) and the detection of regulatory modules (ModuleDigger (Sun et al., 2009)). Recently we explored the possibility of using a strategy of constrained programming for itemset mining (Sun et al., 2011, in revison) (in collaboration with Prof. Luc De Raedt from the department of computer science at KULeuven). In the future we will extend the approach towards itemset mining for differential coexpression analysis (Nijssen et al., 2009).

Bibliography

1. Adham IM, Khulan J, Held T, Schmidt B, Meyer BI, Meinhardt A, Engel W: Fas-associated factor (FAF1) is required for the early cleavage-stages of mouse embryo. Mol Hum Reprod 2008, 14(4):207-213.

2. Aerts S, Lambrechts D, Maity S, Van Loo P, Coessens B, De Smet F, Tranchevent L, De Moor B, Marynen P, Hassan B et al: Gene prioritization through genomic data fusion. Nature Biotechnology 2006, 24:537 - 544.

3. Aerts S, Van Loo P, Moreau Y, De Moor B: A genetic algorithm for the detection of new cis-regulatory modules in sets of coregulated genes. Bioinformatics 2004, 20(12):1974-1976.

4. Aerts S, Van Loo P, Thijs G, Moreau Y, De Moor B: Computational detection of cis-regulatory modules. Bioinformatics 2003, 19 Suppl 2:ii5-14.

5. Agrawal R, imielenski T, Swami A: Mining association rules between sets of items in large databases. Proceedings of the ACM SIGMOD Conference on Management of Data (SIGMOD98) 1993:207-216.

6. Alkema WB, Johansson O, Lagergren J, Wasserman WW: MSCAN: identification of functional clusters of transcription factor binding sites. Nucleic Acids Res 2004, 32(Web Server issue):W195-198.

7. Ameur A, Rada-Iglesias A, Komorowski J, Wadelius C: Identification of candidate regulatory SNPs by combination of transcription-factor-binding site prediction, SNP genotyping and haploChIP. Nucleic Acids Res 2009, 37(12):e85.

8. Antequera F: Structure, function and evolution of CpG island promoters. Cell Mol Life Sci 2003, 60(8):1647-1658.

9. Bailey TL, Noble WS: Searching for statistically significant regulatory modules. Bioinformatics 2003, 19 Suppl 2:ii16-25.

10. Bailey TL, Williams N, Misleh C, Li WW: MEME: discovering and analyzing DNA and protein sequence motifs. Nucleic Acids Res 2006, 34(Web Server

issue):W369-373.

11. Bar-Joseph Z, Gerber GK, Lee TI, Rinaldi NJ, Yoo JY, Robert F, Gordon DB, Fraenkel E, Jaakkola TS, Young RA et al: Computational discovery of gene modules and regulatory networks. Nat Biotechnol 2003, 21(11):1337-1342.

12. Barkow S, Bleuler S, Prelic A, Zimmermann P, Zitzler E: BicAT: a biclustering analysis toolbox. Bioinformatics 2006, 22(10):1282-1283.

13. Barrett T, Troup DB, Wilhite SE, Ledoux P, Rudnev D, Evangelista C, Kim IF, Soboleva A, Tomashevsky M, Marshall KA et al: NCBI GEO: archive for high-throughput functional genomic data. Nucleic Acids Res 2009, 37(Database issue):D885-890.

14. Barski A, Cuddapah S, Cui K, Roh TY, Schones DE, Wang Z, Wei G, Chepelev I, Zhao K: High-resolution profiling of histone methylations in the human genome. Cell 2007, 129(4):823-837.

15. Becquet C, Blachon S, Jeudy B, Boulicaut JF, Gandrillon O: Strong-association-rule mining for large-scale gene-expression data analysis: a case study on human SAGE data. Genome Biol 2002, 3(12):RESEARCH0067.

16. Berman BP, Nibu Y, Pfeiffer BD, Tomancak P, Celniker SE, Levine M, Rubin GM, Eisen MB: Exploiting transcription factor binding site clustering to identify cis-regulatory modules involved in pattern formation in the Drosophila genome. Proc Natl Acad Sci U S A 2002, 99(2):757-762.

17. Birney E, Stamatoyannopoulos JA, Dutta A, Guigo R, Gingeras TR, Margulies EH, Weng Z, Snyder M, Dermitzakis ET, Thurman RE et al: Identification and analysis of functional elements in 1% of the human genome by the ENCODE pilot project. Nature 2007, 447(7146):799-816.

18. Bock C, Lengauer T: Computational epigenetics. Bioinformatics 2008, 24(1):1-10.

19. Bock C, Paulsen M, Tierling S, Mikeska T, Lengauer T, Walter J: CpG island methylation in human lymphocytes is highly correlated with DNA sequence, repeats, and predicted DNA structure. PLoS Genet 2006, 2(3):e26.

20. Boiani M, Gentile L, Gambles VV, Cavaleri F, Redi CA, Scholer HR: Variable reprogramming of the pluripotent stem cell marker Oct4 in mouse clones: distinct developmental potentials in different culture environments. Stem Cells 2005, 23(8):1089-1104.

21. Bollen J. Master thesis on Bioinformatics layout and visualization of gene module networksT̃ at ESAT. KULeuven. Belgium. June 2006

22. Borgel J, Guibert S, Li Y, Chiba H, Schubeler D, Sasaki H, Forne T, Weber

M: Targets and dynamics of promoter DNA methylation during early mouse development. Nat Genet 2010, 42(12):1093-1100.

23. Bourillot P, Savatier P: Krupel-like transcription factors and control of pluripotency. BMC Biology 2010, 8(125).

24. Boyle AP, Davis S, Shulha HP, Meltzer P, Margulies EH, Weng Z, Furey TS, Crawford GE: High-resolution mapping and characterization of open chromatin across the genome. Cell 2008, 132(2):311-322.

25. Bray N, Dubchak I, Pachter L: AVID: A global alignment program. Genome Res 2003, 13(1):97-102.

26. Brazma A, Vilo J, Ukkonen E, Valtonen K: Data mining for regulatory elements in yeast genome. Proc Int Conf Intell Syst Mol Biol 1997:65-74.

27. Brudno M, Do CB, Cooper GM, Kim MF, Davydov E, Green ED, Sidow A, Batzoglou S: LAGAN and Multi-LAGAN: efficient tools for large-scale multiple alignment of genomic DNA. Genome Res 2003, 13(4):721-731.

28. Buck MJ, Lieb JD: ChIP-chip: considerations for the design, analysis, and application of genomewide chromatin immunoprecipitation experiments. Genomics 2004, 83(3):349-360.

29. Cawley S, Bekiranov S, Ng HH, Kapranov P, Sekinger EA, Kampa D, Piccolboni A, Sementchenko V, Cheng J, Williams AJ et al: Unbiased mapping of transcription factor binding sites along human chromosomes 21 and 22 points to widespread regulation of noncoding RNAs. Cell 2004, 116(4):499-509.

30. Celniker SE, Dillon LA, Gerstein MB, Gunsalus KC, Henikoff S, Karpen GH, Kellis M, Lai EC, Lieb JD, MacAlpine DM et al: Unlocking the secrets of the genome. Nature 2009, 459(7249):927-930.

31. Chen K, Rajewsky N: The evolution of gene regulation by transcription factors and microRNAs. Nat Rev Genet 2007, 8(2):93-103.

32. Chen X, Xu H, Yuan P, Fang F, Huss M, Vega VB, Wong E, Orlov YL, Zhang W, Jiang J et al: Integration of external signaling pathways with the core transcriptional network in embryonic stem cells. Cell 2008, 133(6):1106-1117.

33. Cheng KO, Law NF, Siu WC, Lau TH: BiVisu: software tool for bicluster detection and visualization. Bioinformatics 2007, 23(17):2342-2344.

34. Chiu SH, Chen CC, Yuan GF, Lin TH: Association algorithm to mine the rules that govern enzyme definition and to classify protein sequences. BMC Bioinformatics 2006, 7:304.

35. Chodavarapu RK, Feng S, Bernatavichute YV, Chen PY, Stroud H, Yu Y, Hetzel JA, Kuo F, Kim J, Cokus SJ et al: Relationship between nucleosome

positioning and DNA methylation. Nature 2010, 466(7304):388-392.

36. Christian S, Guido T: Weakly monotonic propogators. Lecture Notes in Computer Science 2009, 5732:723-730.

37. Coessens B, Thijs G, Aerts S, Marchal K, De Smet F, Engelen K, Glenisson P, Moreau Y, Mathys J, De Moor B: INCLUSive: A web portal and service registry for microarray and regulatory sequence analysis. Nucleic Acids Res 2003, 31(13):3468-3470.

38. Crawford GE, Davis S, Scacheri PC, Renaud G, Halawi MJ, Erdos MR, Green R, Meltzer PS, Wolfsberg TG, Collins FS: DNase-chip: a high-resolution method to identify DNase I hypersensitive sites using tiled microarrays. Nature Methods 2006, 3(7):503-509.

39. Daenen F, van Roy F, De Bleser PJ: Low nucleosome occupancy is encoded around functional human transcription factor binding sites. BMC Genomics 2008, 9:332.

40. Dai Z, Dai X, Xiang Q, Feng J: Nucleosomal context of binding sites influences transcription factor binding affinity and gene regulation. Genomics Proteomics Bioinformatics 2009, 7(4):155-162.

41. Davidson EH: Genomic regulatory systems: development and evolution. San Diego: Academic Press; 2001.

42. De Raedt L, Guns T, Nijssen S: Constraint programming for itemset mining. Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining 2008.

43. Down TA, Hubbard TJ: NestedMICA: sensitive inference of over-represented motifs in nucleic acid sequence. Nucleic Acids Res 2005, 33(5):1445-1453.

44. Duret L, Bucher P: Searching for regulatory elements in human noncoding sequences. Curr Opin Struct Biol 1997, 7(3):399-406.

45. Ernst J, Kellis M: Discovery and characterization of chromatin states for systematic annotation of the human genome. Nat Biotechnol 2010, 28(8):817-825.

46. Ernst J, Plasterer HL, Simon I, Bar-Joseph Z: Integrating multiple evidence sources to predict transcription factor binding in the human genome. Genome Res 2010, 20(4):526-536.

47. Evans PM, Zhang W, Chen X, Yang J, Bhakat KK, Liu C: Kruppel-like factor 4 is acetylated by p300 and regulates gene transcription via modulation of histone acetylation. J Biol Chem 2007, 282(47):33994-34002.

48. Falls JG, Pulford DJ, Wylie AA, Jirtle RL: Genomic imprinting: implications for human disease. Am J Pathol 1999, 154(3):635-647.

49. Felsenfeld G, Groudine M: Controlling the double helix. Nature 2003, 421(6921):448-453.

50. Field Y, Kaplan N, Fondufe-Mittendorf Y, Moore IK, Sharon E, Lubling Y, Widom J, Segal E: Distinct modes of regulation by chromatin encoded through nucleosome positioning signals. PLoS Comput Biol 2008, 4(11):e1000216.

51. Foshay KM, Gallicano GI: Regulation of Sox2 by STAT3 initiates commitment to the neural precursor cell fate. Stem Cells Dev 2008, 17(2):269-278.

52. Frith MC, Fu Y, Yu L, Chen JF, Hansen U, Weng Z: Detection of functional DNA motifs via statistical over-representation. Nucleic Acids Res 2004, 32(4):1372-1381.

53. Frith MC, Hansen U, Weng Z: Detection of cis-element clusters in higher eukaryotic DNA. Bioinformatics 2001, 17(10):878-889.

54. Frith MC, Li MC, Weng Z: Cluster-Buster: Finding dense clusters of motifs in DNA sequences. Nucleic Acids Res 2003, 31(13):3666-3668.

55. Frith MC, Spouge JL, Hansen U, Weng Z: Statistical significance of clusters of motifs represented by position specific scoring matrices in nucleotide sequences. Nucleic Acids Res 2002, 30(14):3214-3224.

56. Fu W, Ray P, Xing EP: DISCOVER: a feature-based discriminative method for motif search in complex genomes. Bioinformatics 2009, 25(12):i321-329.

57. Gallo A, De Bie T, Cristianini N: MINI: Mining informative nonredundant itemset. Proceedings of the 11 th conference on principles and practice of knowledge discovery in databases 2007.

58. Gama-Castro S, Jimenez-Jacinto V, Peralta-Gil M, Santos-Zavaleta A, Penaloza-Spinola MI, Contreras-Moreira B, Segura-Salazar J, Muniz-Rascado L, Martinez-Flores I, Salgado H et al: RegulonDB (version 6.0): gene regulation model of Escherichia coli K-12 beyond transcription, active (experimental) annotated promoters and Textpresso navigation. Nucleic Acids Res 2008, 36(Database issue):D120-124.

59. Gao S, Alarcon C, Sapkota G, Rahman S, Chen PY, Goerner N, Macias MJ, Erdjument-Bromage H, Tempst P, Massague J: Ubiquitin Ligase Nedd4L Targets Activated Smad2/3 to Limit TGF-beta Signaling. Molecular Cell 2009, 36(3):457-468.

60. GecodeTeam: Gecode: Generic constraint development environment. 2006.

61. Georgii E, Richter L, Ruckert U, Kramer S: Analyzing microarray data using quantitative association rules. Bioinformatics 2005, 21 Suppl 2:ii123-129.

62. Giannoukakis N, Deal C, Paquette J, Goodyer CG, Polychronakos C: Parental genomic imprinting of the human IGF2 gene. Nat Genet 1993, 4(1):98-101.

63. Gold JD, Pedersen RA: Mechanisms of genomic imprinting in mammals. Curr Top Dev Biol 1994, 29:227-280.

64. Gotea V, Ovcharenko I: DiRE: identifying distant regulatory elements of co-expressed genes. Nucleic Acids Res 2008, 36(Web Server issue):W133-139.

65. Grant PA: A tale of histone modifications. Genome Biol 2001, 2(4):REVIEWS0003.

66. Grimson A, Farh KK, Johnston WK, Garrett-Engele P, Lim LP, Bartel DP: MicroRNA targeting specificity in mammals: determinants beyond seed pairing. Mol Cell 2007, 27(1):91-105.

67. Grothaus GA, Mufti A, Murali TM: Automatic layout and visualization of biclusters. Algorithms Mol Biol 2006, 1:15.

68. Gu H, Bock C, Mikkelsen TS, Jager N, Smith ZD, Tomazou E, Gnirke A, Lander, ES, Meissner A: Genome-scale DNA methylation mapping of clinical samples at single-nucleotide resolution. Nat Methods 2010, 7(2):133-136.

69. Guelzim N, Bottani S, Bourgine P, Kepes F: Topological and causal structure of the yeast transcriptional regulatory network. Nat Genet 2002, 31(1):60-63.

70. Guenther MG, Levine SS, Boyer LA, Jaenisch R, Young RA: A chromatin landmark and transcription initiation at most promoters in human cells. Cell 2007, 130(1):77-88.

71. Guns T, Sun H, Nijssen S, Marchal K, L. DR: Cis-regulatory module detection using constraint programming. Proceedings of IEEE International Conference on Bioinformatics and Biomedicine 2010.

72. Gupta M, Liu JS: De novo cis-regulatory module elicitation for eukaryotic genomes. Proc Natl Acad Sci U S A 2005, 102(20):7079-7084.

73. Gupta S, Dennis J, Thurman RE, Kingston R, Stamatoyannopoulos JA, Noble WS: Predicting human nucleosome occupancy from primary sequence. PLoS Comput Biol 2008, 4(8):e1000134.

74. Hajkova P, el-Maarri O, Engemann S, Oswald J, Olek A, Walter J: DNA-methylation analysis by the bisulfite-assisted genomic sequencing method. Methods Mol Biol 2002, 200:143-154.

75. Halfon MS, Gallo SM, Bergman CM: REDfly 2.0: an integrated database of cis-regulatory modules and transcription factor binding sites in Drosophila. Nucleic Acids Res 2008, 36(Database issue):D594-598.

76. Hall J, Guo G, Wray J, Eyres I, Nichols J, Grotewold L, Morfopoulou S, Humphreys P, Mansfield W, Walker R et al: Oct4 and LIF/Stat3 additively induce Kruppel factors to sustain embryonic stem cell self-renewal. Cell Stem Cell 2009, 5(6):597-609.

77. Hallikas O, Palin K, Sinjushina N, Rautiainen R, Partanen J, Ukkonen E, Taipale J: Genomewide prediction of mammalian enhancers based on analysis of transcription-factor binding affinity. Cell 2006, 124(1):47-59.

78. Hansen JC, Tse C, Wolffe AP: Structure and function of the core histone N-termini: more than meets the eye. Biochemistry 1998, 37(51):17637-17641.

79. Hartwell LH, Hopfield JJ, Leibler S, Murray AW: From molecular to modular cell biology. Nature 1999, 402(6761 Suppl):C47-52.

80. Heintzman ND, Stuart RK, Hon G, Fu Y, Ching CW, Hawkins RD, Barrera LO, Van Calcar S, Qu C, Ching KA et al: Distinct and predictive chromatin signatures of transcriptional promoters and enhancers in the human genome. Nat Genet 2007, 39(3):311-318.

81. Hertzberg L, Zuk O, Getz G, Domany E: Finding motifs in promoter regions. J Comput Biol 2005, 12(3):314-330.

82. Hu J, Li B, Kihara D: Limitations and potentials of current motif discovery algorithms. Nucleic Acids Res 2005, 33(15):4899-4913.

83. Huang Y, Li H, Hu H, Yan X, Waterman MS, Huang H, Zhou XJ: Systematic discovery of functional modules and context-specific functional annotation of human genome. Bioinformatics 2007, 23(13):i222-229.

84. Hubbard T, Barker D, Birney E, Cameron G, Chen Y, Clark L, Cox T, Cuff J, Curwen V, Down T et al: The Ensembl genome database project. Nucleic Acids Res 2002, 30(1):38-41.

85. Ideraabdullah FY, Vigneau S, Bartolomei MS: Genomic imprinting mechanisms in mammals. Mutat Res 2008, 647(1-2):77-85.

86. Ioshikhes I, Bolshoy A, Derenshteyn K, Borodovsky M, Trifonov EN: Nucleosome DNA sequence pattern revealed by multiple alignment of experimentally mapped sequences. J Mol Biol 1996, 262(2):129-139.

87. Ivan A, Halfon MS, Sinha S: Computational discovery of cis-regulatory modules in Drosophila without prior knowledge of motifs. Genome Biol 2008, 9(1):R22.

88. Ivan G, Szabadka Z, Grolmusz V: Being a binding site: characterizing residue composition of binding sites on proteins. Bioinformation 2007, 2(5):216-221.

89. Ji X, Li W, Song J, Wei L, Liu XS: CEAS: cis-regulatory element annotation system. Nucleic Acids Res 2006, 34(Web Server issue):W551-554.

90. Jiang C, Pugh BF: A compiled and systematic reference map of nucleosome positions across the Saccharomyces cerevisiae genome. Genome Biol 2009, 10(10):R109.

91. Jiang J, Chan YS, Loh YH, Cai J, Tong GQ, Lim CA, Robson P, Zhong S, Ng HH: A core Klf circuitry regulates self-renewal of embryonic stem cells. Nat Cell Biol 2008, 10(3):353-360.

92. Johnson DS, Mortazavi A, Myers RM, Wold B: genomewide mapping of in vivo protein-DNA interactions. Science 2007, 316(5830):1497-1502.

93. Jones PA, Martienssen R: A blueprint for a Human Epigenome Project: the AACR Human Epigenome Workshop. Cancer Res 2005, 65(24):11241-11246.

94. Jothi R, Cuddapah S, Barski A, Cui K, Zhao K: genomewide identification of in vivo protein-DNA binding sites from ChIP-Seq data. Nucleic Acids Res 2008, 36(16):5221-5231.

95. Jurka J, Kapitonov VV, Pavlicek A, Klonowski P, Kohany O, Walichiewicz J: Repbase Update, a database of eukaryotic repetitive elements. Cytogenet Genome Res 2005, 110(1-4):462-467.

96. Kadonaga JT: Eukaryotic transcription: an interlaced network of transcription factors and chromatin-modifying machines. Cell 1998, 92(3):307-313.

97. Kaplan N, Moore IK, Fondufe-Mittendorf Y, Gossett AJ, Tillo D, Field Y, LeProust EM, Hughes TR, Lieb JD, Widom J et al: The DNA-encoded nucleosome organization of a eukaryotic genome. Nature 2009, 458(7236):362-366.

98. Kel A, Konovalova T, Waleev T, Cheremushkin E, Kel-Margoulis O, Wingender E: Composite Module Analyst: a fitness-based tool for identification of transcription factor binding site combinations. Bioinformatics 2006, 22(10):1190-1197.

99. Kel-Margoulis OV, Kel AE, Reuter I, Deineko IV, Wingender E: TRANSCompel: a database on composite regulatory elements in eukaryotic genes. Nucleic Acids Res 2002, 30(1):332-334.

100. Klepper K, Drablos F: PriorsEditor: a tool for the creation and use of positional priors in motif discovery. Bioinformatics 2010.

101. Klepper K, Sandve GK, Abul O, Johansen J, Drablos F: Assessment of composite motif discovery methods. BMC Bioinformatics 2008, 9:123.

102. Koch CM, Andrews RM, Flicek P, Dillon SC, Karaoz U, Clelland GK, Wilcox S, Beare DM, Fowler JC, Couttet P et al: The landscape of histone modifications across 1% of the human genome in five human cell lines. Genome Res 2007, 17(6):691-707.

103. Kouzarides T: Chromatin modifications and their function. Cell 2007, 128(4):693-705.

104. Kouzarides T: Acetylation: a regulatory modification to rival phosphorylation? EMBO J 2000, 19(6):1176-1179.

105. Kratzer I, Wernig K, Panzenboeck U, Bernhart E, Reicher H, Wronski R, Windisch M, Hammer A, Malle E, Zimmer A et al: Apolipoprotein A-I coating of protamine-oligonucleotide nanoparticles increases particle uptake and transcytosis in an in vitro model of the blood-brain barrier. J Control Release 2007, 117(3):301-311.

106. Krivan W, Wasserman WW: A predictive model for regulatory sequences directing liver-specific transcription. Genome Res 2001, 11(9):1559-1566.

107. Lahdesmaki H, Rust AG, Shmulevich I: Probabilistic inference of transcription factor binding from multiple data sources. PLoS One 2008, 3(3):e1820.

108. Laing ME, Cummins R, O'Grady A, O'Kelly P, Kay EW, Murphy GM: Aberrant DNA methylation associated with MTHFR C677T genetic polymorphism in cutaneous squamous cell carcinoma in renal transplant patients. Br J Dermatol 2010, 163(2):345-352.

109. Lee CK, Shibata Y, Rao B, Strahl BD, Lieb JD: Evidence for nucleosome depletion at active regulatory regions genomewide. Nat Genet 2004, 36(8):900-905.

110. Lee JS, Smith E, Shilatifard A: The language of histone crosstalk. Cell 2010, 142(5):682-685.

111. Lee W, Tillo D, Bray N, Morse RH, Davis RW, Hughes TR, Nislow C: A high-resolution atlas of nucleosome occupancy in yeast. Nat Genet 2007, 39(10):1235-1244.

112. Lemmens K, De Bie T, Dhollander T, De Keersmaecker SC, Thijs IM, Schoofs G, De Weerdt A, De Moor B, Vanderleyden J, Collado-Vides J et al: DISTILLER: a data integration framework to reveal condition dependency of complex regulons in Escherichia coli. Genome Biol 2009, 10(3):R27.

113. Lemmens K, Dhollander T, De Bie T, Monsieurs P, Engelen K, Smets B, Winderickx J, De Moor B, Marchal K: Inferring transcriptional modules from ChIP-chip, motif and microarray data. Genome Biol 2006, 7(5):R37.

114. Li E, Beard C, Jaenisch R: Role for DNA methylation in genomic imprinting. Nature 1993, 366(6453):362-365.

115. Li HX, Han M, Bernier M, Zheng B, Sun SG, Su M, Zhang R, Fu JR, Wen JK: Kruppel-like factor 4 promotes differentiation by transforming growth factor-beta receptor-mediated Smad and p38 MAPK signaling in vascular smooth muscle cells.

J Biol Chem 2010, 285(23):17846-17856.

116. Liu ET, Pott S, Huss M: QA: ChIP-seq technologies and the study of gene regulation. BMC Biol 2010, 8:56.

117. Liu XS, Brutlag DL, Liu JS: An algorithm for finding protein-DNA binding sites with applications to chromatin-immunoprecipitation microarray experiments. Nat Biotechnol 2002, 20(8):835-839.

118. Liu Y, Balaraman Y, Wang G, Nephew KP, Zhou FC: Alcohol exposure alters DNA methylation profiles in mouse embryos at early neurulation. Epigenetics 2009, 4(7):500-511.

119. Loh YH, Wu Q, Chew JL, Vega VB, Zhang W, Chen X, Bourque G, George J, Leong B, Liu J et al: The Oct4 and Nanog transcription network regulates pluripotency in mouse embryonic stem cells. Nat Genet 2006, 38(4):431-440.

120. Lopez FJ, Blanco A, Garcia F, Cano C, Marin A: Fuzzy association rules for biological data analysis: a case study on yeast. BMC Bioinformatics 2008, 9:107.

121. Macintyre G, Bailey J, Haviv I, Kowalczyk A: is-rSNP: a novel technique for in silico regulatory SNP detection. Bioinformatics 2010, 26(18):i524-530.

122. Madeira SC, Oliveira AL: Biclustering algorithms for biological data analysis: a survey. IEEE/ACM Trans Comput Biol Bioinform 2004, 1(1):24-45.

123. Maherali N, Sridharan R, Xie W, Utikal J, Eminli S, Arnold K, Stadtfeld M, Yachechko R, Tchieu J, Jaenisch R et al: Directly reprogrammed fibroblasts show global epigenetic remodeling and widespread tissue contribution. Cell Stem Cell 2007, 1(1):55-70.

124. Maragkakis M, Reczko M, Simossis VA, Alexiou P, Papadopoulos GL, Dalamagas T, Giannopoulos G, Goumas G, Koukis E, Kourtis K et al: DIANA-microT web server: elucidating microRNA functions through target prediction. Nucleic Acids Res 2009, 37(Web Server issue):W273-276.

125. Marks P, Rifkind RA, Richon VM, Breslow R, Miller T, Kelly WK: Histone deacetylases and cancer: causes and therapies. Nat Rev Cancer 2001, 1(3):194-202.

126. Martin P, Makepeace K, Hill SA, Hood DW, Moxon ER: Microsatellite instability regulates transcription factor binding and gene expression. Proc Natl Acad Sci U S A 2005, 102(10):3800-3804.

127. Matys V, Kel-Margoulis OV, Fricke E, Liebich I, Land S, Barre-Dirrie A, Reuter I, Chekmenev D, Krull M, Hornischer K et al: TRANSFAC and its module TRANSCompel: transcriptional gene regulation in eukaryotes. Nucleic Acids Res 2006, 34(Database issue):D108-110.

128. Mavrich TN, Ioshikhes IP, Venters BJ, Jiang C, Tomsho LP, Qi J, Schuster

SC, Albert I, Pugh BF: A barrier nucleosome model for statistical positioning of nucleosomes throughout the yeast genome. Genome Res 2008, 18(7):1073-1083.

129. Meissner A, Mikkelsen TS, Gu H, Wernig M, Hanna J, Sivachenko A, Zhang X, Bernstein BE, Nusbaum C, Jaffe DB et al: Genome-scale DNA methylation maps of pluripotent and differentiated cells. Nature 2008, 454(7205):766-770.

130. Mikkelsen TS, Ku M, Jaffe DB, Issac B, Lieberman E, Giannoukos G, Alvarez P, Brockman W, Kim TK, Koche RP et al: Genomewide maps of chromatin state in pluripotent and lineage-committed cells. Nature 2007, 448(7153):553-560.

131. Miller W, Rosenbloom K, Hardison RC, Hou M, Taylor J, Raney B, Burhans R, King DC, Baertsch R, Blankenberg D et al: 28-way vertebrate alignment and conservation track in the UCSC Genome Browser. Genome Res 2007, 17(12):1797-1808.

132. Mohn F, Weber M, Schubeler D, Roloff TC: Methylated DNA immunoprecipitation (MeDIP). Methods Mol Biol 2009, 507:55-64.

133. Morgan XC, Ni S, Miranker DP, Iyer VR: Predicting combinatorial binding of transcription factors to regulatory elements in the human genome by association rule mining. BMC Bioinformatics 2007, 8:445.

134. Nykter M, Lahdesmaki H, Rust A, Thorsson V, Shmulevich I: A data integration framework for prediction of transcription factor targets. Ann N Y Acad Sci 2009, 1158:205-214.

135. Odom DT, Dowell RD, Jacobsen ES, Gordon W, Danford TW, MacIsaac KD, Rolfe PA, Conboy CM, Gifford DK, Fraenkel E: Tissue-specific transcriptional regulation has diverged significantly between human and mouse. Nat Genet 2007, 39(6):730-732.

136. Okada Y, Yamagata K, Hong K, Wakayama T, Zhang Y: A role for the elongator complex in zygotic paternal genome demethylation. Nature 2010, 463(7280):554-558.

137. Ooi SK, Qiu C, Bernstein E, Li K, Jia D, Yang Z, Erdjument-Bromage H, Tempst P, Lin SP, Allis CD et al: DNMT3L connects unmethylated lysine 4 of histone H3 to de novo methylation of DNA. Nature 2007, 448(7154):714-717.

138. Osada R, Zaslavsky E, Singh M: Comparative analysis of methods for representing and searching for transcription factor binding sites. Bioinformatics 2004, 20(18):3516-3525.

139. Oyama T, Kitano K, Satou K, Ito T: Extraction of knowledge on protein-protein interaction by association rule discovery. Bioinformatics 2002, 18(5):705-714.

140. Ozsolak F, Song JS, Liu XS, Fisher DE: High-throughput mapping of the chromatin structure of human promoters. Nat Biotechnol 2007, 25(2):244-248.

141. Peckham HE, Thurman RE, Fu Y, Stamatoyannopoulos JA, Noble WS, Struhl K, Weng Z: Nucleosome positioning signals in genomic DNA. Genome Res 2007, 17(8):1170-1177.

142. Pepke S, Wold B, Mortazavi A: Computation for ChIP-seq and RNA-seq studies. Nat Methods 2009, 6(11 Suppl):S22-32.

143. Pham TH, Satou K, Ho TB: Mining yeast transcriptional regulatory modules from factor DNA-binding sites and gene expression data. Genome Inform 2004, 15(2):287-295.

144. Philippakis AA, He FS, Bulyk ML: Modulefinder: a tool for computational discovery of cis-regulatory modules. Pac Symp Biocomput 2005:519-530.

145. Rabinowitz JE, Rutishauser U, Magnuson T: Targeted mutation of Ncam to produce a secreted molecule results in a dominant embryonic lethality. Proc Natl Acad Sci U S A 1996, 93(13):6421-6424.

146. Rajewsky N, Vergassola M, Gaul U, Siggia ED: Computational detection of genomic cis-regulatory modules applied to body patterning in the early Drosophila embryo. BMC Bioinformatics 2002, 3:30.

147. Ramsey SA, Knijnenburg TA, Kennedy KA, Zak DE, Gilchrist M, Gold ES, Johnson CD, Lampano AE, Litvak V, Navarro G et al: Genomewide histone acetylation data improve prediction of mammalian transcription factor binding sites. Bioinformatics 2010, 26(17):2071-2075.

148. Richmond TJ, Davey CA: The structure of DNA in the nucleosome core. Nature 2003, 423(6936):145-150.

149. Rosenbloom KR, Dreszer TR, Pheasant M, Barber GP, Meyer LR, Pohl A, Raney BJ, Wang T, Hinrichs AS, Zweig AS et al: ENCODE whole-genome data in the UCSC Genome Browser. Nucleic Acids Res 2009, 38(Database issue):D620-625.

150. Royce TE, Rozowsky JS, Bertone P, Samanta M, Stolc V, Weissman S, Snyder M, Gerstein M: Issues in the analysis of oligonucleotide tiling microarrays for transcript mapping. Trends Genet 2005, 21(8):466-475.

151. Sakabe NJ, Nobrega MA: Genomewide maps of transcription regulatory elements. Wiley Interdiscip Rev Syst Biol Med 2010, 2(4):422-437.

152. Sandelin A, Alkema W, Engstrom P, Wasserman WW, Lenhard B: JASPAR: an open-access database for eukaryotic transcription factor binding profiles. Nucleic Acids Res 2004, 32(Database issue):D91-94.

153. Sandve GK, Abul O, Drablos F: Compo: composite motif discovery using discrete models. BMC Bioinformatics 2008, 9:527.

154. Santamaria R, Theron R, Quintales L: BicOverlapper: a tool for bicluster visualization. Bioinformatics 2008, 24(9):1212-1213.

155. Sebestyen E, Nagy T, Suhai S, Barta E: DoOPSearch: a web-based tool for finding and analysing common conserved motifs in the promoter regions of different chordate and plant genes. BMC Bioinformatics 2009, 10 Suppl 6:S6.

156. Shamir R, Maron-Katz A, Tanay A, Linhart C, Steinfeld I, Sharan R, Shiloh Y, Elkon R: EXPANDER-an integrative program suite for microarray data analysis. BMC Bioinformatics 2005, 6:232.

157. Sharan R, Ovcharenko I, Ben-Hur A, Karp RM: CREME: a framework for identifying cis-regulatory modules in human-mouse conserved segments. Bioinformatics 2003, 19 Suppl 1:i283-291.

158. Shin H, Liu T, Manrai AK, Liu XS: CEAS: cis-regulatory element annotation system. Bioinformatics 2009, 25(19):2605-2606.

159. Shivaswamy S, Bhinge A, Zhao Y, Jones S, Hirst M, Iyer VR: Dynamic remodeling of individual nucleosomes across a eukaryotic genome in response to transcriptional perturbation. PLoS Biol 2008, 6(3):e65.

160. Shobhit Gupta JAS, Timothy L Bailey and William S Noble: Quantifying similarity between motifs. Genome Biol 2007, 8(R24).

161. Sinha S, Liang Y, Siggia E: Stubb: a program for discovery and analysis of cis-regulatory modules. Nucleic Acids Res 2006, 34(Web Server issue):W555-559.

162. Smith GH, Chepko G: Mammary epithelial stem cells. Microsc Res Tech 2001, 52(2):190-203.

163. Smith ZD, Gu H, Bock C, Gnirke A, Meissner A: High-throughput bisulfite sequencing in mammalian genomes. Methods 2009, 48(3):226-232.

164. Stormo GD: DNA binding sites: representation and discovery. Bioinformatics 2000, 16(1):16-23.

165. Strahl BD, Allis CD: The language of covalent histone modifications. Nature 2000, 403(6765):41-45.

166. Straussman R, Nejman D, Roberts D, Steinfeld I, Blum B, Benvenisty N, Simon I, Yakhini Z, Cedar H: Developmental programming of CpG island methylation profiles in the human genome. Nat Struct Mol Biol 2009, 16(5):564-571.

167. Sun H, De Bie T, Storms V, Fu Q, Dhollander T, Lemmens K, Verstuyf A, De Moor B, Marchal K: ModuleDigger: an itemset mining framework for the detection

of cis-regulatory modules. BMC Bioinformatics 2009, 10 Suppl 1:S30.

168. Sun H, Lemmens K, Van den Bulcke T, Engelen K, De Moor B, Marchal K. (2009). ViTraM: Visualization of Transcriptional Modules. Bioinformatics, 25(18):2450-2451;

169. Sun H, Lemmens K, Van den Bulcke T, Engelen K, De Moor B, Marchal K. (2009). Layout and Post-Processing of Transcriptional Modules. Proc. International Joint Conference on Bioinformatics, Systems Biology and Intelligent Computing (IJCBS2009), IEEE computer society, 10.1109/IJCBS.2009.95, page:116-121.

170. Sun H, Guns T, Fierro AC, Thorrez L, Nijseen S, Marchal K. (2011). Unveiling combinatorial regulation through the combination of ChIP information and *in silico cis*-regulatory module detection. *In revision.*

171. Sun H, Storms V, Meysman P, Marchal K. (2011). The past and future trends of cis-regulatory module screening, from DNA sequence based to multi-evidence based. *In preparation.*

172. Suzuki MM, Bird A: DNA methylation landscapes: provocative insights from epigenomics. Nat Rev Genet 2008, 9(6):465-476.

173. Tamura M, D'Haeseleer P: Microbial genotype-phenotype mapping by class association rule mining. Bioinformatics 2008, 24(13):1523-1529.

174. Thomas DJ, Rosenbloom KR, Clawson H, Hinrichs AS, Trumbower H, Raney BJ, Karolchik D, Barber GP, Harte RA, Hillman-Jackson J et al: The ENCODE Project at UC Santa Cruz. Nucleic Acids Res 2007, 35(Database issue):D663-667.

175. Tillo D, Kaplan N, Moore IK, Fondufe-Mittendorf, Yvonne and Gossett, Andrea J. and Field, Yair and Lieb, Jason D, Widom J, Segal E et al: High Nucleosome Occupancy Is Encoded at Human Regulatory Sequences. PLoS ONE 2010, 5(2):e9129.

176. Tompa M, Li N, Bailey TL, Church GM, De Moor B, Eskin E, Favorov AV, Frith MC, Fu Y, Kent WJ et al: Assessing computational tools for the discovery of transcription factor binding sites. Nat Biotechnol 2005, 23(1):137-144.

177. Tony K: Chromatin Modifications and Their Function. Cell 2007, 128(4):693-705.

178. Turi A, Loglisci C, Salvemini E, Grillo G, Malerba D, D'Elia D: Computational annotation of UTR cis-regulatory modules through Frequent Pattern Mining. BMC Bioinformatics 2009, 10 Suppl 6:S25.

179. Tusher VG, Tibshirani R, Chu G: Significance analysis of microarrays applied to the ionizing radiation response. Proc Natl Acad Sci U S A 2001, 98(9):5116-5121.

180. Ucar D, Beyer A, Parthasarathy S, Workman CT: Predicting functionality

of protein-DNA interactions by integrating diverse evidence. Bioinformatics 2009, 25(12):i137-144.

181. Valouev A, Johnson DS, Sundquist A, Medina C, Anton E, Batzoglou S, Myers RM, Sidow A: Genomewide analysis of transcription factor binding sites based on ChIP-Seq data. Nat Methods 2008, 5(9):829-834.

182. Van Holde KE: Chromatin. Springer, New York 1989.

183. Van Loo P, Aerts S, Thienpont B, De Moor B, Moreau Y, Marynen P: ModuleMiner - improved computational detection of cis-regulatory modules: are there different modes of gene regulation in embryonic development and adult tissues? Genome Biol 2008, 9(4):R66.

184. Van Loo P, Marynen P: Computational methods for the detection of cis-regulatory modules. Brief Bioinform 2009, 10(5):509-524.

185. Vettese-Dadey M, Grant PA, Hebbes TR, Crane- Robinson C, Allis CD, Workman JL: Acetylation of histone H4 plays a primary role in enhancing transcription factor binding to nucleosomal DNA in vitro. EMBO J 1996, 15(10):2508-2518.

186. Vinogradov AE: Dualism of gene GC content and CpG pattern in regard to expression in the human genome: magnitude versus breadth. Trends Genet 2005, 21(12):639-643.

187. Whitington T, Perkins AC, Bailey TL: High-throughput chromatin information enables accurate tissue-specific prediction of transcription factor binding sites. Nucleic Acids Res 2009, 37(1):14-25.

188. Xi LQ, Fondufe-Mittendorf Y, Xia L, Flatow J, Widom J, W JP: Predicting nucleosome positioning using a duration Hidden Markov Model. BMC Bioinformatics 2010, 11(346).

189. Xie D, Cai J, Chia NY, Ng HH, Zhong S: Cross-species de novo identification of cis-regulatory modules with GibbsModule: application to gene regulation in embryonic stem cells. Genome Res 2008, 18(8):1325-1335.

190. Zaki MJ, Hsiao CJ: CHARM: An Efficient Algorithm for Closed Itemset Mining. Proceedings Second SIAM International Conference on Data Mining 2002.

191. Zhou Q, Wong WH: CisModule: de novo discovery of cis-regulatory modules by hierarchical mixture modeling. Proc Natl Acad Sci U S A 2004, 101(33):12114-12119.

List of Publications

Sun, H., De Bie, T., Storms, V., Fu, Q., Dhollander, T., Lemmens, K., Verstuyf, A., De Moor, B., Marchal, K. (2009). ModuleDigger: an itemset mining framework for the detection of *cis*-regulatory modules. *BMC Bioinformatics*, 10(Suppl 1):S30; doi:10.1186/1471-2105-10-S1-S30.

Sun, H., Lemmens, K., Van den Bulcke, T., Engelen, K., De Moor, B., Marchal, K. (2009). ViTraM: visualization of transcriptional modules. *Bioinformatics*, 25(18):2450-2451; doi:10.1093/Bioinformatics/btp400.

Sun, H., Lemmens, K., Van den Bulcke, T., Engelen, K., De Moor, B., Marchal, K. (2009). Layout and post-processing of transcriptional modules. *IEEE Computer Society*, 10.1109/IJCBS.2009.95, p116-121.

Sun, H., Guns, T., Fierro, AC., Thorrez, L., Nijseen, S., Marchal, K. (2011). Unveiling combinatorial regulation through the combination of ChIP information and *in silico cis*-regulatory module detection. *In revision.*

Guns, T., Sun, H., Marchal, K., Nijssen S. (2010). *Cis*-regulatory module detection using constraint programming. *IEEE Computer Society*, 10.1109/BIBM.2010.5706592, p363-368.

Fu, Q., Lemmens, K., Thijs, I., Meysman, P., Sanchez, A., Sun, H., Fierro, C., Engelen, K., Marchal, K. (2011). Directed module detection in a large-scale expression compendium. In: Van Helden J., Toussaint A., Thieffry D. (Eds.), *Methods in Molecular Biology*, New York: Springer New York.

In preparation

Sun, H., Storms, V., Meysman, P., Marchal, K. (2011). The past and future trends of *cis*-regulatory module screening, from DNA sequence based to multi-evidence based.

Claeys, M., Storms, V., Sun, H., Marchal, K. (2011). MotifSuite: work flow for regulatory motif detection with various motif assessment tools.

Arenberg Doctoral School of Science, Engineering & Technology
Faculty of Engineering
Department of Electrical Engineering
Research group SCD/SISTA/BIOI
Address Kasteelpark Arenberg 10, B-3001 Leuven (Belgium)

KATHOLIEKE UNIVERSITEIT
LEUVEN