UNIVERSITÀ
DEGLI STUDI
DI PADOVA

Sede Amministrativa: Università degli Studi di Padova

Dipartimento di Ingegneria dell'Informazione

DOTTORATO DI RICERCA IN

INGEGNERIA DELL'INFORMAZIONE

CICLO XXV

Mining Biological Networks

Supervisore: Prof. Gianna Toffolo

Co-supervisore: Prof. Concettina Guerra

Dottorando: Marco Mina

28 GENNAIO 2013

UNIVERSITÀ
DEGLI STUDI
DI PADOVA

Sede Amministrativa: Università degli Studi di Padova

Dipartimento di Ingegneria dell'Informazione

SCUOLA DI DOTTORATO DI RICERCA IN: Ingegneria dell'Informazione
INDIRIZZO: Scienza e Tecnologia dell'Informazione
CICLO: XXV

Mining Biological Networks

Direttore della Scuola: Ch.mo Prof. Matteo Bertocco
Supervisore: Ch.mo Prof. Gianna Toffolo
Coordinatore di indirizzo: Ch.mo Prof. Carlo Ferrari

Dottorando: Marco Mina

Abstract

This thesis addresses relevant issues related to the analysis of biological networks. Path redundancy was exploited to denoise currently available data, dominated by high levels of wrong or missing information, and applied to the local alignment of protein-protein interaction networks. On another research direction, regulatory networks were employed to explain master regulators' ability of modulating cells' behaviour. In this direction, an existing approach was adapted for the analysis of miRNAs' role in Glioblastoma Multiforme cancer cells. The methodological aspects of this work represent an improvement of the foundamental elements of network analysis techniques.

Sommario

In questa tesi sono stati affrontati alcuni aspetti problematici legati all'analisi di reti biologiche. Concentrandosi sull'allineamento di reti di interazione proteica, è stato studiato l'uso di metodi basati sul ”path redundancy“ per filtrare il rumore attualmente presente nei dati. In una seconda direzione di ricerca le reti di regolazione sono state sfruttate per spiegare la capacità dei ”master regulators“ di modulare il comportamento delle cellule, adattando di una pipeline d'analisi allo studio degli effetti dei miRNA nel Glioblastoma Multiforme. I progressi metodologici introdotti in questo lavoro contengono potenziali miglioramenti ad alcuni elementi comuni ad altre tecniche di analisi di reti.

Contents

CHAPTER 1

Introduction

Up to few years ago, when little was known about cellular machineries, research efforts focused on the full but isolate understanding of their single components. The amount of knowledge gained in these studies and the emergence of more advanced techniques to quantify the action of each molecule has recently transformed this scenario. We have in fact the opportunity of looking at biological data from an higher perspective, considering the combinatorial effects of the single components. Indeed, a better understanding of biology cannot overlook this aspect: biomolecules rarely work alone, but are intertwined in a network of interactions. For instance, proteins interact with each other, often binding in functional complexes. Furthermore, the cellular machinery is tightly controlled at different levels (i.e. transcriptional, post-transcriptional, post-translational) by the combined action of regulatory molecules. Such regulators work synegistically to coordinate the global behaviour of cells, driving them toward specific phenotypes, observed for instance during cell differentiation and replication.

In the last two decades, this new perspective encouraged the development of experimental techniques for the determination of the set of molecular interactions, collected in several growing databases publicly available. Graphs are a natural representation for these data, and the growth of biological information is fostering a multiplicity of network-centric investigations. Analysing molecules' behaviour in this context adds a new dimension to the understanding of the cellular machinery, since it exposes the combinatorial effects otherwise not observable when considering single players alone [1] [2] [3] [4]. Transcriptional and post-transcriptional networks, for instance, have been used in the analysis of the effects of drug combinations, and in the investigation of resistance phenomena. Evolutionary studies, on the other hand, have been extended to an higher level by exploiting protein-protein interaction networks, and have the potential to unravel interaction patterns common to different species.

Despite the attention received, there are some issues that might discourage the use of network-centric approaches. First, in many cases reliable interaction data are lacking. Protein-protein interaction networks, for instance, are affected by high levels of noise, and their use requires the development of methodologies able to filter false and missing interactions. In other cases networks are not specific and fail to consistently represent a particular disease or cellular condition, describing instead a rough organization too general for supporting some analyses. Recently some algorithms have been proposed to infer context-specific regulatory networks from gene expression data.

This thesis addresses some of the relevant issues related to the analysis of biological networks, and focuses on two different biological problems.

The first problem, presented in Chapter 2, is that of aligning protein-protein interaction networks. The final purpose is to uncover the functional components that have been conserved in different species across evolution. The resulting pipeline, AlignMCL, is scalable on the size of proteomes, and introduces a mature strategy to denoise input networks.

The second problem, discussed in Chapter 3, is the application of Master Regulator Analysis for the identification of key miRNAs in the regulatory program of Glioblastoma Multiforme (GBM). The analysis has the potential to increase the actual comprehension of GBM, uncovering in particular the collaborative ability of some molecules to mantain specific phenotypes. This study promises intriguing developments, some of which would exploit biological networks to a deeper level.

CHAPTER 2

Network Alignment

2.1 Introduction

Proteins play their biological role by interacting among them. The study of the whole set of protein-protein interactions (PPI), also known as interactome, is becoming an important research area [1, 2, 5, 6, 3, 4]. Analyzing protein behaviors in this context adds a new dimension to the understanding of the cellular machinery, since it exposes the combinatorial effects otherwise not observable when considering single proteins alone.

The availability of new technologies has led to the accumulation of large amounts of interaction data, creating the demand of automated analysis methods. Formalism from graph theory provides the best framework to represent and analyze PPI data [7]. A protein-protein interaction network (PIN) is a graph $G = \{V, E\}$, where V is a set of labeled nodes representing proteins, and E is a set of edges representing the interactions between the proteins. The inspection of PINs aims to elucidate the relation among topological and biological properties. For instance, small dense regions, i.e. regions with an high number of interactions, often represent sets of mutually interacting proteins, namely protein complexes [8, 9, 10].

In previous decades researchers have focused on the impact of the evolution at the genomic scale, i.e. how to reconstruct evolution by analysing genomic sequences. A significant result has been the identification of ortholog proteins, that is groups of evolutionary related proteins descending from the same ancestor.

More recently, the availability of high-throughput data on protein-protein interactions allowed to look at evolutionary changes by comparing the PINs of different species. Such analysis, known as **PPI network alignment**, is the counterpart on PINs of what sequence alignment is for gene and protein primary sequences. Goals of this field include the identification of conserved patterns of interactions among species as well as the identification of novel orthologs. The rationale is that homologous proteins working in conserved mechanisms (i.e. protein complexes and pathways) should present similar interaction patterns in the different species.

There are two different instances of the alignment problem: the **local network alignment** tries to find relatively small similar subnetworks that are likely to represent conserved functional components, while the **global network alignmnent** looks for the best superimposition of the whole input networks (i.e. the alignment that minimizes a cost function). More formally, given two input graphs, G1 = {V1 , E1 } and G2 = {V2 , E2 }, the task of aligning globally G1 and G2 can be formulated as the problem of finding a mapping $M\ :\ \{V_1^* \longrightarrow V_2^*,\ V_1^* \subseteq V_1,\ V_2^* \subseteq V2\}$ that maximizes an associated cost function defined on nodes and edges. From a biological perspective the global alignment answers an evolutionary question searching for a single comprehensive mapping of the whole set of protein interactions from different species, while local alignment searches for evolutionary conserved building blocks of the cellular machinery, disregarding the overall similarity between the networks. This work focuses on the latter problem.

2.1.1 Local network alignment

The general computational problem addressed by the local network alignment is to identify subgraphs of two or more PINs with **similar** and **meaningful** interaction patterns. In a more formal way, given two input graphs, $G_1 = \{V_1, E_1\}$ and $G_2 = \{V_2, E_2\}$, the problem of locally aligning G_1 and G_2 consists of finding sets of node pairs

$$S_i = \{(x, y),\ x \in V_1,\ y \in V_2\}$$

that satisfy or maximize the following criteria:

- (similarity criterion) the subgraph induced on G_1 by $S_i^1 = \{x \mid (x, y) \in S_i\}$ is similar to the subgraph induced on G_2 by $S_i^2 = \{y \mid (x, y) \in S_i\}$, according to some criterion

- (quality criterion) the induced subgraphs show meaningful interactions patterns

It's important that both the requirements are satisfied. Two regions of two networks might be highly similar by chance, but the common topology might not be meaningful.

Several criteria have been proposed, their formulation driven by computational paradigms, the underpinning biological rationale, and the quality of available data.

The simplest similarity criterion corresponds to the exact graph isomorfism test. Similarly to the exact pattern matching problem in the sequence alignment framework, this criterion requires that the induced subgraphs of S_i show the same topology, that is the interactions between the respective orthologs are either present or absent in both G_1 and G_2. More formally:

$$(x_i, x_j) \in E_1 \Leftrightarrow (y_i, y_j) \in E_2 \quad \forall (x_i, y_i), (x_j, y_j) \in S_i$$

However, this formulation is too strict to uncover most of the conserved subgraphs. Due to the scarce and non-uniform knowledge about protein interactions [11, 12], current PINs present high values of missing or wrong interactions. A more relaxed criterion allows gaps and mismatches between the induced subgraphs, assigning to S_i a score given by the number of conserved interactions and mismatches. Indeed, existing algorithms deal with the missing interactions introducing less restrictive similarity criteria, i.e. by verifying whether the corresponding proteins are at distance less than or equal to k in the original PINs, instead of checking only for direct interactions. Beyond the noise there are also biological motivations for employing flexible criteria. In some cases a (small) component of a conserved complex might differ between different species. For instance a direct interaction in one organism might work through an additional bridge protein in another. A rigid criterion would be unable to deal with this situation.

Event though the similarity criterion generally drives the alignment, the quality criterion should be considered as well to determine whether regions are interesting or not. Quality criteria are usually formalized starting from biological motivations. There are different biological questions underpinning the local network alignment problem, since there are different interesting structures that might have been conserved across evolution. For instance, many proteins perform their tasks by merging into protein complexes. It has been suggested that protein complexes should be represented in PINs by densely connected subgraphs. This consideration naturally leads to a quality criterion that requires the induced subgraphs to be relatively dense [13].

Related Work

Beyond the different criteria used to guide the alignment, the network alignment problem proves to be extremely complex in the general case [28]. Different heuristics have been proposed to align two or more networks under specific conditions to find (i) conserved linear paths [15], (ii) conserved highly connected regions [13, 17, 29], and (iii) conserved modules of arbitrary topology [14, 7, 30, 27]. These approaches usually identify relatively small sets of protein pairs that minimize an ad hoc cost function based on the similarity of the interaction patterns of the putative orthologs. The cost functions often embed an a priori node similarity generally derived from protein sequence alignment. This choice is biologically sound, since proteins with high sequence similarity are likely to be functionally related. The existing algorithms try to establish a trade-off between the information derived from the topology of the PINs and homology data provided by sequence alignment. Few algorithms have been designed to work entirely on topological data, and generally produce global alignments [22, 23, 24]. Most notable differences resides on the required input data (e.g. PINs, BLAST e-values, a priori homologies), the structure of the alignment graph, the mining heuristic, and the post-processing steps. An extensive synopsis on available algorithms for both global and local network alignment is provided in Table 2.1.

Existing approaches to detect protein complexes are generally based on the

Algorithm	*Local*(L) / *Global*(G)	*Pairwise*(P) / *Multiwise*(M)	*Input Data*	*Alignment Strategy**
Mawish [14]	L	P	PPI Networks BLAST e-values	Alignment Graph Single node expansion Duplication-divergence model
PathBLAST [15]	L	P	PPI Networks BLAST e-values	Alignment Graph Single node expansion Conserved linear path extraction
NetworkBLAST [16]	L	P	PPI Networks BLAST e-values	Alignment Graph Score for PPI reliability Single node expansion Conserved dense networks extraction
NetworkBLAST-M [17]	L	M	PPI Networks BLAST e-values	Layered Alignment Graph Single node expansion Conserved dense networks extraction
Graemlin [7]	L	M	PPI Networks Cluster of Orthologs	Probability model to score nodes and edges Nodes equivalence classes Single node expansion
Graemlin 2.0 [18]	G/L	M	PPI Networks KEGG Clusters Known Alignment	Machine learning approach for network scoring Single node expansion
ISORANK [19, 20]	G	P	PPI Networks BLAST e-values	Eigenvector of protein pair associations Consistent set of associations extraction
ISORANK-N [21]	G	M	PPI Networks BLAST e-values	Greedy extension of ISORANK
GRAAL [22] (see also [23, 24])	G	P	PPI Networks BLAST e-values	Purely topology based Protein pairs scored based on graphlet signature
HopeMap [25]	L	M	PPI Networks BLAST e-values Inparanoid Clusters KEGG Clusters	Cluster of orthologs Alignment Graph Strong connected component extraction
PHUNKEE [26]	L	P	PPI Networks Metabolic networks BLAST e-values, COG	Expansion process with addition of neighboring modules
NetAligner [27]	L	P	PPI Networks BLAST alignments	Interaction Conservation probabilities

*All methods, as a last step, score and rank the solutions according to a similarity function.

Table 2.1: A synopsis on network alignment tools.

observation that complexes correspond to highly interacting sets of proteins and therefore they look for dense subgraphs in PPI networks. For instance, both versions of NetworkBLAST [13, 17, 29] are based on such hypothesis, evolving from the initial PathBLAST [15] that focused on conserved paths. The method Mawish [14] addresses network alignment as a maximum weight induced subgraph problem, incorporating evolutionary models to assess topological similarity. While effective, this model may be too strict leading to small conserved structures, failing in recovering larger complexes. Other algorithms such as Graemlin [7] and its new version Graemlin 2.0 [18] generalize the previous approach by allowing the search of more general topologies. These methods increase the ability of detecting meaningful alignments by using, in addition to orthology information, paralogy relations between proteins from Inparanoid [31], KEGG pathway annotations, and known alignments.

Alignment Graph based techniques

Many local alignment algorithms are conceptually similar in that they follow the same paradigm: instead of considering the original graphs separately, first they merge all the input data together in a single weighted undirected graph, generally referred to as **alignment graph**, and then apply a mining heuristic on top of the alignment graph. The nodes of an alignment graph correspond to pairs of putative orthologs, and its edges represent potentially conserved interactions. Ideally, according to this definition, two nodes of the alignment graph should be connected only if the corresponding proteins in the two PINs are interacting. While the topology is informative, it has been shown to be often incomplete and reflecting a non-uniform knowledge over proteins [11, 12]. The presence of several false negatives leads to sparse graphs and even sparser alignment graphs, and this may cause approaches looking only for dense subgraphs to fail to detect conserved complexes. Existing algorithms based on alignment graphs deal with the missing interactions introducing less restrictive definitions of alignment graph, i.e. by allowing nodes to be connected when the corresponding proteins are at distance less than or equal to k in the original PINs (e.g. for NetworkBLAST $k = 3$). If not carefully tuned this approach might introduce unreliable links in the alignment graph, leading to incorrect solutions even for small values of k. This effect is likely to increase as the available PINs get more and more complete (since increasing the number of edges increases the number of proteins close to each other in the PIN).

The idea behind the alignment graph is to merge all the input data (PPI networks, putative homologies, ...) in a single graph that can subsequently be mined with classical graph mining strategies. This approach somehow simplifies the problem by reducing it to the analysis of a single graph. However, its value is not limited to that. All the different definitions of alignment graph are based on the idea of assigning a weight to the putative pairs of orthologs depending on the similarity of their neighbourhoods in the respective PPI networks. The purpose of this is to expose the common structure of the different graphs discarding the components that do not show any correspondence.

The dimension of PINs in terms of interactions has tremendously increased in the last few years, and not all the algorithms are able to handle them. For instance, NetworkBlast required more than a week to evaluate the solutions of some alignments on a cluster equipped with Intel Xeon processors.

Contributions

This thesis addresses the major issues reported in the previous section and proposes two new alignment strategies, implemented in the algorithms AlignNemo and AlignMCL. The major contributions consist of a more general and robust definition of alignment graph and an extended assessment pipeline. As previously stated, the purpose of local alignment is to identify modules or complexes that are conserved in PPI networks, i.e. connected and locally similar subgraphs from the input graphs. The following sections describe the work thoroughly following the chronological order of its development.

2.2 AlignNemo

AlignNemo (Aligning Network Modules) introduces a new model of alignment graph that exploits the full extent of interaction data, and features an iterative expansion procedure that explores the local topology of the alignment graph at each step beyond direct interactions. This combination provides a new way to account for topology, and proved effective in detecting a large variety of protein complexes independently of their size or degree of connectivity.

2.2.1 Design and Implementation

The search for conserved modules is performed on the alignment graph and consists of three major steps, as outlined in Figure 2.1.

- First, the alignment graph is constructed from the input networks. Each node in the alignment graph corresponds to a pair of putative orthologous

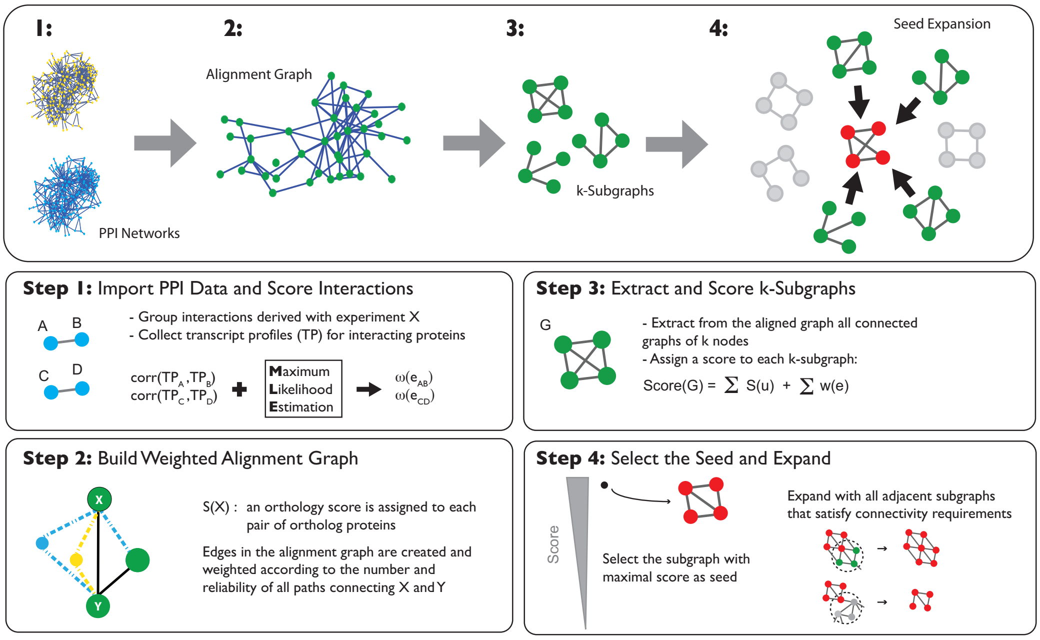

1:
2:
3:
4:
Seed Expansion
Alignment Graph
PPI Networks
k-Subgraphs
Step 1: Import PPI Data and Score Interactions
A
B
C
D
- Group interactions derived with experiment X
- Collect transcript profiles (TP) for interacting proteins
corr(TP_A,TP_B)
corr(TP_C,TP_D)
Maximum Likelihood Estimation
ω(e_AB)
ω(e_CD)
Step 3: Extract and Score k-Subgraphs
G
- Extract from the aligned graph all connected graphs of k nodes
- Assign a score to each k-subgraph:
Score(G) = Σ S(u) + Σ w(e)
Step 2: Build Weighted Alignment Graph
X
Y
S(X) : an orthology score is assigned to each pair of ortholog proteins
Edges in the alignment graph are created and weighted according to the number and reliability of all paths connecting X and Y
Step 4: Select the Seed and Expand
Score
Select the subgraph with maximal score as seed
Expand with all adjacent subgraphs that satisfy connectivity requirements

Figure 2.1: AlignNemo: Method overview

proteins, and scores from Inparanoid are used to weight each node. Each edge of the alignment graph is weighted according to a scoring strategy that incorporates information on the network context in terms of number, reliability and local significance of the paths connecting its endpoints in the input networks. This strategy is implemented by means of an auxiliary structure, the *union graph*, that is crucial to the overall performance of the method.

- Second, all connected k-subgraphs (here $k = 4$) are extracted from the alignment graph and scored based on weights of nodes and edges. Top ranking highly connected k-subgraphs will be used as seeds for the alignment solution.

- Third, each seed is expanded in an iterative fashion by exploring the local neighborhood of the current solution beyond its immediate neighbors. Specifically, the expansion process adds at each step all the subgraphs that are more tightly connected by reliable interactions to the current solution than to the rest of the network.

This approach is in line with recent findings on modularity and organization of complexes in networks, according to which complexes in PPI networks tend to consist of a *core* part and *attachments*. The core is defined as a small group of proteins that are functionally similar and have highly correlated transcriptional profiles. The core is surrounded by less strongly connected proteins, defined attachments, present in multiple complexes which allow diversification of potential functions [4]. This diversification is well reflected by the structure of the solutions provided by AlignNemo, characterized by several overlapping modules, rather the separated subnetworks with no intersection.

Alignment Graph

The alignment graph $G_A = (V_A, E_A)$ is a weighted graph, in which nodes represent pairs of homologous proteins and edges conserved interactions. As already mentioned, the existing definitions of the alignment graph differ in the way edges are set between two nodes. Most representations exploit a limited amount of topological information from the input since they discard almost all the nodes not involved in homologous associations and their interactions.

To overcome this problem it has been designed a new scoring strategy for the edges of the alignment graph that incorporates topological information present in the original networks in terms of number, reliability and significance of paths of

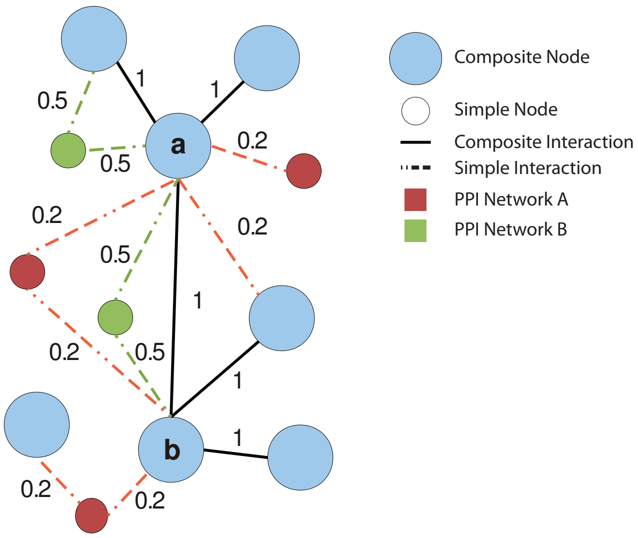


Figure 2.2: Demonstrative union graph used for the path scoring example. Light blue nodes represent composite nodes of V^c; red and green nodes represent nodes $\in V^s$ from V_1 and V_2, respectively. Red and green dotted edges represent interactions in E_1 and E_2, respectively, without a corresponding interaction in the other network. Black solid edges represent conserved interactions between G_1 and G_2.

length less than or equal to 2 between two nodes. The construction and scoring of the alignment graph consists of three steps: (i) merge all input network data into the union graph, (ii) process the union graph to create a raw alignment graph, and finally (iii) perform some pruning operations on the raw alignment graph to remove noise and speed up the overall computation.

Union graph

The purpose of the union graph is to merge all input data into a single graph without losing information. Given two weighted networks $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$, and a set of homologous associations $H = \{(u, v), u \in V_1, v \in V_2\}$ between the nodes of G_1 and G_2, the union graph $U(G_1, G_2, H)$ contains two type of nodes: (i) *composite nodes* representing pairs of homologous proteins, one from each network, as listed by H, and (ii) *simple nodes* representing the proteins of the two input networks that do not have an homolog in the other network. Any edge contained in one of the input networks is represented in the union graph by adding an edge between all pairs of corresponding nodes, either simple or composite. Formally: The **union graph** $U(G_1, G_2, H) = (V_U, E_U)$ is a graph having the following structure:

$V_U = V^s \cup V^c$

where:

$V^s = V_1^s \cup V_2^s$ and

$V_1^s = \{i \quad | \, i \in V_1 \quad \text{and} \quad (i,j) \notin H \quad \forall j \in V_2\}$ is the set of simple nodes from G_1,

$V_2^s = \{i \quad | \, i \in V_2 \quad \text{and} \quad (h,i) \notin H \quad \forall h \in V_1\}$ is the set of simple nodes from G_2,

$V^c = \{i = (u,v) \in H\}$ is the set of composite nodes.

$$E_U = \left\{ (i,j) \;\middle|\; \begin{array}{l} i \in V_1^s, \quad j \in V_1^s \quad \text{and} \quad (i,j) \in E_1, \\ i \in V_2^s, \quad j \in V_2^s \quad \text{and} \quad (i,j) \in E_2, \\ i = (u,v) \in V^c, \quad j \in V_1^s \quad \text{and} \quad (u,j) \in E_1, \\ i = (u,v) \in V^c, \quad j \in V_2^s \quad \text{and} \quad (v,j) \in E_2, \\ i = (u,v) \in V^c, \quad j = (x,w) \in V^c \quad \text{and} \quad ((u,x) \in E_1 \quad \text{and/or} \quad (v,w) \in E_2) \end{array} \right\}$$

Assume that each edge e of E_1 and E_2 is labeled with a reliability score $w(e)$, and each association $k \in H$ is labeled with a reliability score $w(k)$. Then edge

(i, j) in $U(G_1, G_2, H)$ is assigned a score $w(i, j)$ given by the score of the corresponding edge in the input network; the only exception is when both i and j are in V^c, i.e. they are composite nodes, and there is a corresponding edge in both input networks, in such a case $w(i, j)$ is the sum of the scores of the two original edges. Figure 2.2 gives an example of the structure of a union graph.

Raw alignment graph

The alignment graph $G_A = (V_A, E_A)$ can be seen as a reduced version of the union graph in which only composite nodes are retained and an edge connects two nodes if there is at least one path of length less than or equal to 2 between the two nodes in the union graph. The intermediate node of a path of length 2 may be either simple or composite. The most important part of the definition of the alignment graph consists of an edge scoring strategy that summarizes the local topology of the union graph by taking into account all paths connecting two nodes in the union graph that satisfy certain criteria. This strategy is based on the assumption that homologous proteins connected by a large number of paths are likely to be functionally related. Each path between the two nodes thus provides additional evidence of their relatedness.

The choice of considering pairs of nodes at a distance not grater than 2 in the union graph appears reasonable. On the one hand, considering only directly connected node pairs is not suited for aligning evolutionary distant species and it is not robust against missing interactions in original PPI networks. On the other hand, adding edges between node pairs at a distance greater than 2 significantly increases the number of edges of the alignment graph, without providing any benefit in terms of quality of results, as our experiments showed. It has to be noted that some paths of length 2 in the union graph are spurious, i.e. they do not correspond to a path in an input network. Such paths are ignored in our analysis.

Paths of length 2, henceforth referred to as *indirect paths*, take a major role due to the missing interactions in the original PPI networks. However, not all the indirect paths have the same significance. In particular, indirect paths may pass through highly or loosely interacting proteins. If a node is highly interacting within the union graph then the probability that two nodes communicate through it is high. Moreover, the edges composing different paths could have different confidence scores and might represent conserved or non-conserved interactions.

A score based on Jaccard index [32] was used to take all these observations into account. Each edge $e_A = (a, b)$ in the alignment graph is scored based on the number of paths of length 2 that link a and b. The final score of the edge between two nodes a and b of G_A is given by the sum of two terms: a direct contribution S_1 and an indirect contribution S_2. The direct contribution is evaluated as the ratio of the score of the direct path (a, b) connecting a and b in the union graph (if it exists) divided by the sum of the scores of all the direct paths connecting a or b to any other composite node in the union graph. Analogously, the indirect contribution is evaluated as the ratio of the score of the paths of length 2 connecting a and b in the union graph divided by the sum of the scores of all the paths of length 2 connecting a or b to any other composite node in the union graph. This collection of paths connecting two composite nodes, referred to as *extended local interactome*, is formally defined as follows:

Extended Local Interactome (ELI) score

Let $w(a, b)$ represent the score of the edge connecting nodes a and b in the union graph ($w(a,b)=0$ if $(a,b)\notin E_A$), and $w(p_{ab})=w(a,i_1)+\ldots+w(i_{k-1},b)$ be the score of a path of length k connecting a and b. Then, if $E_k(a)$ is the set of paths connecting a to its neighbors at distance k, and $w(E_k(a))$ is the sum of the scores associated to these paths, then:

$$S_1(a,b)=\frac{w(E_1(a)\bigcap E_1(b))}{w(E_1(a)\bigcup E_1(b))}$$

$$S_2(a,b)=\frac{w(E_2(a)\bigcap E_2(b))}{w(E_2(a)\bigcup E_2(b))}$$

$$ELI(a,b)=S_1(a,b)+S_2(a,b)$$

or, equivalently:

$$ELI(a,b)=S_1(a,b)+S_2(a,b)$$

$$S_1(a,b)=\frac{w(a,b)}{\sum_{x\in N^c(a)}w(a,x)+\sum_{y\in N^c(b)}w(y,b)}$$

$$S_2(a,b)=\frac{\sum_{x\in N(a)\bigcap N(b)}w(a,x,b)}{\sum_{x\in N(a),y\in N^c(x)}w(a,x,y)+\sum_{x\in N(b),y\in N^c(x)}w(a,x,y)}$$

The power of this scoring strategy relies in its ability to account once again for the local neighborhood of aligned nodes: while methods such as NetworkBLAST or Mawish allow for gaps or mismatchs to connect conserved proteins at distance 2 in the aligned graph, it accounts for the whole set of paths connecting pairs of conserved proteins and for their reliability.

An example is presented in Figure 2.2, where for simplicity each solid black edge has score 1, and each edge present only in the first or second network has a score of 0.5 and 0.2, respectively. Consider nodes labeled a and b. The direct path connecting a and b has score $w(a, b) = 1$. Node a has 3 composite nodes connected through conserved edges, and 1 composite node connected through non-conserved edges. Node b has 3 composite nodes connected through conserved edges, and 0 composite nodes connected through unpaired edges. Therefore, the contribution of direct paths is:

$$S_1(a, b) = \frac{1}{3 * 1.0 + 1 * 0.2 + 3 * 1.0 + 0} = \frac{1}{6.2}$$

There are 3 indirect paths between a and b scoring respectively $(0.2 + 0.2) = 0.4, (0.5 + 0.5) = 1, (0.2 + 1) = 1.2$. Node a has 6 indirect paths connecting it to other composite nodes, for a total score of 7.6. Node b has 7 indirect paths connecting it to other composite nodes, for a total score of 8.2. Therefore, the contribution of indirect paths between i and j is

$$S_2(a, b) = \frac{2.6}{7.6 + 8.2} = \frac{2.6}{15.8}$$

The final score is $ELI(a, b) = S_1(a, b) + S_2(a, b) = \frac{1}{6.2} + \frac{2.6}{15.8} = 0.3258$

Pruning the Union Graph

The alignment graphs resulting from the above construction tend to be very dense with edge scores spreading over a wide range of values. Removing less reliable edges is thus necessary for simplifying the alignment graph and reducing the computational cost in the next steps of the alignment procedure. Two interesting facts emerge when looking the distribution of edge scores:

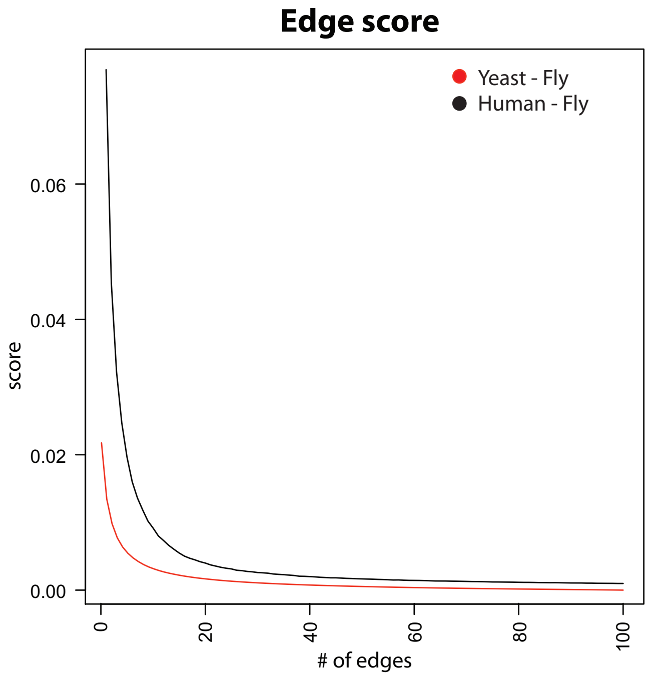


Figure 2.3: Sorted scores of edges incident to two nodes of two alignment graphs. High and low scoring edges can be separated easily. Indeed, scores drop exponentially after the first few edges, leaving ample margins for the pruning threshold.

- Few edges have a score significantly higher than the others.
- Edge scores vary considerably across different regions of the alignment graph and are affected by topological characteristics, such as interaction density. Thus, pruning the edges based on a global threshold may not be appropriate.

Following these two observations, a pruning strategy processes all the edges incident to the same node at once, and retains only locally high scoring edges. A simple yet effective rule has been used:

For each node $x \in G_A$, let $ELI(x, y^*) = max_{y \in N(x)}(ELI(x, y))$. For a given constant t, all the edges $(x, y),\ y \in N(x)$, with score $ELI(x, y) < tELI(x, y^*)$

are deleted.

The pruning strategy is tunable by varying the threshold t, thus allowing to create denser or sparser networks. A $t = 0.5$ has been used in this work. Pruning thresholds t ranging from 0.3 to 0.7 were tested with similar results. This was expected, since the distance between high scoring and low scoring edges incident to the same node is sharp, as clearly visible in Figure 2.3. On the other hand, not pruning low scoring edges (t = 0) introduce a huge number of spurious edges. Indeed, the application of this procedure leads to a drastic reduction of the number of edges of the alignment graph.

Dealing with multiple orthologs

Homology associations are tipically many-to-many and proteins associated to many putative orthologs will appear as multiple nodes in the alignment graphs. This becomes critical when such proteins are included multiple times within the same solution, decreasing the accuracy on the final mapping.

The following strategy exploits the topology of the networks to correct the weight of the edges connecting nodes involved in multiple homologous associations. Assume that $y_1, y_2, \cdots, y_k$, $y_i = (u, v_i)$, are nodes of G_A corresponding to multiple associations of the same node $u \in V_1$, with k nodes $v_1, \cdots, v_k$ of V_2 . Furthermore, assume that $y_1, y_2, \cdots, y_k$ are all adjacent to node x in the alignment graph. The goal is to identify, among these perhaps conflicting associations, the ones that most likely correspond to true interactions with x. To exploit the topology of the network, the edges $(x, y_1), (x, y_2), ..., (x, y_k)$ are sorted according to their score $S(x, y_i)$, and their ranks $r(x, y_i)$ in the sorted list are considered.

The corrected score for each edge is obtained by dividing it by its rank:

$$ELI'(x, y_i) = \frac{ELI(x, y_i)}{r(x, y_i)}$$

This correction reduces the weight of the edges leaving the highest scoring ones unaffected. This procedure has been applied before the pruning step described above. A significant improvement both in terms of the quality of the solution and computational costs has been observed. In the rest of the manuscript the term ELI refers to this corrected score.

Table 2.2 reports statistics on the alignment graphs produced for the $S.cerevisiae$-$D.melanogaster$ and $H.sapiens$-$D.melanogaster$ network alignments.

	human - fly	fly - yeast
G_1 nodes	12113	8042
G_1 edges	78559	24235
G_2 nodes	8042	5185
G_2 edges	24235	24932
Orthologies	6137	10045
Union graph nodes	18535	19844
Union graph edges	51515	303341
Alignment graph nodes (no multiple-ortholog correction)	1992	8809
Alignment graph edges (no multiple-ortholog correction)	3526	38789
Alignment graph nodes	1941	5554
Alignment graph edges	2973	4740

Table 2.2: Statistics for the input dataset, and the resulting Union graphs and Alignment graphs.

Graphlet extraction and Seed Generation

Once the final alignment graph is built, a mining step is performed to identify interesting subgraphs with interesting properties. AlignNemo features a greedy strategy that tries to expand local solutions starting from potential seeds. A seed consists of a small subgraph of the alignment graph of fixed size k, i.e. a k-subgraph. First, all k-subgraphs are extracted from G_A, allowing arbitrary overlap of nodes and edges, then the non-overlapping top scoring ones are selected as seeds while the rest will be picked iteratively to expand the seeds. In this work $k = 4$ has been used in all the experiments.

Enumerating all k-subgraphs with arbitrary overlap can be time consuming due to the large number of small subgraphs that is possible to extract even from sparse networks. To optimize the extraction process, a simple heuristic to avoid counting multiple times the same instance has been implemented, so that each subgraph is found exactly once. More precisely, first an arbitrary order is imposed on the nodes of the graph $\mathcal{O} : V_A \rightarrow \mathbb{N}$, and then all the subgraphs containing node u are extracted by iteratively looking at nodes at distance less than k from u in the graph, $N_k(u)$, such that $\mathcal{O}(v) > \mathcal{O}(u)$, for each $v \in N_k(u)$.

A score is then assigned to each k-subgraph based on the individual scores of its components, i.e. nodes and edges. Precisely, given a subgraph g of the alignment graph G_A, and denoted by $V_A(g)$ and $E_A(g)$ the set of nodes and edges of the subgraph g, respectively, let's define

$$Score(g) = \sum_{k \in V_A(g)} w(k) + \sum_{(i,j) \in E_A(g)} ELI(i,j)$$

where $w(k)$ scores the confidence in the the two associated proteins being orthologous, and $S(i, j)$ is the score of the edge (i, j) in the alignment graph as defined above.

Module Discovery

Once all k-subgraphs have been extracted and scored, the algorithm ranks them according to their scores and selects the one with highest score as $seed$. Starting from the seed, the algorithm expands the candidate solution iteratively. From now on, the candidate solution will be referred as *module*. The algorithm consists of a number of expansion steps. During each expansion step, all the k-subgraphs adjacent to the module, i.e. sharing at least one node with it, are picked as candidates for expansion. All the k-subgraphs that satisfy specific requirements are added to the module, thus at each step one or more k-subgraphs are added.

The selection of the k-subgraphs to add to the module is the key point of the method. Let's denote by $IE(v)$ the set of edges of graph G_A incident on node v, and by $IE_g(v)$ the set of edges of subgraph g incident on node v. Finally, for a subset S of T, let's denote by $T \backslash S$ the subset of elements of T that are not in S. Given the current module M, a candidate subgraph g, and the remaining part of the alignment graph $N = G_A \backslash \{M, g\}$, the set of edges incident on a node $v \in g$ can be divided into subsets according to which subset the other endpoint belongs to, i.e. g, $M \backslash g$, or N. Formally:

$$IE(v) = IE_g(v) \cup IE_{M \backslash g}(v) \cup IE_N(v).$$

A k-subgraph is $tightly$ connected to the module if

$$IE_{M\setminus g}(v) \neq \emptyset, \quad \forall v \in g.$$

Tightly connected subgraphs are always added to the module. $Loosely$ connected subgraphs are attached if they connect to the module with more reliable links than to the rest of the network.

Using the notation introduced above, for a given k-subgraph g:

$$w(IE_M(g)) = \sum_{v \in g} \sum_{e \in IE_M(v)} ELI(e),$$

$$w(IE_N(g)) = \sum_{v \in g} \sum_{e \in IE_N(v)} ELI(e)$$

the sum of the weights of edges connecting g to the module, and the sum of the weights of edges connecting g to the rest of the network, respectively. Then g is added to the module if:

$$\Delta w = w(IE_N(g)) - w(IE_M(g)) < 0.$$

At the end of the expansion stage all accepted k-subgraphs are added to the module at once. The process is repeated until no more k-subgraphs can be added, thus there is not a fixed upper limit to the size of obtainable complexes. On the other hand the solutions are required to have at least 5 nodes, a limit imposed by the size of the seed (4 nodes) and the requirement of at least one expansion step to be completed. It is important to remark that expanding the module by k-subgraphs rather than by a single node at a time is not only crucial for the good performance of the method, but it is the key to account for multiple dependencies between a protein and its immediate neighbors.

Implementation

AlignNemo is fully implemented in Java and has no dependencies from external libraries. The alignment of S.cerevisiae and D.melanogaster required 3 minutes and 30 seconds, while the alignment of H.sapiens and D.melanogaster required 43 seconds. Both NetworkBLAST and Mawish are written in C; AlignNemo shows run-times that are generally comparable to those of NetworkBLAST, while Mawish showed faster performance requiring 10 seconds for both the alignments.

2.2.2 Assessment

AlignNemo has been tested on three well studied organisms: *D. Melanogaster* (fruit fly) *S. Cerevisiae* (baker's yeast) and *H. Sapiens* (human).

AlignNemo, NetworkBLAST, Mawish and NetAligner have been applied on the same datasets, each algorithm producing a set of solutions, or modules, possibly overlapping. A module M is a graph containing a set of protein pairs from the two input networks. The set of proteins from network G_1 and G_2 in M will be referred to as M_{G_1} and M_{G_2}, respectively. In this assessment the whole sets of solutions are first compared to a dataset of known complexes. Then, the associations of proteins from different species are biologically sound using the concept of semantic similarity applied to Gene Ontology vocabularies. Finally, few specific cases to highlight weaknesses and strengths of each method are presented.

Input Data

Protein-protein interactions for *D. melanogaster* and *S. cerevisiae* were derived from the Database of Interacting Proteins (DIP - updated 10/27/2011) [33]. They include 7548 proteins and 22969 interactions in fly, and 5053 proteins and 22254 interactions in yeast. Inparanoid [31] was used to select 10045 pairs of putative orthologous proteins from the two networks, involving 1878 proteins from yeast and 1511 proteins from fruit fly. *H. sapiens* PPI network was derived from the HIPPIE database [34]; it includes 12113 proteins and 78559 weighted interactions coming from 17 different sources. A set of putative orthologous protein pairs from human and fly were obtained from the Gerstein Lab [35].

These data sets integrate multiple sources and include interactions derived from different methodologies including high-throughput and small scale experiments. To account for such diversity a reliability score has been assigned to each edge.

For both networks derived from DIP (fruit fly and yeast) a maximum likelihood estimation procedure defined in [36] has been employed to assess the reliability of protein interactions determined through the same experimental procedure. This method is based on the observation that correlations of gene expression profiles through different time points are good features to evaluate PPI reliability: interacting proteins typically show high correlation values. In applying this method random pairs of proteins not known to be interacting are considered as *true non-interacting* proteins, and interactions determined by small scale experiments as *true interacting* proteins, estimating from these two sets the respective distributions of correlation coefficients. For yeast proteins the set of expression profiles

Algorithm	No. of S.	M.S.	$F_1 > 0.3$	S.C.R.
	Fly-Yeast			
Mawish	175	32	29	16
NetworkBLAST	329	46	30	18
NetAligner	140	32	41	**49**
AlignNemo	242	54	**52**	27
	Fly-human			
Mawish	87	37	60	33
NetworkBLAST	45	23	13	24
NetAligner	133	40	81	84
AlignNemo	115	53	**87**	**89**

No. of S.: Number of Solutions
M.S.: Matching Solutions
S.C.R.: Small Complex Recovered

Table 2.3: Comparison of AlignNemo, Mawish, NetworkBLAST, and NetAligner, on yeast-fly and fly-human alignments. The number of solutions found by each algorithm (No. of S.) is listed in column 2. The number of solutions that match at least one known complex is reported in columns 3 (M.S. - Matching Solutions). The number of high-quality matches for complexes of size ≥ 4 is summarized in columns 4 ($F_1 > 0.3$), while the number of small complexes (2-3 proteins) recovered is in columns 5 (S.C.R. - Small Complex Recovered).

reported in the SGD database [37] has been used, and assigned a confidence score to each experimental method described in DIP and to combination of them. The scores of the fly interactions were computed based on the assumption that a given experimental method works equally well in different organisms and therefore the confidence scores based on yeast data were transfered to fly interactions. Reliability scores for the human protein interactions network were available through the web server HIPPIE.

Detection of known complexes

The quality of the results has been assessed by evaluating the agreement of the modules found by each method with known complexes. Given a module and a known complex two widely used measures from information retrieval, precision (π) and recall (ρ), are computed. $Precision$ is defined as the percentage of proteins in the module that are also present in the complex; $recall$ is defined as the percentage of proteins in the complex that are also present in the module. The F_1-score function, defined as the harmonic mean of precision and recall, is used to integrate these measures into a single score. Formally, these measures are defined as follow:

$$\pi = \frac{TP}{TP + FP}, \quad \rho = \frac{TP}{TP + FN}, \quad F_1\text{-score} = \frac{2\pi\rho}{\pi + \rho}$$

where TP is the number of true positives, i.e. the number of proteins found in a solution that are also in the complex. Analogously, FP and FN are the number of false positives and false negatives. The F_1-score ranges in the interval [0, 1], with 1 corresponding to perfect agreement. In this analysis each known complex of species G_i is matched to all the modules M_{G_i} from a given algorithm, and the module with highest F_1-score is selected as best match.

The set of complexes in CYC2008 [38], a comprehensive catalogue of 408 yeast protein complexes derived from small scale experiments and literature mining, has been considered to evaluate the results for the alignment of *S. cerevisiae* and *D. melanogaster*. For the alignment of *D. melanogaster* and *H. sapiens*, instead, the complexes in CORUM [39], a dataset of 1682 human protein complexes, have been used. About the 28% of CYC2008 and CORUM complexes are composed by only 2 or 3 proteins (132 for CYC2008 and 474 for CORUM). This may be problematic as statistical measures tend to be hardly interpretable for such small complexes. For this reason, the following analysis is restricted to complexes with at least 4 proteins. However, to avoid biases the ability of each method to recover small complexes (2-3 proteins) has been verified as well. A small complex is recovered (hit by a solution) if at least 2 of its proteins overlap with an alignment solution, excluding the solutions exceeding 20 nodes. Table 2.2.2 summarizes the performance of the four algorithms. The number of modules found by each algorithm and, among those, the number of high quality modules, i.e. those that match a known complex with an F_1-score greater than 0.3, is reported for each algorithm. The overall distribution of F_1-scores obtained by AlignNemo, Mawish, and NetworkBLAST is estimated by the respective kernel density distribution and shown in Figure 2.4 (A-B). In Figure 2.4 (A-B) The performance of each method are reported in terms of precision and recall separately. Both NetworkBLAST and AlignNemo perform better on the yeast-fly alignment, with the latter having overall higher values of both precision and recall. The small solutions found by Mawish have in general high precision while inevitably failing in recovering most proteins in a complex. AlignNemo clearly outperforms the other approaches in recovering known complexes, showing the highest percentage of high quality modules. It should be noticed that while Mawish performs similarly well for the fly-human alignment, the majority of modules produced by this method have small size, specifically 90% of them consists of 2 nodes only.

Protein mapping between species

The analysis described in the previous section shows that AlignNemo is able to recapitulate known protein complexes and that detected conserved sub-networks generally reflect known biology within each single species. On the other side, the quality of the mapping between proteins from different species needs further evaluation. To assess the biological relevance of the discovered mappings the semantic (functional) similarity has been employed. This analysis requires the use of prior biological knowledge that is encoded into ontologies. The Gene Ontology (GO) and its annotations are used as input data to determine the functional similarity between two proteins from different species, by using the concept of *semantic similarity* [40]. For each solution a semantic similarity score is computed using the set of annotations from the Biological Process (BP) and Molecular Function (MF) ontologies in GO. Only the results for BP are reported here as this ontology more closely reflects the idea of protein complexes as sub-cellular units involved in specific processes. Additional results can be found in Table S2 of the Supplementary Materials of [30].

Given two proteins p_1 and p_2, and their set of GO annotations $GO(p_1)$ and $GO(p_2)$, the Resnik similarity measure [41, 42] is used to score each pair (go_i, go_j) with $go_i \in GO(p_1)$ and $go_j \in GO(p_2)$. The semantic similarity of p_1 and p_2 is defined as the average of the scores of the best match for each GO term in $GO(p_1)$ and $GO(p_2)$ according to the Resnik measure [43]. Semantic similarities were computed using the tool FastSemSim [44].

In total 356 solutions has been tested for AlignNemo, 85% of which have between 5 and 15 proteins and the largest 93 proteins; 362 solutions for Network-

BLAST, each including between 5 and 15 proteins, the latter being a limit imposed by the method; and 260 solutions for Mawish, each including between 2 and 6 proteins. Given the striking difference in terms of size of the detected sub-networks, The results obtained by the three methods separately for small complexes (< 7 proteins) and large ones (≥ 7 proteins) are summarized in Figure 2.4 (C-D).

Results for both protein network alignments show similar performance for the three algorithms in terms of semantic similarity, with better performance for the *H.sapiens* - *D.melanogaster* protein alignment.

Topology of conserved modules

As discussed in the Introduction, protein complexes are typically composed of densely interacting proteins. However, recent findings on modularity and organization of complexes in PPI networks show that they tend to consist of a densely connected *core* and a less strongly connected set proteins defined *attachment*. The latter is typically present in multiple complexes and allows diversification of potential functions [4].

Following this model, AlignNemo looks for *relatively* densely connected proteins, i.e. proteins that have more interactions among themselves than with the rest of the network, rather then imposing rigid and fixed constraints on the topology of a candidate solution.

It is important to test whether this strategy puts at risk the ability to detect densely connected cores, including among the solutions sparse sub-networks unlikely to be actual protein complexes. To this purpose, 1000 random networks for each PPI network have been generated, preserving their node degree distribution;

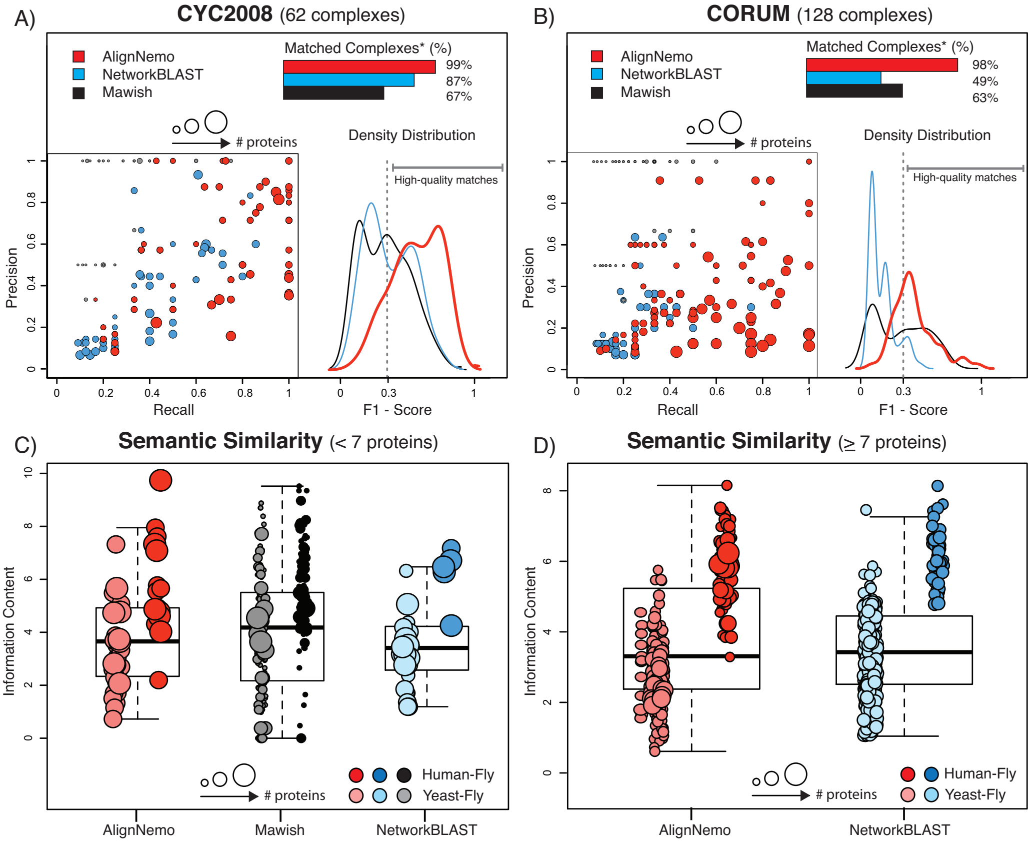


Figure 2.4: Comparison of AlignNemo, NetworkBLAST, and Mawish. The three algorithms are evaluated in terms of recovering known protein complexes in both $S.cerevisiae$ (CYC2008) and $H.sapiens$ (CORUM). Solutions matching known complexes are scored by means of precison, recall, and F_1 score. Obtained score distributions for each method are plotted in panel (A) for yeast-fly alignment, and panel (B) for human-fly alignment. Panels (C) and (D) show the average semantic similarity between proteins from different species mapped by each solution. Each solution is represented by a circle with the radius proportional to the size of the solution. The size of the solutions from each method varies significantly, thus small (<7 nodes) and big (≥ 7 nodes) solutions are shown separately. Percentages refer to the set of complexes matched by at least one method.

then for each module its connectivity is evaluated in the original PPI networks and in the random set. As connectivity score the number of interactions between the proteins within the solution has been used. For each species and each solution it is now possible to estimate a background distribution of its connectivity. The deviation of the observed connectivity in the real network, c_i, from such background distribution can be measured using a Z-score:

$$Z = \frac{c_i - \bar{c}_{rand}}{\sigma_{rand}}$$

where $\bar{c}_{rand}$ is the average connectivity for this set of proteins in the random set and σ_{rand} its standard deviation.

The two sets of proteins defined by each solution, one for each species, are first considered separately, and the maximum Z-score between the two obtained is associated to each solution. This strategy takes into account the cases where to a relatively poorly connected set of proteins in one species corresponds a group of orthologs that are densely interacting in the other species. A p-value is empirically derived for each module from this background distribution, and it is given by the number of random networks that led to a greater or equal Z-score for the tested module over all possible networks. Interestingly, the 95% of the solutions, both for the human-fly and the the yeast-fly alignments, show statistically significant higher connectivity than those observed in the randomized networks.

In conclusion, AlignNemo outperforms both Mawish and NetworkBLAST in correctly detecting protein complexes within single species given their interactomes and orthology relationships. Furthermore, protein mappings between different species are biologically sound as proven by the average semantic similarity between proteins in the same module. Finally, despite AlignNemo does not impose rigid constraints on the module topology, exploring less strongly connected components of a protein complex, the extracted subnetworks are more densely connected than expected by chance.

Comparison with NetAligner

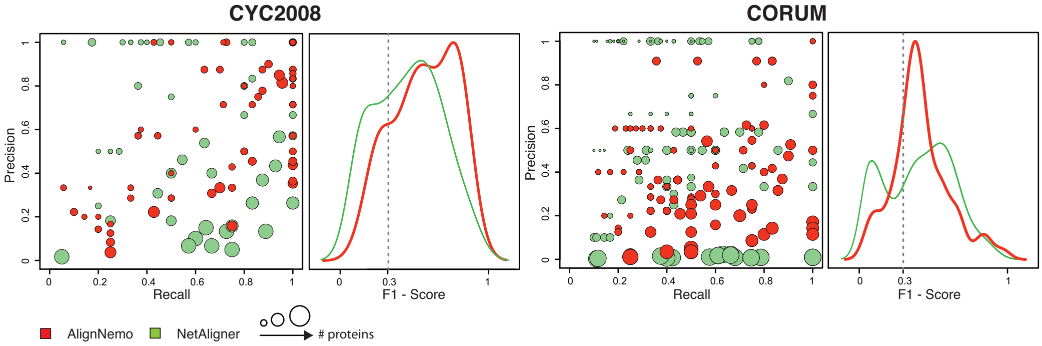


Figure 2.5: **Comparison of AlignNemo and NetAligner**. The two algorithms are evaluated in terms of recovering known protein complexes in both $S.cerevisiae$ (CYC2008) and $H.sapiens$ (CORUM). Solutions matching known complexes are scored by means of precison, recall, and F_1 score.

NetAligner relies on a novel algorithmic approach to compute probalitities associated to conserved interactions, based on protein sequence similarity between proteins from different species. Given two pairs of putative orthologs, NetAligner evaluates the likelihood that they share a conserved interaction by considering the difference of evolutionary distances between the two orthologous pairs. In this work NetAligner has been tested under different configurations and input data, including the original proteomes and homologies provided with the tool. According to our analysis NetAligner achieves the best performance when using the *predict likely conserved interactions* setting, together with the parameters suggested in its reference paper [27]. NetAligner extracts a bigger and more reliable set of alignments on its own dataset. For this reason the results of NetAligner run on its own dataset has been used in the following comparison.

When the solutions are matched to the reference complexes (CYC2008 and CORUM), the two methods perform similarly (see Figure 2.5 and Table 2.2.2). AlignNemo shows again better overall performance for the $S.cerevisiae$-$D.melanogaster$ alignment. In the $H.sapiens$-$D.melanogaster$ alignment, NetAligner finds a set of higher scoring small solutions, but at the same time several matches are produced by a very large solution including 463 nodes, leading to high recall values despite a precision close to zero (Figure 2.5).

Conserved Complexes

This section focuses specifically on few complexes of CYC2008 and CORUM to better dissect the performance of different methods. Cases discussed here include a small complex, *Arp2/3*, and two relatively large complexes, *TFIID (general transcription factor)* and *20S Proteasome*, with different level of connectivity. Table 2.4.4 reports the proteins of these complexes that have been correctly asssociated and recovered by at least one between AlignNemo, NetworkBLAST, and Mawish in the $H.sapiens$ and $D.melanogaster$ network alignment. For both the Transcription Factor TFIID and Arp2/3 complexes AlignNemo performs better according to both F_1-score and semantic similarity. In detecting the 20S Proteasome, AlignNemo and NetworkBLAST have comparable recall for yeast-fly alignment, but AlignNemo has higher precision. Also, AlignNemo shows a superior performance in the human-fly alignment. Significantly enriched GO catagories for our

Complex Name: Actin related protein 2/3 (ARP 2/3)			**Complex size: 7 proteins**		
Method:			**Mawish**	**AlignNemo**	**N.BLAST**
Solution size:			**-**	**6**	**-**
Protein Function	ID Human	ID Fly	Correctly selected		
ARP 3B	ARP3B	P32392		●	
ARP 2/3 subunit 2	ARPC2	Q9VIM5		●	
ARP 2/3 subunit 3	ARPC3	Q9VX82		●	
ARP 2/3 subunit 5	ARPC5	Q9VQD8		●	

Complex Name: Transcription Factor IID (TFIID)			**Complex size: 13 proteins**		
Method:			**Mawish**	**AlignNemo**	**N.BLAST**
Solution size:			**2**	**19**	**10**
Protein Function	ID Human	ID Fly	Correctly selected		
TFIID subunit 1	TAF1	P51123		●	●
TFIID subunit 1 like	TAF1L	P51123		●	●
TFIID subunit 10b	TAF10	Q9XZT7	●	●	
TFIID subunit 11	TAF11	P49906		●	
TFIID subunit 6	TAF6	P49847		●	●
TFIID subunit 7	TAF7	Q9VHY5		●	
TFIID subunit 8	TAF8	Q9VWY6	●	●	
TFIID subunit 9	TAF9B	Q27272		●	
TBP	TBP	P20227		●	●

Complex Name: 20S Proteasome			**Complex size: 14 proteins**		
Method:			**Mawish**	**AlignNemo**	**N.BLAST**
Solution size:			**2**	**11**	**11**
Protein Function	ID Human	ID Fly	Correctly selected		
Proteasome sub. alpha type-1	PSA1	P12881		●	●
Proteasome sub. alpha type-2	PSA2	P40301		●	●
Proteasome sub. alpha type-3	PSA3	Q9V5C6		●	
Proteasome sub. alpha type-4	PSA4	P18053		●	●
Proteasome sub. alpha type-5	PSA5	Q95083		●	
Proteasome sub. alpha type-7	PSA7	P22769	●	●	●
Proteasome sub. beta type-1	PSB1	P40304		●	
Proteasome sub. beta type-2	PSB2	Q9VQE5		●	
Proteasome sub. beta type-3	PSB3	Q9XYN7	●	●	
Proteasome sub. beta type-7	PSB7	Q9VUJ1		●	

Table 2.4: **Comparison of the best matching solutions for Arp 2/3, TFIID, and 20S proteasome complexes.** Homologous proteins correctly included in the best matching solution of at least one algorithm. For Arp 2/3 complex, 4 out of 6 proteins really participate to Arp2/3 human complex, while the other 2 (omitted) are homologous proteins incorrectly included in the solution. NetworkBLAST and Mawish did not provide any solution overlapping with this complex. For TFIID and 20S proteasome complexes, the quality of AlignNemo solution is highlighted by the number of protein pairs belonging to the complex but not selected by Mawish and NetworkBLAST.

Complex Name: Actin related protein 2/3 (ARP 2/3)		**Complex size: 7 proteins**	
Method	GO Term	P-value Human	P-value Fly
AlignNemo	regulation of actin filament polymerization	2.63e-008	1.93e-009

Complex Name: Transcription Factor IID (TFIID)		**Complex size: 13 proteins**	
Method	GO Term	P-value Human	P-value Fly
NetworkBlast	transcription initiation, DNA-dependent	1.14e-05	9.36e-06
AlignNemo	transcription initiation, DNA-dependent	4.57e-25	3.34e-26

Complex Name: 20S Proteasome		**Complex size: 14 proteins**	
Method	GO Term	P-value Human	P-value Fly
NetworkBlast	regulation of ubiquitin-protein ligase activity involved in mitotic cell cycle	4.22e-018	-
	proteolysis involved in cellular protein catabolic process	-	2.92E-011
AlignNemo	negative regulation of ubiquitin-protein ligase activity involved in mitotic cell cycle	9.40e-026	-
	proteolysis involved in cellular protein catabolic process	-	9.12e-022

Table 2.5: GO Enrichment on the best matching solutions for Arp 2/3, TFIID, and 20S proteasome complexes.

solutions have been computed with GOTermFinder [45] and are reported in Table 2.2.2. In both alignments the cross-species semantic similarity is higher for AlignNemo indicating an improvement in biological quality, the details of which are discussed below.

Transcription Factor TFIID Complex

RNA polymerases (I, II, and III) catalyze transcription of nuclear genes and rely on general transcription factors to recognize target promoters; in particular RNA polymerase II relies on the TFIID complex to initiate transcription. The general transcription factor TFIID is mainly composed of the TATA box binding protein (TBP) and a set of TBP-associated factors (TAF_{II}s) or subunits that are well conserved across species [46].

AlignNemo outperformed existing methods in uncovering this complex: it found 9 proteins of TFIID in a solution of 19 nodes; it correctly mapped human proteins into fly proteins corresponding to the same subunit in the two organisms (see Table 2.4.4). Mawish features a solution with only 2 nodes, also included in our alignment, while NetworkBLAST returned a solution of 10 nodes that match 4 proteins pairs belonging to TFIID complex.

Because of the high-connectivity of this complex, AlignNemo and NetworkBLAST solutions extend beyond the boundaries of TFIID complex as defined in CORUM. The quality of these solutions is verified through GO Term enrichment. Up to 16 out of 17 fly proteins and 18 out of 19 human proteins in AlignNemo's solution are enriched for the same GO terms including *Transcription from RNA polymerase II promoter* ($p_{fly} = 1.21e - 23$, $p_{human} = 8.16e - 18$). By contrast, the solution of NetworkBLAST reported only 4 out of 10 proteins in both networks with a common and specific biological role (see Table S3).

Arp2/3

Arp2/3 complex consists of 7 units and plays an important role in the regulation of the actin cytoskeleton. It is a major component of the actin cytoskeleton and is found in most actin cytoskeleton-containing eukaryotic cells [47].

Interestingly, the level of connectivity between these proteins in the original PPI network varies significantly, from 17 interactions found in human to none identified in $D.melanogaster$. Incomplete information makes this complex particularly challenging to recover. Indeed, only AlignNemo was able to identify this conserved complex in $H.sapiens$ and $D.melanogaster$, while both NetworkBLAST and Mawish did not have any solution in overlap with it. Table 2.4.4 lists the correctly detected homologous proteins that were found in the solution of AlignNemo. All 4 are annotated with the *regulation of actin filament polymerization function* GO term ($p_{fly} = 3.07e - 08$ and $p_{human} = 1.24e - 09$). This case nicely points at the importance of considering conserved paths, rather than only direct interactions, to complement missing information in one network.

20S Proteasome Complex

The 20S Proteasome is a large protein complex present in several organisms, in particular in all three organisms considered here. According to CYC2008 and CORUM, the 20S proteasome consists of 14 proteins in yeast and 16 proteins in both human and fly. The topology of the complex is relatively dense and the interactions are reliable.

For the case of $S.cerevisiae$-$D.melanogaster$ network alignment all three methods have comparable values of recall; as for the precision, NetworkBLAST obtains a much lower value since it finds several proteins outside the complex. On the other hand, AlignNemo outperforms the other methods in identifying the 20S Proteasome complex in the $H.sapiens$-$D.melanogaster$ network alignment (see Table 2.4.4). Indeed, it correctly selected 11 proteins of the 20S Proteasome in human and 12 in fly, while NetworkBLAST found only 4 in human and 5 in fly and Mawish only 2 in both networks.

2.3 AlignMCL

The assessment presented in the previous section confirms the quality of AlignNemo's results. As already stated, the two factors that influenced the quality of the results are the model of alignment graph used and the mining strategy. In particular, the new definition of alignment graph that summarizes all the topological information of the input networks is of critical importance. On the other side, employing a strategy that considers multiple nodes at each time, instead of greedily trying to add single nodes to an expanding solution, seems to be quite important as well. One of the major drawbacks of using the graphlets is the combinatorial explosion of the number of candidate motifs to consider. Not only the time required to process all the graphlets increases exponentially with the complexity of the alignment graph, but also the memory required to store all the graphlets is significant. In some cases it might be possible to reduce the size of the alignment graph, for example by emplyoing stricter pruning strategies and thresholds. However, in the last two years the size of available PINs has increased considerably, extending beyond the 100'000 interactions in some cases (i.e. for the human PIN from the i2d database). Moreover, each network has peculiar features, and an accurate tuning of the alignment graph would be required for AlignNemo to obtain good results. To overcome this problem, and avoid the use of naive greedy expansion procedures, a different mining stretegy has been used in a novel pipeline called AlignMCL.

2.3.1 Design and Implementation

AlignMCL is a local network alignment algorithm based on the same strategy of AlignNemo. All the input data are first merged in a single graph that is examined afterwards. The same alignment graph model introduced with AlignNemo is used in AlignMCL. The Markov CLuster (MCL) algorithm [48] is used to extract the conserved subnetworks, instead of the motif-based engine of AlignNemo.

Mining strategy: Markov Clustering Algorithm (MCL)

The Markov Cluster algorithm (MCL) [48] is a well known algorithm used to find clusters on graphs, robust to noise and graph alterations. Brohee and Van Helden demostrated in an extensive comparison [49] that MCL outperforms other clustering algorithms, such as MCODE [10], RNSC [50] and Super Paramagnetic Clustering [51], in different conditions and using suboptimal parameters. It is quite surprising that no previous work ever considered the use of such algorithm in network alignment problems. To the best of our knowledge, our work is the first to adopt such an algorithm in the context of network alignment. Intuitively, a cluster on a network is a collection of nodes that are more connected to each other than to the other nodes of the network. It follows that a random walk starting in any of these nodes is more likely to stay within the cluster rather than to travel between clusters. MCL simulates a stochastic flow on the network that resembles a set of random walks on the graph.

Briefly, MCL consists on two main operations: expand and inflate. The expand step spreads the flow out of a vertex to potentially new vertices, particularly enhancing the flow toward those vertices that are reachable by multiple (and short) paths. The inflation step introduces a modification into the process, enhancing the flows within the clusters and weakening the inter-cluster flows. In this way the initial distribution of flows, relatively uniform, becomes more and more non-uniform, inducing the emergence of a cluster structure, i.e. local regions with high level of flow. The inflation process is tuned by the *inflation* parameter.

Implementation

The Java-based implementation used in AlignNemo to build the alignment graph is too slow and requires too much memory to be scalable on the PINs currently available. Three different algorithmic strategies have been tested, following the idea of designing space-optimized code. The final implementations are able to align the input networks requiring less than half an hour in the worst case, on a normal desktop system equipped with 8 GB of ram.

AlignMCL is divided in two components: a routine written in Python to build the alignment graphs, and the mining engine implemented in C. The separation is quite functional, since it allows to use the two components separately. For instance, the mining strategy has been successfully applied on the global alignment graph produced by mi-GRAAL, producting few high-quality alignments.

2.3.2 Assessment

AlignMCL is compared against the other state-of-the-art algorithms. Since one of the goals of a local alignment algorithm is to uncover conserved complexes, its quality can be assessed by evaluating how well the solutions produced resemble known complexes in the aligned species. The quality of the overlap between the solutions of an algorithm and a set of known complexes can been expressed through precision (π), recall (ρ), and F-index.

Evaluating the algorithm's performance over an extensive dataset is fundamental to avoid overfitting and biases, and prove the general adaptability of the algorithm. For this reason an assessment has been devised, based on a set of 14 alignments between five of the most studied species: Drosophila Melanogaster

Species (Source)	Proteins	Interactions
D. Melanogaster (DroID)	9181	88122
D. Melanogaster (I2D)	9854	37979
H. Sapiens (I2D)	14567	138258
M. Musculus (I2D)	4261	9547
C. Elegans (I2D)	4755	9995
S. Cerevisiae (I2D)	6182	147408

Table 2.6: Statistics of PINs used in the assessment.

(DM), Saccaromices Cerevisiae (SC), Homo Sapiens (HS), Caenorhabditis Elegans (CE), and Mus Musculus (MM). Alignments are symmetric: HS-DM and DM-HS, for instance, refer to the same set of solutions. Some tools, such as NetAligner, are not strictly symmetric. In these cases the alignment with the best results has been used in the comparison.

Input Datasets

The interaction networks of mouse, yeast, human, worm and fly have been downloaded from the I2D database [52] (release of 2011). An additional PIN has been retrieved from DroID [53] for the fly organism. Not all the algorithms have been able to deal with the biggest networks, including NetworkBlast and AlignNemo. The size of the various PINs is reported in Table 2.3.2.

The Integrative Ortholog Prediction Tool (DIOPT) [54] has been used to build a comprehensive set of input homology associations. Table 2.3.2 summarizes the number of homologies provided by DIOPT. Since some algorithms require BLAST [55] data in addition or in substitution of the data provided by DIOPT, the complete primary sequence dataset has been downloaded from the NCBI website [56], and a BLAST sequence alignment has been performed between the proteins of the different species. The standard parameters suggested in the BLAST docu-

Ort	Homologies	Nodes side 1	Nodes side 2
DM-HS	19755	7735	10256
DM-MM	15689	7475	8544
DM-CE	11661	6312	5177
DM-SC	7494	4679	3430
MM-HS	23843	11858	13491
MM-CE	11412	7004	5272
MM-SC	7997	5055	3543
CE-HS	14743	5434	8602
SC-HS	9812	3651	6030
SC-CE	5570	3131	3503

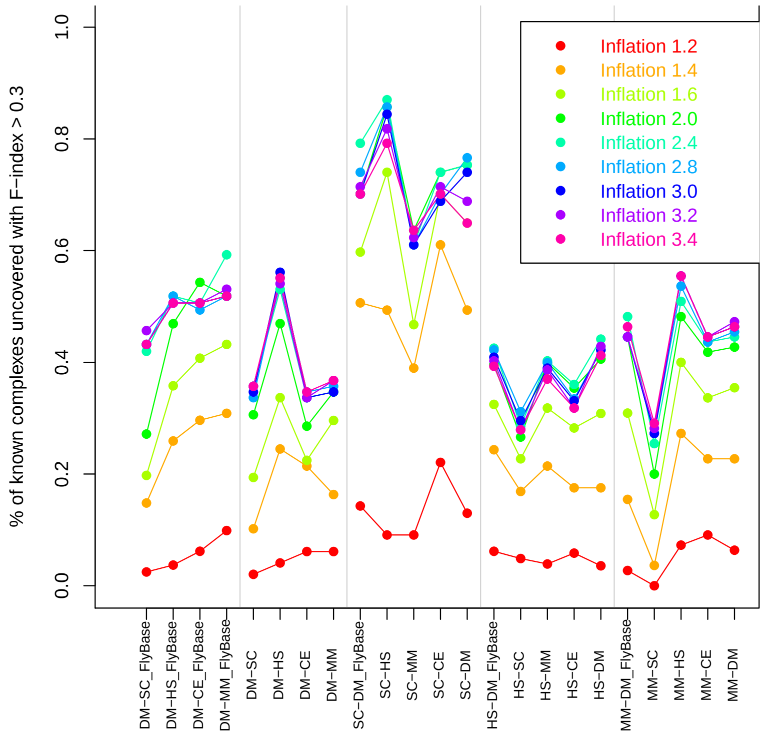
Table 2.7: Statistics of homology data downloaded from DIOPT.

Species	Dataset	|Raw complexes	Complexes with at least 4 proteins
D. Melanogaster	DPIM	554	153
H. Sapiens	CORUM	1349	606
M. Musculus	CORUM	289	248
S. Cerevisiae	CYC2008	408	-

Table 2.8: Statistics of known complexes datasets used in the assessment.

mentation have been used.

The alignments involving the S. cerevisiae were evaluated using as gold standard the complexes in CYC2008 [38], a comprehensive catalogue of 408 yeast protein complexes derived from small scale experiments and literature mining. For the alignments involving the D. melanogaster DPIM [57] was considered, while for H. sapiens and M.Musculus the CORUM database [39] was selected. As done for the assessment of AlignNemo, the complexes smaller than 4 proteins were ignored (see Table 2.3.2).



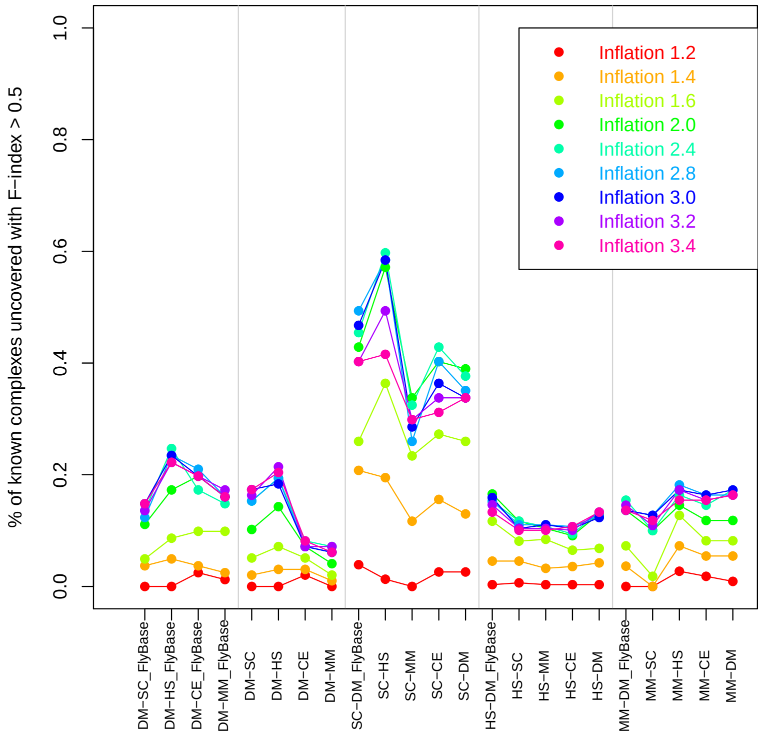


Figure 2.6: Performance of AlignMCL in terms on fraction of protein complexes recovered with F-index ≥ 0.3 and ≥ 0.5, respectively.

Parameter tuning

The most important parameters of AlignMCL are the *pruning threshold*, used in the alignment graph pruning step and already described in the previous section, and the *inflation*, used in the MCL-based clustering algorithm. The impact of varying the pruning threshold has been widely tested, and it has been shown that it does not affect the outcome of the alignment if kept between 0.3 and 0.7. [30].

In this section the influence of the inflation level is evaluated. The best results, in terms of precision and robustness in the identification of already known complexes, are achieved when the inflation ranges between 2.4 and 3.0. The MCL algorithm proves to be quite stable within this range. Inflation levels below 2.4 determine a quick degradation of the quality of the solutions; performance decreases slowly when the inflation increases beyond 3.0. An effective representation of this behavior is presented in Figure 2.6.

Comparison with other algorithms

In the following evaluation an inflation level of 2.8 has been used for AlignMCL. Only the PINs from i2d have been considered in this comparison. The number of alignments produced by the various algorithms are reported in Table 2.9.

Figure 2.7 compares the different algorithms in terms of F-index. NetworkBlast has been omitted since its results were always inconsistent. The algorithms have a similar trend across the different datasets, showing a certain degree of agreement on the quality of the solutions. The best results are achieved in the alignments involving the yeast organism.

The comparison in Figure 2.7 is useful for summarizing the results, but might

Alignment	Number of solutions			
	MaWish	AlignMCL	NetAligner	NetworkBlast
DM-SC	574	793	121	2030
DM-HS	976	1746	119	-
DM-CE	289	1071	72	235
DM-MM	268	1316	30	683
SC-HS	957	826	156	-
SC-MM	368	1159	15	1864
SC-CE	426	2221	62	707
HS-MM	975	2361	51	-
HS-CE	848	871	76	-
MM-CE	188	1324	7	226

Table 2.9: Number of solutions produced by each algorithm in thew different alignments.

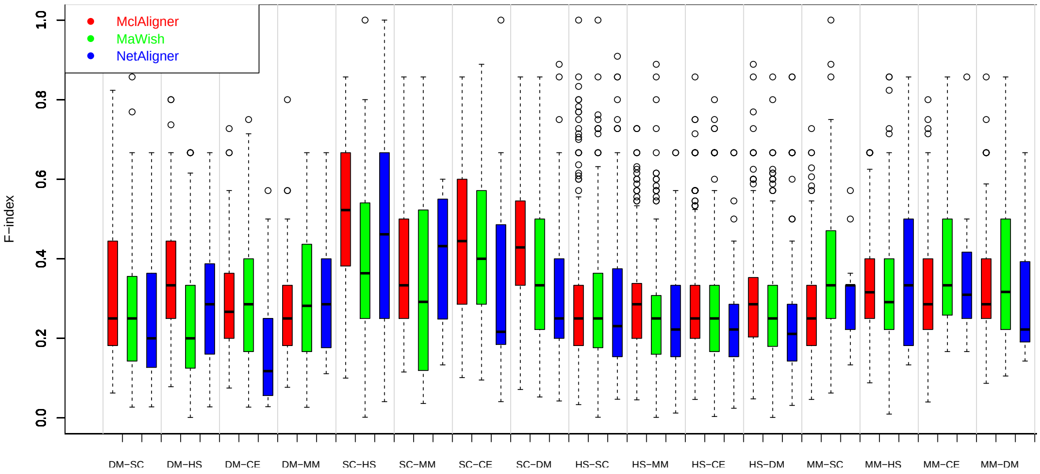


Figure 2.7: Comparison of alignment algorithms in terms of F-index. For each algorithm, the corresponding boxplot summarizes the quality in matching the known complexes. Known complexes not overlapping with any solution for a given algorithm have been ignored.

be somehow misleading. In fact, even though MaWish and NetAligner seem to be more accurate, in almost all the alignments AlignMCL uncovered an higher number of complexes with good quality respect to the other tools. Table 2.10 reports the number of high quality solutions for each algorithm. AlignMCL outperforms MaWish and NetAligner in most of the cases.

Further investigations show that while NetworkBlast works well on sparse networks, its performance decreases quickly as the input PINs become more reliable and complete. The worst scenarios are the alignments involving the human ppi. In this case NetworkBlast has not been able to conclude its work in one week, on a cluster equipped with Intel Xeon processors. Indeed, after considering the alignment graph built by this tool, it is clear that the problem resides in the too relaxed definition of alignment graph, that leads to the an extremely dense alignment graph on which the mining algorithm eventually hangs. NetworkBlast is not able to discern between the weak edges and the good ones, in a scenario where each node appears to be connected to each other, and ends up extracting cliques of 15 nodes, the maximum size allowed by the algorithm, completely at random. For example, when aligning yeast and fly, NetworkBlast produces an alignment graph with almost 1 million edges between 7000 nodes. The improved definition of alignment graph developed in this work takes into account the problems reported for NetworkBlast.

On the other side MaWish, that usually generates small solutions due to the strict definition of alignment graph, performs well on the newest networks, while it was less convincing on more sparse networks. MaWish proved to be the fastest algorithm, completing each alignment in less than 5 minutes and requiring a reduced amount of memory. NetAligner produced few solutions, but many of them

Alignment	Number of known complexes hit (F-index ≥ 0.3)			
	MaWish	AlignMCL	NetAligner	NetworkBlast
DM-SC	24	**33**	15	1
DM-HS	26	**54**	16	-
DM-CE	21	**34**	4	0
DM-MM	14	**35**	2	0
SC-HS	45	**66**	28	-
SC-MM	20	**47**	3	1
SC-CE	43	**54**	6	0
SC-DM	36	**59**	16	0
HS-SC	90	**96**	63	-
HS-MM	80	**123**	29	-
HS-CE	85	**103**	27	-
HS-DM	97	**130**	24	-
MM-SC	39	**30**	7	0
MM-HS	52	**59**	21	-
MM-CE	37	**48**	6	1
MM-DM	38	**50**	5	0

Alignment	Number of known complexes hit (F-index ≥ 0.5)			
	MaWish	AlignMCL	NetAligner	NetworkBlast
DM-SC	6	**15**	6	0
DM-HS	11	**19**	5	-
DM-CE	**7**	**7**	2	0
DM-MM	**7**	6	1	0
SC-HS	25	**45**	20	-
SC-MM	11	**20**	2	0
SC-CE	22	**31**	5	0
SC-DM	18	**27**	9	0
HS-SC	32	**34**	32	-
HS-MM	25	**34**	8	-
HS-CE	24	**33**	5	-
HS-DM	22	**39**	11	-
MM-SC	**16**	14	2	0
MM-HS	18	**20**	11	-
MM-CE	**18**	**18**	3	0
MM-DM	**19**	18	5	0

Table 2.10: Number of known complexes recovered by the different algorithms.

proved to be of extremely high quality. The trends of MaWish and NetworkBlast show how influent is the definition of alignment graph on the outcome of the alignment.

2.4 Extensive assessment

In previous sections, the ability of the different alignment algorithms to uncover known modules have been tested. However, few alignments have been considered, overlooking on stability and general applicability. This is indeed a common trend of many other works, where alignment algorithms are compared on the basis of few cases. In this section, some important aspects concerning the assessment of alignment algorithms are discussed, with some considerations on the input datasets and the evaluation strategies.

2.4.1 Considerations on the input datasets

A recent trend in interactomics is the integration of multiple data sources to build more complete and reliable datasets [58]. It is therefore important to evaluate the alignment algorithms on up-to-date datasets, not only for their higher quality, but also to verify the ability of the implementations to deal with the size of current networks.

On the other side, evaluating the performance of an algorithm over a dataset of different PINs is fundamental to avoid overfitting, and prove its general adaptability. Indeed, a big challenge in the analysis of PINs is presented by the different characteristics of the input data. Some PINs are rather complete, while others count fewer interactions. More importantly, some PINs contain only interactions uncovered in high-quality experiments, and others include data from high-throughput low-quality experiments. An interesting aspect is that several PPI datasets, with different levels of reliability and completeness, exist for the same organisms (see, for instance, Table 2.4.3). The stability of an algorithm can be studied by considering whether its performance significantly change across alignments involving different PINs of the same organisms.

As rules of thumb, it is therefore preferable to

- assess an algorithm on several different alignments, to prove its general applicability,
- consider different PINs of the same organism, to verify the stability of the algorithm.

2.4.2 Assessment strategies

The purpose of an alignment is to identify evolutionary conserved modules between different species. Thus, the ideal assessment consists in verifying whether evolutionary converved modules are effectively grouped in a single solution. However, not all the modules are currently known for all the organisms, and some modules do not have a (known) counterpart in other species, Different strategies can be employed to evaluate an alignment, based on available knowledge. For instance, given a set of known modules for one of two aligned organisms, it is possible to check the ability of the alignment to recapitulate it.

More in general, an alignment can be analyzed from an intra-species or an inter-species point of view. An intra-species agreement tells, for instance, whether the proteins collected in a single solution belong to a known module. Instead, an inter-species assessment tries to understand whether the putative orthologs within a single cluster share some functional roles. Even though the inter-species perspective might seem more significant, an intra-species analysis might be preferable when there is not much information available on one of the two aligned species. Indeed, most of the works assessed the algorithms performance relying mainly on intra-species analysis [13] [7] [18]. A first evaluation of the solutions in terms of semantic similarity have been proposed in [30]. In general, it is a good practice to validate an alignment algorithm in both the senses.

Intra-species assessment: comparison with known modules

Given a solution and a known module, there are several ways to compare them. For instance, one can look for a summary agreement, considering the overlap between the sets of proteins participating to the solution and the module, respectively. A finer measure would consider the internal connections between the proteins as well.

In this work the simpler strategy has been used, since currently available datasets of known modules are not usually annotated with fine information on their internal topology. Moreover, solutions provided by current alignment algorithms are often not enough specific to reconstruct the internal topology with high quality, mainly due to the noise of input data. Therefore, a fine comparison might not be suited in most of the cases, and indeed most of the previous works relied on the simpler strategy.

The quality of the overlap between two sets can be expressed through precision (π), recall (ρ) and F-index, already introduced in this chapter for the assessment of AlignNemo. In the following analyses, each known complex of a species is compared to all the solutions of a given alignment, and the solution with highest F-index is selected as best match. The ability of an algorithm to recapitulate known complexes can be derived by the comparison with known modules.

Inter-species assessment: employing semantic similarity

In general, modules are groups of interacting proteins that share common functions or play similar biological roles. For instance, a biological pathway is a number of biochemical steps, linked together, that perform a process inside cells.

GO functional enrichment [59] has been used to evaluate the significant presence of common functions in the solutions. Functional enrichment generally considers proteins from the same organism, and the inter-species comparison is usually performed by checking for common enriched functions. This approach has some drawbacks, since in general similar functions are considered as not corresponding at all. Moreover, there are some biases introduced by the size of the assessed sets (see [59] for a complete discussion).

To address these problems, the use of semantic similarity (SS) has first been proposed in [30]. SS measures are able to quantify the functional similarity of pairs of proteins/genes, comparing the GO terms that annotate them. Thus, there are no constraints on the minimum set size [40]. Since proteins within the same pathway are involved in the same biological process, they are likely to have high semantic similarity. In a similar way, protein belonging to the same complex are likely to have similar biological roles, and therefore they should have high semantic similarity. The idea is to verify whether the semantic similarity between the proteins within a solution is significantly higher than random expectation.

Inter-species Semantic Similarity Given an solution S_k, its inter-species semantic similarity $SS_i(S_k)$ is defined as

$$SS_i(S_k) = \frac{\sum\limits_{x_i \in S_k^1} \sum\limits_{y_j \in S_k^2} SS(x_i, y_j)}{|S_k^1||S_k^2|}$$

where $SS(x_i, y_i)$ is the semantic similarity between proteins x_i and y_j.

Note that in general $|S_k^{1,2}| \leq |S_k|$, since a protein can appear in more than one association.

The inter-species semantic similarity can be directly used to compare the quality of different solutions and, by extension, algorithms. It is worth noting, however, that smaller solutions are more likely to have higher SS scores [30] [60].

Another interesting possibility consists of comparing the real alignments against random ones, in order to prove their statistical significance. Given a solution S_i, a first null hypothesis H_0^1 to test is: *the inter-species semantic similarity $SS_i(S_i)$ is drawn from the background distribution*, where the background distribution can be estimated from the SS_i of random solutions. As usual, the hypothesis can be rejected if the results p-value is lower than a given threshold, commonly fixed to 0.05 or 0.001. This approach is useful to prioritize or filter the solutions in a post-processing step. However, if the purpose is to validate the entire alignment, it presents two issues. First, instead of a single p-value, many are returned, and merging them is not straightforward. Second, the p-values need to be corrected for multiple hypothesis testing.

Let $A = \{A_1, A_2, ...A_n\}$ be the set of solutions of a given alignment problem, and $SS_i(A) = \{SS_i(A_1), SS_i(A_2), ...SS_i(A_n)\}$ as the semantic similarity profile of A. The null hypothesis H_0^1 can be replaced by H_0^2: *the inter-species semantic similarity profile $SS(A)$ of alignment A has the same distribution of the similarity profiles of random alignments*. The new formulation produces a single p-value, avoiding the problems of the first one. The following strategy is used to test H_0^2:

1. Build the size profile $Size(A) = (|A_1|, |A_2|, ..., |A_n|)$
2. Generate 1000 random solutions R_i such that $|R_i| = n$, and $Size(R_i) = Size(A)$. In other words, each R_i is a group of random sets $\{R_{i,j} \ : \ |R_{i,j}| = |A_i|\}$
3. Calculate $SS(A)$, and $SS(R_i)$ $\forall R_i$
4. Merge all the $SS(R_i)$ into a single vector $SS(R)$
5. Compare $SS(A)$ and $SS(R)$ with the Mann-Whitney test (or Wilcoxon rank-sum/unpaired test) to estimate the p-value $P_SS(A)$
6. Reject the null hypothesis H_0^2 if $P_SS(A) \leq 0.05$

A non-parametric test is used because, as verified, the distribution of inter-species semantic similarity scores does not follow a normal distribution.

Intra-species assessment with semantic similarity

As a final remark, semantic similarity can be used to assess the intra-species agreement as well. Given a solution S_k, the intra-species semantic similarity of S_k is separately defined on the two species as

$$SS_1(S_k) = \frac{\sum\limits_{x_i \in S_k^1} \sum\limits_{y_j \neq x_i \in S_k^1} SS(x_i, y_j)}{|S_k^1||S_k^1 - 1|}$$

and

$$SS_2(S_k) = \frac{\sum\limits_{x_i \in S_k^2} \sum\limits_{y_j \neq x_i \in S_k^2} SS(x_i, y_j)}{|S_k^2||S_k^2 - 1|}$$

2.4.3 Results

Following the indications of the previous section, AlignMCL has been compared to MaWish [14], NetAligner [27] and NetworkBlast [61]. All the algorithms require two PINs and a set of putative orthologs as input data. Graemlin [7] and Graemlin 2.0 [18] have not been considered because they require additional data to learn some alignment parameters, and these are not readily available for all the organisms. The assessment dataset count 89 alignments between several PINs of 5 different species. The comparison not only demonstrates that AlignMCL outperforms the other algorithms, but also that it is more stable when different PINs of the same organisms are used.

As anticipated, NetworkBlast is not able to deal with the size and complexity of current PINs. For all the alignments, when able to conclude the computation (on a Linux CentOS Cluster, equipped with two Intel Xeon processors and 8Gb of RAM), it produced few random solutions. For this reason, NetworkBlast's results are not shown in the comparisons.

Input Dataset

The assessment dataset includes 15 PINs from five of the most studied species: Drosophila Melanogaster (fly), Saccaromices Cerevisiae (yeast), Homo Sapiens (human), Cenhorabditis Elegans (worm), and Mus Musculus (mouse) - for a total of 89 different alignments. Up-to-date PINs have been downloaded from i2d [52],

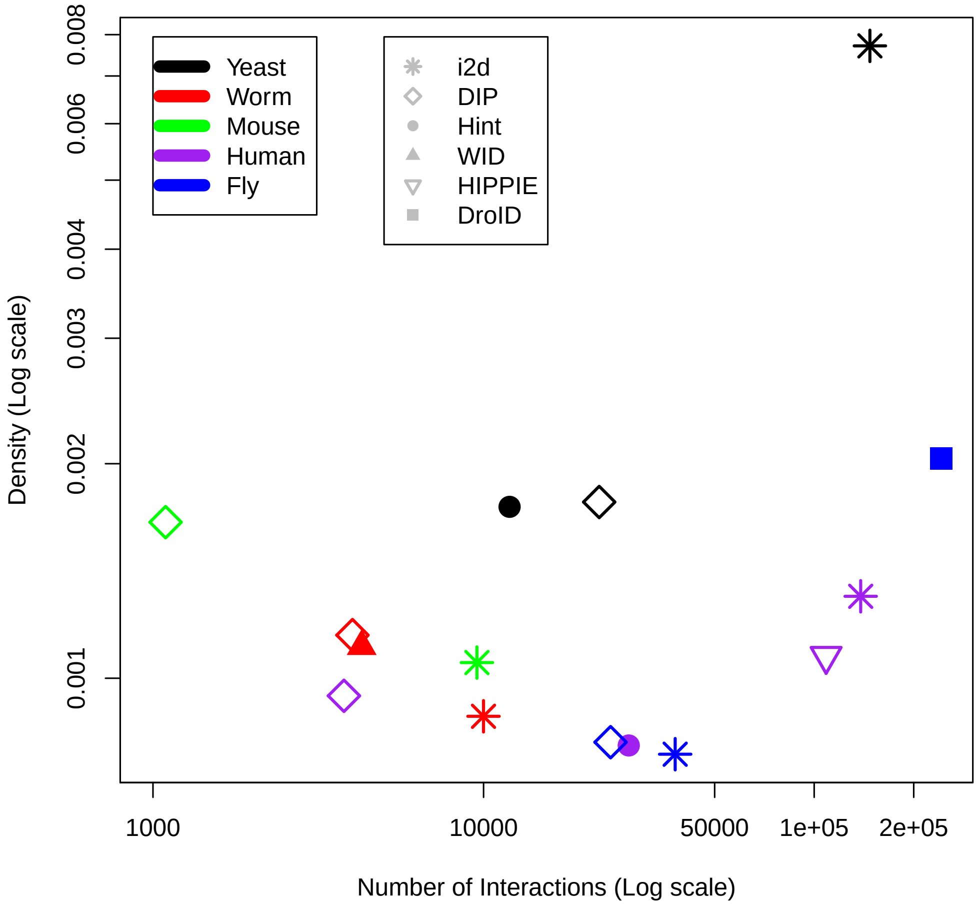


Figure 2.8: Log-plot of the density and the number of interactions of each dataset.

Species	Dataset	Release	Proteins	Interactions
Drosophila Melanogaster (Fly)	DIP	2012.05.18	7718	24220
	DroID	2011.11	15428	242187
	i2d	v. 1_9	9854	37979
Homo Sapiens (Human)	DIP	2012.05.18	2830	3782
	HIPPIE	2012.04.23	14224	108661
	Hint	2012.08.09	8265	27487
	i2d	v. 1_9	14567	138258
Saccharomyces Cerevisiae (Yeast)	DIP	2012.05.18	5033	22377
	Hint	2012.08.09	3712	11985
	i2d	v. 1_9	6182	147408
Mus Musculus (Mouse)	DIP	2012.05.18	1149	1092
	i2d	v. 1_9	4261	9547
Caenorhabditis Elegans (Worm)	DIP	2012.05.18	2643	4013
	WID	2012.08.10	2779	4279
	i2d	v. 1_9	4755	9995

Table 2.11: Statistics of PINs used in the assessment. All the proteins have been mapped to Uniprot AC ids.

DroID [53], Hint [62], HIPPIE [34], WID [63], and DIP [33] databases. Statistics for the 15 PINs are reported in Table 2.4.3. A comparison in terms of number of interactions and network density is shown in Figure 2.8, where network density d is a global graph statistics defined as

$$d(G) = \frac{2|E|}{|V|(|V|-1)}$$

PINs vary a lot in terms of number of interactions, even for the same organism. For instance, human PINs range from the ~4000 (DIP) to ~140000 interactions (i2d). A similar thend is observable for yeast and fly. DIP networks count the lowest numbers of interactions, with an exception in the yeast case. On the contrary, i2d networks are the most complete, but for fly. Density values range between

Species	Homologies	Proteins Species 1	Proteins Species 2
Fly-Human	51238	11078	14615
Fly-Mouse	34781	10667	12932
Fly-Worm	16732	7177	5175
Fly-Yeast	10743	6094	3743
Mouse-Human	64679	19555	18757
Mouse-Worm	18733	8840	5461
Yeast-Mouse	12360	3955	7090
Worm-Human	27048	5633	10895
Yeast-Human	17884	4088	8891
Yeast-Worm	5756	2850	3283

Table 2.12: Statistics of homology data downloaded from DIOPT. Only proteins with an Uniprot AC have been considered.

0.001 and 0.002, with the only exception of yeast i2d network (~0.008). For human, mouse and worm, densities are comparable between the different PINs. In general, the lower number of proteins in DIP's networks is responsible for their slightly higher levels of density.

The Integrative Ortholog Prediction Tool (DIOPT) [54] waas used to build a comprehensive set of putative orthology associations between the different organisms (Table 2.4.3). Some algorithms (i.e. NetAligner) require BLAST data in addition or in substitution of data provided by DIOPT. The complete sequence dataset for the five species have been downloaded from the NCBI website [56], and a BLAST sequence alignment between the proteins of the different species has been performed. The standard parameters reported in BLAST documentation were used.

Species	Dataset	Raw Complexes	Merged complexes
Human	CORUM	1685	606
Yeast	CYC2008	408	345
Mouse	CORUM	439	248
Fly	DPIM (DroID)	556	153

Table 2.13: Statistics of datasets of known protein complexes used as gold standard. All the dataset are updated to 2012. For the fly, out of the 556 complexes, only the 153 with a functional p-value lower than 10^{-3} in the original work [57] were considered.

2.4.4 Intra-species assessment

For each species a dataset of known complexes was selected as benchmark dataset. To evaluate the results for the alignments involving the yeast, the 408 complexes in CYC2008 [38], a comprehensive catalogue of complexes derived from small scale experiments and literature mining, were considered. For fly, the 556 complexes in the DPIM dataset [57] were used. Finally, for human and mouse the CORUM database [39] was selected (1685 and 439 complexes, respectively).

Within each dataset many complexes with similar biological functions, and highly overlapping with each other, were identified. This might lead to a biased evaluation, since a solution might overlap with more than a known complex, and therefore might be counted more than once. Moreover, these overlapping complexes are often quite small (2-4 proteins). To address this isse, highly overlapping complexes have been clustered together. More in detail, FastSemsSim [44] has been used to evaluate a quantitative measure of the functional similarity between the overlapping complexes. Afterwards, complexes have been clustered together using ClusterMaker [64]. This process produced a smaller number of complexes, as shown in Table 2.4.4. The performance of the algorithms were assessed also on the original sets of complexes. The results are less clear, but lead to the same conclusions. The only significant difference is that some solutions encompass different small complexes (2-4 proteins), partially overlapping. As previously done, protein complexes and solutions of size 2 were ignored.

Figure 2.9 presents a broad comparison on some of the alignments. AlignMCL outperforms MaWish and NetAligner in most of the cases. As positive note, the algorithms show a similar trend across the different datasets, showing a certain degree of agreement on the solutions. The best results are achieved by all the algorithms in the alignments involving denser PINs. When PINs from DIP database are used, instead, all the algorithms fail to uncover many complexes.

A more specific comparison on the fly-yeast alignments is proposed in Figure 2.4.4, where the F-indexes of best matching solutions are compared. Considering the top 10 matches in the comparison of DroID fly and i2d yeast PINs, MaWish provided the best results both on fly and yeast sides. In general, however, AlignMCL features an higher number of high-quality solutions (with 50 solutions matching a complex with F-index ≥ 0.6 on the yeast side, against the 21 of MaWish). A more interesting aspect is the stability of the algorithms. AlignMCL provides alignments with similar quality regardless the considered PINs. The quality of MaWish results, instead, is more variable, with few consistent matches for the alignment of DroID fly and i2d yeast PINs. Similar conclusions can be drawn from the alignments of other species.

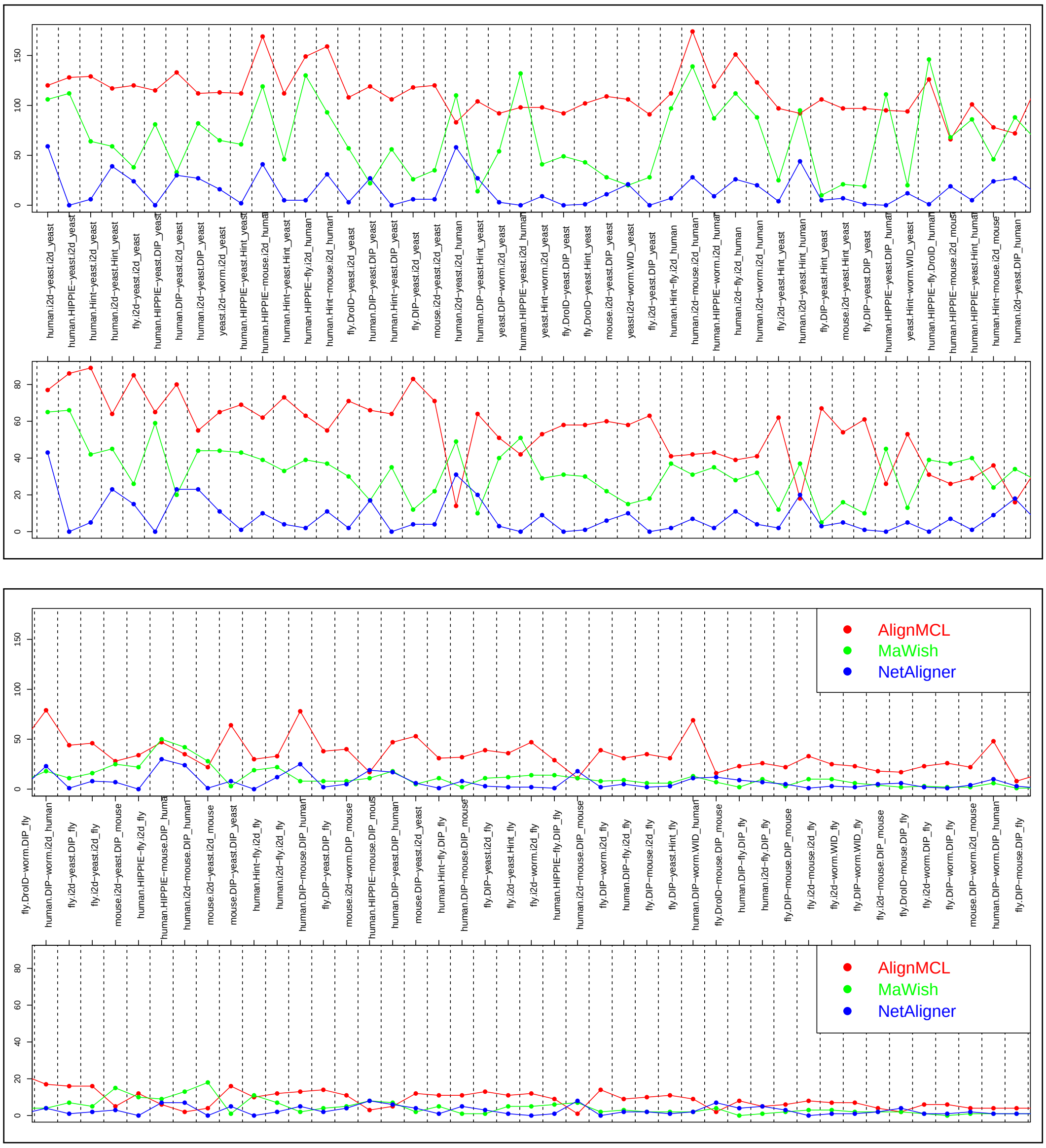


Figure 2.9: Comparison of alignment algorithms in terms of intra-species similarity. Each algorithm is evaluated separately on both the aligned species. The first and third rows report the number of known complexes uncovered with an F-index ≥ 0.3. For the second and fourth rows, instead, only matches with F-index ≥ 0.5 were selected.

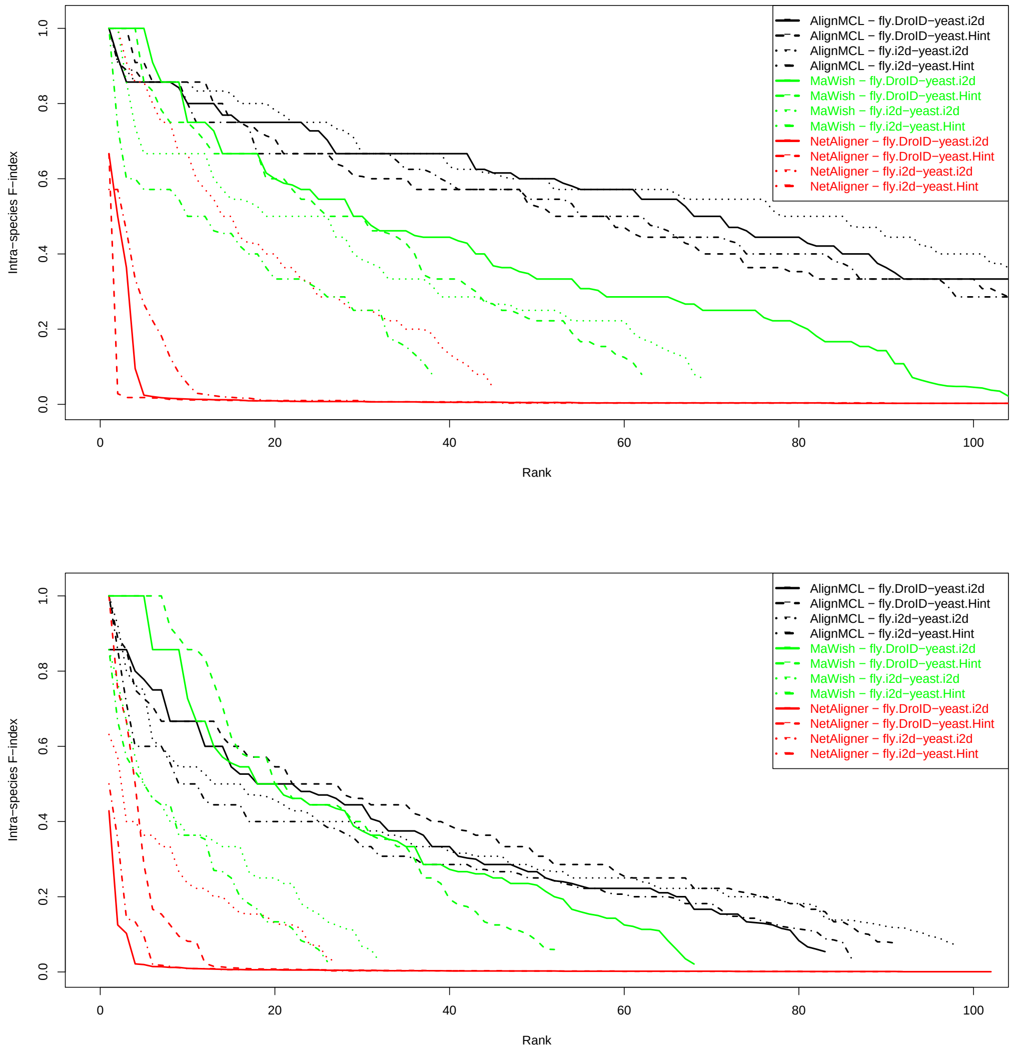


Figure 2.10: Comparison of alignment algorithms on fly-yeast alignments. The graphics show the F-index of the top 100 solutions (ranked by F-index). The upper figure represents the assessment on the yeast side, while the lower figure focuses on the fly side.

2.4.5 Inter-species assessment

The statistical significance of alignments provided by AlignMCL was verified by employing the strategy based on the inter-species SS (hypothesis H_0^2). The Gene

Ontology and GO Term annotations were downloaded from the Gene Ontology website on October 2012. FastSemSim [44] was used to evaluate the inter-species semantic similarity for all the solutions. SimGIC [65] was selected as semantic measure. The null hypothesis H_0^2 has be rejected for all the alignments, but yeast-DIP vs mouse-DIP, and yeast-DIP vs fly-DIP. This is in line with the results of the intra-species assessment, where DIP PINs produced the worst results.

As a second assessment, the SS scores of the solutions of AlignMCL and the other algorithms have been compared. In Figure 2.4.5 we propose a comparison of human-fly and fly-yeast alignments. AlignMCL produces alignments with higher inter-species semantic similarity. In terms of stability, AlignMCL and MaWish produce results with a similar quality across the different input PINs. NetAligner, instead, is more sensible to the input data. Similar conclusions can be drawned from the other comparisons.

2.4.6 Integrated comparison

The results of inter- and intra-species analyses were afterward merged in an integrated assessment. The scatter plot in Figure 2.12 compares the algorithms both in terms of inter-species and intra-species alignment quality. The number of known complexes recapitulated with F-index ≥ 0.5 has been used as indicator of the intra-species quality of the alignment. The inter-species quality, instead, is represented by the number of solutions with F-index ≥ 0.3. AlignMCL's alignments concentrate on the top-right area of the figure. Many alignments overlap, since their quality is similar for the alignments involving the same organisms (effect observable also when considering the plateau in Figure 2.9, for instance). The more

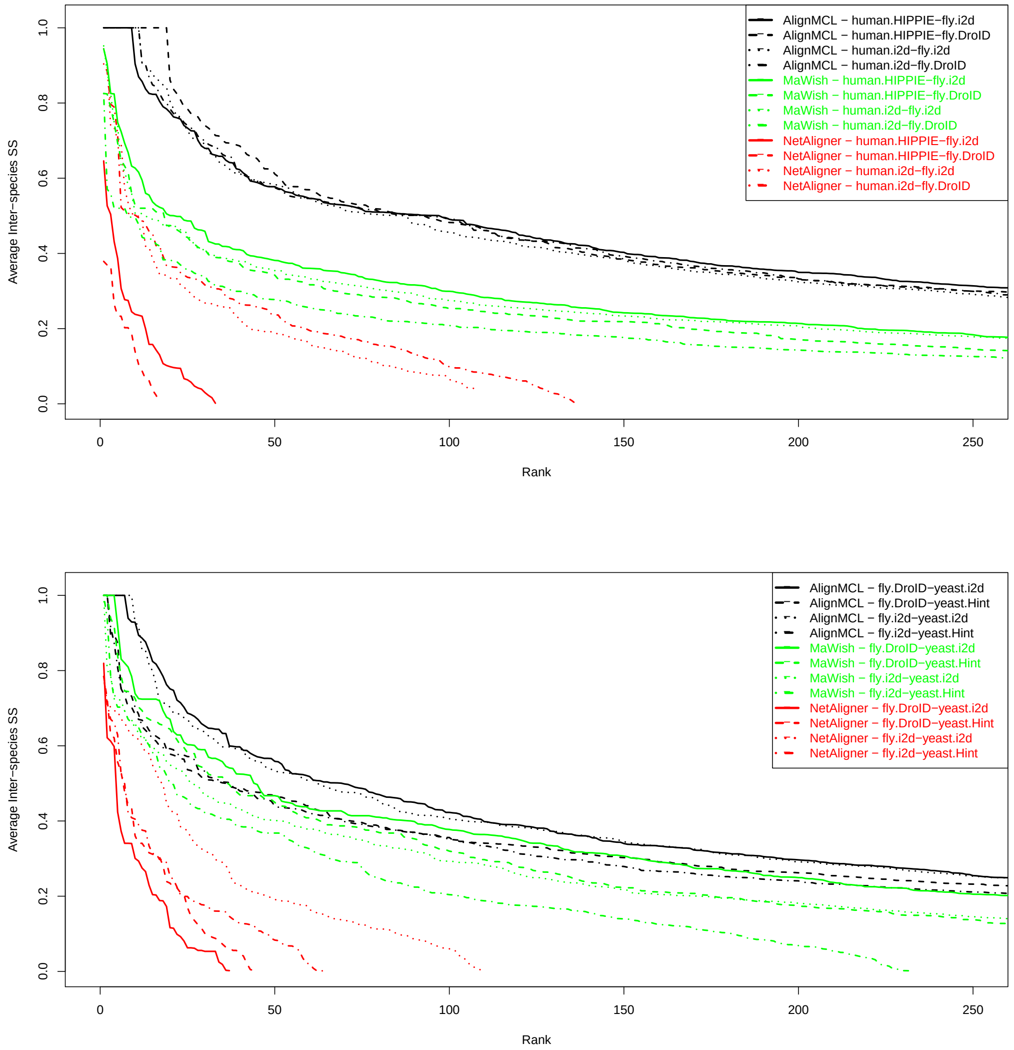


Figure 2.11: Comparison of alignment algorithms on human-fly and fly-yeast alignments. The graphics show the average intra-species SS for the top 250 solutions (ranked by SS).

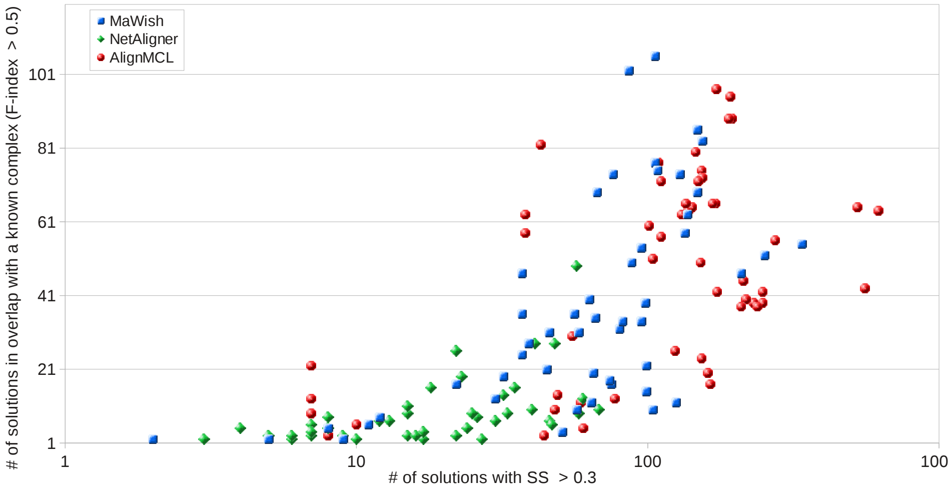


Figure 2.12: Comparison of SS and F-index results

variable results of MaWish are comparable, in some cases, to AlignMCL's ones, confirming that MaWish behaves well on some networks, but has some problems in dealing with the sparsest ones.

A finer integrated comparison is proposed in Figures 2.13 and 2.14. In the first, all the solutions with either a good inter-species SS or a good intra-species F-index are selected. In the latter, instead, only the solutions with good statistics in both the inter-species and intra-species comparisons are considered. Imposing strict quality constraints (i.e. $SS_i \geq 0.5$) on both intra- and inter-species analyses drastically reduces the number of solutions selected. In terms of solutions AlignMCL is more stable, while MaWish shows a greater variance. This is in agreement with what already observable, for instance, in Figures 2.4.4 and 2.4.5.

The combined comparisons highlight a common trend for AlignMCL and MaWish (see Figures 2.13 and 2.14). For fly-mouse, human-mouse, human-worm, human-fly, and yeast-worm alignments, AlignMCL's and MaWish's results improve and worsen following a similar pattern. A weaker but still observable correlation is noticeable for fly-worm and mouse-yeast alignments (especially in Figure 2.13). In fly-yeast alignments, and on a lower degree in human-yeast alignments, no correlation is apparent.

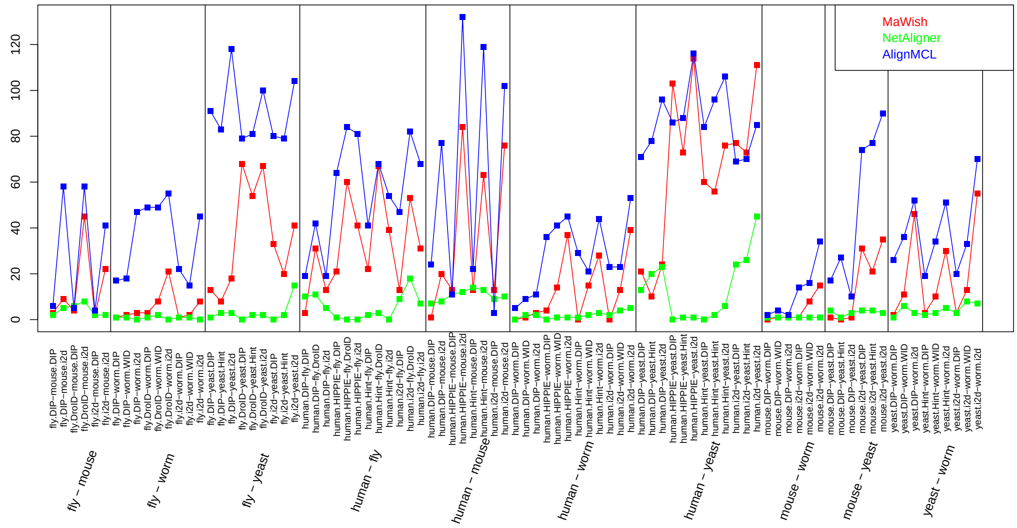


Figure 2.13: Number of solutions with inter-species SS ≥ 0.5 or intra-species F-index ≥ 0.5 in one of the two species.

2.5 Conclusions

In this chapter, the step by step development of AlignMCL was thoroughly described. First, the characteristics of current PINs that are problematic in the network alignment framework were critically discussed. Then, a new model of alignment graph was designed and implemented in AlignNemo, addressing in particular the aspect affecting previously existing alignment algorithms. Finally, AlignMCL, a local network alignment algorithm based on AlignNemo's model of alignment graph, was developed. Its mining strategy based on Markov Clustering is able to identify conserved modules without limiting constraints on the topology

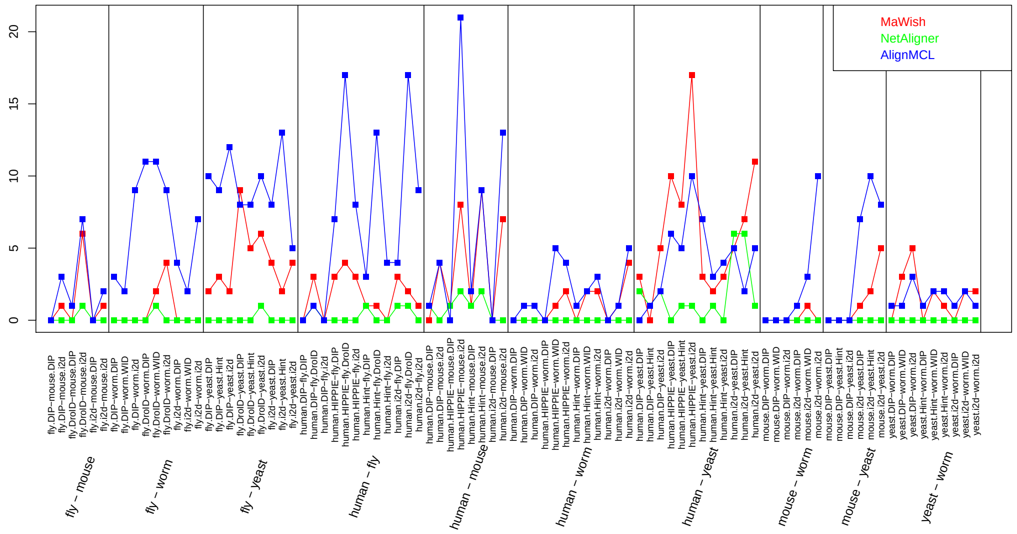


Figure 2.14: Number of solutions with inter-species SS ≥ 0.5 and either intra-species F-index ≥ 0.5 or intra-species SS ≥ 0.5 in one of the two species.

of the solutions.

The performance and stability of AlignMCL have been extensively tested. In particular, the comparison allowed to identify a common behaviour between AlignMCL and MaWish, suggesting that the algorithms show a common sensitivity to the same type of noise.

This work can be further extended in different directions. First, the MCL engine can be combined with global network alignment algorithms. Even though the main purpose of global network alignment is to evaluate the best superimposition of input PINs, it is possible to reinterpret their output as an alignment graph, and consequently apply MCL to extract the modules. An advantage of this strategy is that many global network alignment algorithms do no require putative orthologs as input data, and could therefore provide less constrained results.

The big issue with current PINs is the amount of missing or wrong interactions. Different PINs can be merged in a single network to increase its completeness. An alternative approach consists of first aligning all the PINs of two organisms, and then derive a consensus between the pairwise alignments. This strategy might be preferable in some cases, since different networks might have different levels of false positives and false negatives, or be biased in different ways. Moreover, an algorithm that reports similar solutions starting from different datasets is likely to be robust and reliable, with overlapping solutions more likely to be correct. Obviously, a consensus is unlikely to emerge if the alignment algorithm used is not robust, such as in the case of NetAliger. We believe AlignMCL fulfills this requirement, and might therefore be used in this strategy.

Finally, it should be noted that the MCL implementation used in this work performs a hard-clustering of the alignment graph. It means that solutions are not allowed to overlap with each other, or in other words, that the solutions form a partition of the graph. This might not be the optimal solution. For instance, an hard-clustering approach might merge together the cores of two overlapping protein complexes featuring topological properties in agreement with the core and attachment model [4]. Thus, in some cases, a soft-clustering approach that allows solutions to overlap might be preferable. In MCL, the hard-clustering constraint is not related to the flow simulation process: raw solutions emerging after the inflation-expansion steps can overlap, and such overlap is removed in a postprocessing step. Indeed, soft-clustering variants of MCL have recently been proposed [66], and might be employed in an extension of AlignMCL.

CHAPTER 3

Master Regulator Analysis

3.1 Introduction

Understanding complex polygenetic phenotypes - stage of differentiation, disease state, responsiveness to exogenous perturbations and so on - requires the identification of sets of related genes associated with phenotipic changes. This challenging problem led to the development of high performance experimental procedures and analytical methods. On the experimental side, techniques such as high-throughput sequencing and gene/protein profiling have transformed biological research by enabling comprehensive monitoring of a biological system. On the analytical side, most of the current approaches to the analysis of high-throughput data typically yield a list of differentially expressed genes or proteins. This list is extremely useful in identifying genes that may have a role in a given phenomenon or phenotype. In many cases, however, the list of differentially expressed genes fails to provide mechanistic insights into the underlying biology of the condition being studied [67]. Thus it is important to consider, instead of individual genes, the expression of sets of genes functionally related, for instance those participating to the same pathways.

Analyzing high-throughput molecular measurements at the level of functionally related groups of genes is very appealing for two reasons. First, grouping thousands of genes, proteins, and other biological molecules in the modules in which they are involved reduces the complexity of the experiment. Second, identifying active pathways that differ between two conditions can have more explanatory power than a simple list of different genes or proteins.

There are two different components in these system-level studies: on one side (i) identifying gene sets that were not previously known to be related, and on the other (ii) determining, among a known collection of sets, the ones that are related to a specific phenotype [68]. The latter component is generally called knowledge base-driven pathway analysis [67]. It identifies, among a group of a-priori known modules, those that correlate with condition-specific gene expression patterns. Despite the name, these modules are not necessarily biological pathways. In the following, to avoid misunderstandings, the terms pathways and functional modules will be used interchangeably to refer to sets of functionally related proteins. Since the knowledge base-driven pathway analysis requires a list of a-priori known modules, researchers have developed a large number of knowledge bases describing biological processes, components, and pathways in which individual genes and proteins are known to be involved. In the last decade different approaches to the base-driven pathway analysis have been proposed, classified and evaluated to highlight their strenghts and weaknesses [69, 70, 71, 72, 73, 67, 68, 59, 74, 75, 76, 77, 78]. It is not a purpose of this thesis to provide a solid review of the existing methods. However, for a better introduction to the problem addressed, a brief overview of the chronological development of such methods, as proposed in [67], is here reported and extended.

Knowledge base-driven pathway analysis methodologies can be divided in three categories with respect to their temporal order of appearance:

- First Generation: Over-Representation Analysis (ORA)

Over-representation analysis (ORA) approaches, alternatively called Singular Enrichment Analysis (SEA) [59], statistically evaluate the fraction of genes in a particular pathway that show changes in expression. The general pipeline is as follows: first, a list DE of differentially expressed genes is created using a certain threshold or criterion. Then, for each pathway, the genes in DE that are part of the pathway are counted. Next, every pathway is tested for over- or under-representation in the list DE of input genes. The most commonly used tests are based on the hypergeometric, chi-square, or binomial distribution. Background distributions are estimated by drawing random sets of genes.

This first approach clearly resembles other enrichment techniques working on different types of data, such as the GO Term Enrichment, aimed at identifying the GO Terms enriched in a given set of genes/proteins [59].

- Second Generation: Functional Class Scoring (FCS) Approaches

The hypothesis of functional class scoring (FCS) is that although large changes in individual genes can have significant effects on pathways, weaker but coordinated changes in sets of functionally related genes can also have significant effects. The general analysis pipeline consists of three steps [71]: first, a gene-level statistic is computed using the molecular measurements (i.e. expression profiles). Second, the gene-level statistics for all genes in a pathway are aggregated into a single gene set-level statistic. Finally, the statistical significance of the set-level statistic is evaluated by comparing it to a background distribution.

- Third Generation: Pathway Topology (PT)-Based Approaches

A large number of publicly available pathway knowledge bases provide information beyond simple lists of genes for each pathway. ORA and FCS methods consider only the number of genes in a pathway and gene co-expression to identify significant pathways, and ignore the additional information available from these knowledge bases. PT-based methods, alternatively classified as Modular Enrichment Analysis (MEA) methods in [59], are essentially the same as FCS methods in that they perform the same three steps as FCS methods. The key difference between the two is the use of pathway topology to compute gene-level statistics.

3.2 Gene Set Enrichment Analysis (GSEA)

Gene Set Enrichment Analysis (GSEA) is an FCS algorithm proposed by Subramanian et al. [79] to identify groups of proteins/genes that show common traits and significant correlation with a given phenotype. After its first application, several variants of GSEA have been proposed, resulting in a long list of successful studies and important results. Nowadays the term GSEA is commonly used to refer to the entire group of FCS algorithms, and sometimes extends to PT-based approaches as well [59]. In this thesis the term GSEA will be used to refer to the entire class of FCS algorithms.

GSEA algorithms require two types of input data: gene expression data and the gene sets to be evaluated. Expression data generally come from microarray experiments, and more recently from RNA-seq. Usually the expression levels of thousands of genes across different tissue samples/conditions are provided. Since one of the purposes of GSEA is to identify correlation between a gene set and a phenotype change, samples are supposed to be classified in different classes (i.e. control and condition, or disease subtypes). The gene sets to test have to be specified as well. There are several ways to define a group of gene sets. The most common one is to retrieve a set of pathways from a database, or to define groups of related genes sharing the same significant Gene Ontology (GO) terms.

The key steps performed by almost all the versions of GSEA are presented in Figure 3.1:

1. Data Preprocessing

Normalization, imputation of missing data, and probe mapping are three important data preprocessing steps.

Normalization allows expression values obtained from different experiments to be directly comparable. A number of methods are available for normalization [80, 81]. The most common normalization algorithms, RMA [16] and MAS 5.0 [18], are designed for expression levels generated with microarrays that follow a lognormal distribution.

The expression values of some genes may be missing in different microarray experiments due to technical issues. Imputation of missing data is thus important for maximal data coverage when the results of multiple experiments are compared. The performance of the imputation methods may vary drastically depending on the experimental settings and questions under study. Missing data can be imputed using methods based on K-Nearest Neighbors (KNN), Singular Value Decomposition (SVD), or Least Square (LS) regression models. Least Square regression algorithms and Bayesian Principal-Component Analysis (BPCA) were reported to produce lower estimation error than other methods [82, 83].

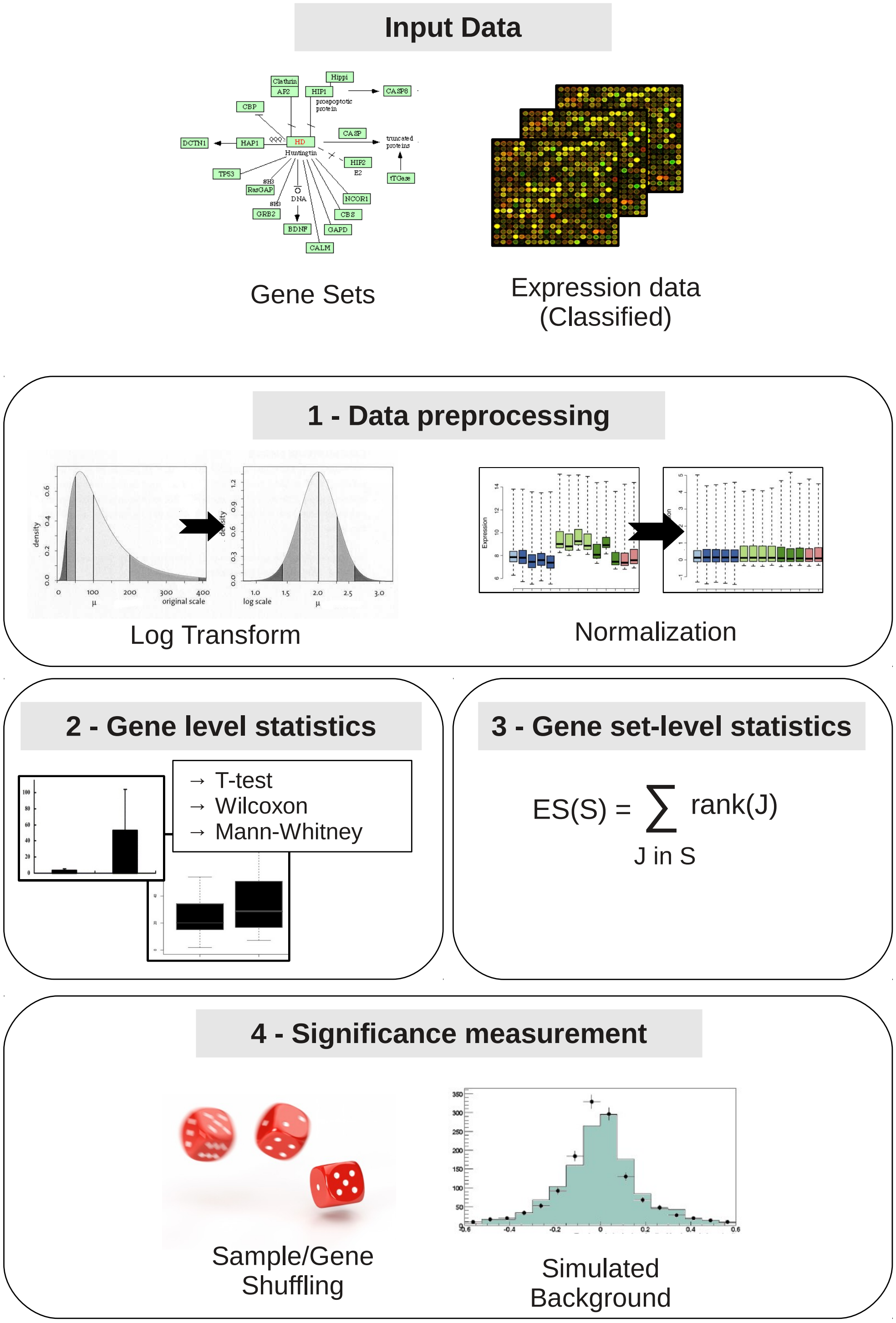


Figure 3.1: Typical workflow of GSEA algorithms.

As for the third preprocessing step, the many-to-many correspondences between genes and probe sets on a microarray creates ambiguity in determining expression levels of genes. The most common practices, generally based on the evaluation of the mean or median expression levels, directly merge the expression levels of the probe sets that correspond to the same gene. More recently new meta-analysis heuristics have been proposed [84]. These strategies evaluate probe set-level statistics by integrating the expression levels of single probes using methods such as Fisher's [85] or Stouffer's [86].

2-3. Gene- and Gene Set- level statistics

The second step in GSEA is to compute a gene-level statistic of differential expression, e.g. a t-statistic, a signal to noise ratio (mean to standard deviation ratio), a fold change or a Wilcoxon rank-sum statistic. Most of the proposed algorithms eliminate the direction of the differential expression, for instance by taking the absolute or square of their statistic [71, 87]. Intuitively, the statistic registers the association between gene expression and sample classification. In practice, the final product of this step is a list of genes, generally called Gene Expression Signature (GES), scored and ranked according to the selected statistic.

The gene-level statistics for all genes in a gene set are then aggregated into a single gene set-level statistic. Gene set-level statistics commonly used in current approaches include Kolmogorov-Smirnov statistic [79, 88], sum, mean, median [89], Wilcoxon rank-sum [90], and maxmean statistic [91]. Gene set-level statistics are alternatively referred to as Enrichment Scores (ESs).

4. Statistical significance evaluation

The purpose of a gene set-level statistic is to decide whether a gene set is distinct in some statistically significant way. In order to do that, it is necessary to specify a null hypothesis. Two popular null hypotheses have been defined by Tian et al. [92], and are respectively referred to as “self-contained” and “competitive” [93]. The former focuses on the single gene sets, and tests whether their association with a phenotype change are distinguishable from randomly shuffled phenotype changes. The latter verifies whether the gene set-level statistic of the gene set being tested differs from those of other (random) gene-sets [93]. The rationale for using this latter hypothesis is that a significant gene set should be distinguishable from an equal size set composed of randomly chosen genes.

Regardless the null hypotheses selected, a background distribution should be defined in order to test it. The background distribution can sometimes be written analytically, as in the case of a Gaussian distribution, and it can always be simulated by shuffling experimental data.

The procedure to simulate a background distribution is somehow shaped by the choice of the null hypothesis. To test a competitive hypothesis, the background distribution is usually obtained by shuffling genes; instead, for self-contained hypothesis, the background distribution is obtained by shuffling phenotypes. The latter procedure preserves the relationship of the genes in the set, and for this reason is generally favored [93]. Moreover, as already stated, it directly addresses the question of finding gene sets whose expression changes correlate with phenotype changes.

Under a self-contained hypothesis, the common procedure consists of shuffling the phenotype labels, calculate the differential expression of each gene, and compute a new statistic for the same gene set of the random dataset. The entire process is repeated multiple times to obtain a distribution of random gene set-levels statistics. Finally, a P-value of the real enrichment score can be evaluated as the fraction of random ESs at least as great as the real one. Although simulating the background distribution obviates the requirement of an analytical background, it can be computational expensive.

It is finally worth recalling that the P-value is the appropriate measure of statistical significance when only one gene set is tested. When a large number of gene sets are tested, there can be many false positives among the gene sets that receive seemingly highly significant P-values. This effect is generally called the multiple hypothesis testing problem. Several correction strategies have been proposed [68].

3.3 Master Regulator Analysis (MRA)

An important aspect of most pattern discovery methodologies is that they require as input data a group of related gene sets. Indeed, one of the problems purposely overlooked in the introduction is the definition of such input sets. Although the several available databases have been of some help in understanding cellular dynamics, the following drawbacks have been identified:

- not all the existing pathways are known

- a phenotypic condition might be the effect of small perturbations in several pathways; considering each pathway singularly might conceal crosstalking, and testing all the possible combinations of pathways would be unfeasible

- some multi-pathway perturbations may be disease-specific, and therefore not known a-priori

An alternative approach consists of determining and test custom gene sets. Within the pletora of possible strategies to define such sets, there is one that converges toward the identification of Master Regulators (MRs) of gene expression. Master regulators can be defined as molecules, such as Transcription Factors (TF) and miRNAs, driving and mantaining with their activity specific cellular phenotypes.
TFs are regulatory proteins that bind to the promoter regions of target genes (TGs) to regulate their levels of expression. MiRNAs, instead, are short non-coding RNAs that are incorporated into the RNA-induced silencing complex (RISC) to regulate the stability and translation of messenger RNA (mRNA) transcripts [94]. The activity of such regulators is often not visible at the mRNA level: TFs are frequently modulated at the post-transcriptional level [95], and miRNAs are usually not profiled. Thus, expression at the mRNA level is often a poor predictor of a regulator activity, and an even worse predictor of its biological relevance in regulating phenotype-specific programs. An appealing solution consists in consider the sets of downstream targets of master regulators, commonly referred to as regulons, and explore their correlation with a given phenotype exploiting GSEA techniques [78]. This type of analysis is commonly referred to as Master Regulator Analysis (MRA).

MRA proved to be invaluable for system level studies of cellular conditions. Indeed, MRs do not operate in isolated processes, but are interconnected in a network of common targets and mutual interactions and interferences. Therefore, they are able to affect several pathways, regulating some of their components from the transcriptional to the post-translational level. In this context, the regulons of MRs can be extracted from repertoires of molecular interactions, referred to as interactomes, because they provide an integrated view of regulatory programs in the cell [96].

Most available interactomes still lack context-specificity because their interactions are supported by ex vivo assays or literature data assembled from a diverse mix of cellular phenotypes. However, the real regulatory networks in multicellular organisms are cell-context specific. Different algorithms have been proposed to infer disease and tissue specific interaction networks. For instance, ARACNe [97] is an algorithm for the dissection of transcriptional networks that can infer the targets of transcription factors from microarray expression profiles. Employing ARACNe and Mindy [98], a human B-cell interactome has been constructed by reverse-engineering the interactions at the transcriptional and post-translational level in mature human B-cell [99]. Exploiting context-specific interaction networks allows to investigate the regulatory activity of specific cellular phenotypes. Some MRA works that exploit context-specific networks are described in the following, together with the corresponding algorithms.

3.3.1 MRA via Fisher Exact Test (MRA-FET)

Master Regulator Analysis via Fisher's Exact Test (MRA-FET) [100] is an algorithm used to identify transcription factors whose targets are enriched for a particular gene signature. It belongs to the first category of pathway discovery approaches (ORA). Given a group of regulons, and a set of "significant" genes P, MRA computes the statistical significance of the overlap between each regulon and P. The P-values are computed by Fisher's exact test (FET).

MRA-FET has been applied to breast cancer data [101]. Briefly, a disease-specific transcriptional network is reconstructed from a set of gene expression profiles in tumoral samples using ARACNe. Then, the significance of the overlap between each regulon and different sets of signature genes [102, 103] is evaluated.

High-grade gliomas (HGGs) are the most common brain tumours in humans. In [104] a "mesenchymal" gene expression signature has been identified. The overexpression of the "mesenchymal" gene expression signature characterizes tumor aggressiveness in human glioma. Unfortunately, the regulatory program that leads to the drift towards this mesenchymal signature is not clear, and the molecular events that activate the mesenchymal signature remain unknown. In [100], the application of the MRA-FET algorithm to high-grade gliomas led to the identification of a transcriptional module that activates the expression of the mesenchymal genes. In particular, two transcription factors, $C/EBP\beta$ and $STAT3$, have been identified as synergistic initiators and master regulators of mesenchymal transformations.

3.3.2 MAster Regulator INference algorithm (MARINa)

MRA-FET requires as input data a list of “interesting” genes (i.e. those differentially expressed). In other words, MRA-FET works on the basis of a binary classification of genes in differentially and not differentially expressed. In many cases this type of data is not available, or it requires a lot of efforts to be built. This approach has two drawbacks. First, a criterion to establish whether a gene is differentially expressed is required. Choosing the right criterion or threshold is not easy, and can influence the downstream analysis. More importantly, gene with small perturbations are not taken into account.

Following the evolution from ORA- to FCS- approaches, a new algorithm has recently been proposed. The MAster Regulator INference algorithm (MARINa) is a GSEA pipeline that determines whether the regulon of a TF is enriched for genes differentially expressed between two classes of samples [99]. Being a FCS algorithm, MARINa does not require as input data a list of differentially expressed genes. All the genes are considered, and ranked according to their association with a phenotype, generally determined using the T-test. The benefits of such strategy are multiple: first of all, there is no need to establish a criterion to decide whether a gene is differentially expressed or not. Second, genes with small variations are taken into account, following the philosophy that several small contributions combined together can be significant.

MARINa requires as input data the expression profiles of genes across different samples, the regulons for each potential master regulator, and a classification of the samples. The algorithm resembles the pipeline described in Figure 3.1. Given two phenotypes, for each gene a gene-level statistics that describes its differential expression is calculated. Then, for each master regulator, a summary score is evaluated, by combining the gene-level statistics of its targets (group-level statistics). Finally, the significances of group-level scores are evaluated by comparing them to random cases, selected by permutations of sample labels.

MARINa has been successfully used in the study of malignant mature human B-cells [99]. The goal was to discover master regulators of key genetic programs in the germinal center (GC) reaction of antigen-mediated immune response. In other words, the problem was to identify the genes required for normal progression through the GC. The regulons for each TF have been collected from a cell-context specific human B-cell interactome (HBCI). HBCI has been built by reverse-engineering transcriptional and post-translational interactions in mature human B-cells, using ARACNe [105] and MINDy [106] algorithms. Interestingly, not only transcriptional interactions were considered, but also PPIs, representing direct and physical interactions, and direct protein-DNA interactions. The GES was obtained by t-test analysis of GC centroblasts versus naive B-cells samples. When applied to 194 TFs displaying ≥ 20 targets in the HBCI, MARINa identified 41 candidate MRs, of which 26 were GC activated and 15 were GC repressed.

3.4 Master Regulator Analysis of miRNAs

The analysis of master regulators can be extended from the transcriptional to the post-transcriptional level, by including, for instance, the effect of miRNAs on gene translation. The biological question is the same: determine which molecules act as master regulators and are likely to drive the cell toward a specific phenotype. It has been shown that miRNAs play an important role in the regulation of the cellular machinery. For instance, it has been demonstrated that the dysregulation of miRNAs is related to tumor initiation and progression [107].

In this thesis, an adapted version of MARINa has been used to identify miRNAs acting as MRs between different subtypes of Glioblastoma Multiforme (GBM). Beyond the methodological contributions, this study proposes a list of miRNAs potentially acting as MRs. In the following, the description of the computational aspects is intertwined to the presentation of the GBM case study, to better highlight the contributions of the thesis.

3.4.1 A case study: Glioblastoma Multiforme (GBM)

Glioblastoma Multiforme (GBM) is one of the most common and malignant forms of brain tumors. Patients affected by GBM generally have a poor prognosis, with a survival of 12 months, on average [108]. Several studies tried to uncover the molecular causes of its development, considering different aspects. For instance, in [108] a gene expression-based molecular classification of GBM samples in Proneural, Neural, Classical and Mesenchymal subtypes has been proposed. Another classification is based on the state of methylation of some CpG islands [109]. According to it, the Proneural subtype can be further separated into GCMP+ and GCMP- subclasses. Interestingly, GCMP+ samples share specific molecular features (particular DNA methylation alterations, and distinct copy-number alterations), and patients show significantly higher survival rates.

The effects of dysregulation of miRNAs in GBM have been subject of several studies. For instance, in [110] the combined analysis of gene expression and microRNA profiles uncovered a post-transcriptional network of 248k miRNAmediated interactions. Biochemical analyses confirmed the presence of miRNA modulators in GBM that, acting as “sponges”, regulate the action of miRNAs. In [111], instead, in vitro and in vivo studies determined that hsa-miR-26a can promote glioblastoma cell growth, enhancing its proliferation and decreasing apoptosis. So far, however, MRA has not been applied to identify miRNAs acting as master regulators.

In this thesis an attempt has been made to identify miRNAs responsible for mantaining a specific GBM phenotype. The computational methodology has some similarities to the original MRA of TFs on malignant human B-cell [99]. Indeed, initially the same version of MARINa has been used as GSEA tool, but it was not suited to the input data and the miRNA’s mode of action. Thus, incremental efforts have been made to adapt the analysis pipeline to the MRA of miRNAs, leading in the end to sound results.

3.4.2 Input data

As for the MRA of TFs, in this analysis a biological network has been exploited to build the regulons of each miRNA. Other required input data are the expression profiles of genes across the different samples and a classification of samples in the different subclasses.

Expression profiles and samples classification

The Cancer Genome Atlas (TCGA) [112] is an initiative, launched in 2006, with the purpose of generating comprehensive, multi-dimensional maps of the key genomic changes in major types and subtypes of cancer. It provides a huge collec-

Subtype	Number of samples
Classical	97
Mesenchymal	97
Neural	64
Proneural GCMP-	67
Proneural GCMP+	20
Total	345

Table 3.1: Classification of GBM samples from TCGA. Note that 146 out of the 491 samples available in TCGA were not classified.

tion of data freely available, that enables researchers anywhere around the world to make and validate their discoveries.

GBM is one of the cancer types currently included in TCGA. On November 2011, about 500 samples of Glioblastoma Multiforme tumors had already been profiled, with microarray and miRNA profiles available. Level 3 data have been considered: probe ids were already associated to the corresponding genes, and expression levels were already log-transformed and normalized. To improve the quality of the analysis, the gene names were mapped to Entrez ID, and checked for inconsistencies. In the end, the profiles of 528 miRNAs and 17286 genes were available for 491 tumoral samples.

Samples were divided in Proneural, Neural, Classical and Mesenchymal classes, following the classification proposed in [108]. Proneural samples were further divided in GCMP+ and GCMP-, as proposed in [109]. Statistics for the final classification are reported in Table 3.4.2.

GBM interactome - miRNA-mRNA network

MiRNA regulons have been extracted from a regulatory network combining the transcriptional-level regulation of TFs to the post-transcriptional action of miR-

NAs. TF-target interactions were inferred by running ARACNe on GBM expression data, as done for the human B-cells. MiRNA targets, instead, have been predicted with Cupid [110], a miR-target prediction algorithm that scores miR-binding sites by integrating (a) predicted site scores from TargetScan [113], PITA [114] and miRanda [115], (b) 46-vertebrate genome cross-species conservation scores by PhasCons [116], and (c) positional information relative to the 3' UTR start site. Cupid was used to build a miRNA-mRNA network of 154341 potential interactions between 462 miRNAs and 7376 target genes. Gene ids have been rigorously mapped to Entrez IDs, and miRNA ids have been disambiguated using the 3p/5p notation. It is worth noting that some miRNAs are intragenic, and are therefore transcribed together with the corresponding host genes. If the host gene of an intragenic miRNA is regulated by a TF, then the miRNA itself is regulated by the same TF. The final regulatory network is a directed (cyclic) graph, with edges connecting miRNAs and TFs to their targets.

3.4.3 First MRA pipeline: one-tail MARINa

MARINa is formalized in Algorithm 1. Required input data are the expression profiles E of the genes in the various samples S, a classification C dividing S in two classes, and the regulons R to be evaluated. Referring to the line numbers reported in the pseudo-code, in line 2 the GES between the compared phenotypes is calculated. Wlech T-test [117] has been used as statistic of the differential expression of genes between the two classes of C. The GES consists of a ranking of the genes, according to their differential expression statistics. The GES is used in line 8 to evaluate a score, equivalent to the Enrichment Score (ES) defined during the description of the GSEA, for each regulon. Similarly to what proposed in one of the first works of GSEA [79], the Kolmogorov-Smirnov statistic is used to merge single gene scores. Afterwards, the significance of each miRNA's activity is estimated by comparing its ES to an empirical background distribution. As anticipated, when generating random data, it is possible to shuffle either gene or sample labels. MARINa can be classified as a self-contained GSEA algorithm, and therefore performs a permutation of sample labels (equivalent to randomly divide samples S in two classes). A background distribution can be built by collecting the GESs of a sufficiently elevated number of random cases (lines $4-5$), and evaluating the group level statistics (random ESs) of the regulons on each random GES (line 10). The fraction of random ESs higher than the ES on the real data can be interpreted as an empirical pvalue. The self-contained null hypothesis can be rejected for a given regulon if the resulting pvalue is below a user-defined threshold.

MARINa has been integrated in a first version of the analysis pipeline, outlined in Algorithm 2. In the design of this first tentative pipeline particular attention has been paid to two aspects related to the definition of miRNA regulons (line 2): the range of influence of master regulators, and their "mode of regulation".

Range of influence of MRs As anticipated, regulons used as input sets of GSEA are extracted from the regulatory network. One of the aspects better emphasized by biological networks is that the effects of a molecule can propagate through the network and influence distant targets. Thus, given a potential, all its downstream targets in the regulatory networks might be included in its regulon. An important consideration regards the magnitude of the regulation that a MR exerts

Algorithm 1: MARINa algorithm

```
Data: $R$: regulons
      $E$: Expression values $E_{i,j}$ of gene $i \in G$ in sample $j \in S$
      $C_{real}$: classification of samples $j \in S$ in two phenotypes
Result: $P$: Pvalue for each regulon $\in R$
begin
    $S \leftarrow GES(E, C_{real})$ ; // array of $|G|$ statistics measuring the
    correlation between each gene and $C_{real}$
    for $l = 1$ to custom number of random samples do
        $C_l \leftarrow$ random classification of samples $S$ in two classes ; // Generate a
        random permutation of sample labels
        $S_l \leftarrow GES(E, C_l)$ ; // Generate the GES for the random
        classification $C_l$
    end
    forall the $R_m$ do
        $E_k \leftarrow GroupLevelStatistics(R_m, S)$ ; // Generate a group-level
        statistics for $R_m$ starting from the real GES $S$
        for l=1 to custom number of random samples do
            $E_{k,l} \leftarrow GroupLevelStatistics(R_m, S_l)$ ; // Generate a
            group-level statistics for $R_m$ starting from the
            GES $S_l$
        end
        Build a null distribution from the random group-level statistics $E_{k,l}$
        $P \leftarrow$ significance of $R_m$ comparing $E_k$ to the null distribution
    end
end
```

on its targets. Some targets are mildly influenced, while other are strongly regulated. Contrary to the intuition, direct targets of a MR (its downstream neighbors) are not always more affected than farther ones. For instance, a master regulators might regulate multiple downstream TFs that together regulate a common target (farther from the MR regulator). This fact, however, is not easy to capture, at least within current networks, and the influence of a MR on close targets is therefore easier to study. Indeed, farther targets can be potentially influenced by several different regulators, and the association between their expression and regulator's activity might not be evident. This is particularly valid in current studies, where fewer samples than features are commonly available. Additionally, transcriptional networks generally used in MRA studies are inferred from expression data, and are characterized by significant levels of false positive interactions. If downstream nodes of a false direct target of a MR are included in the regulon, the effects of the first wrong interaction on the GSEA would be amplified. Moreover, inferred networks usually feature several interactions, and even considering targets at distance 2 generates huge regulons (containing over 1000 genes). From a computational point of view this is problematic, since GSEA results are not reliable on such big sets.

Only direct targets of each TF have been included in the regulons considered in the MRA of malignant human B-cells [99]. After some preliminary tests, also in this work the direct targets of each miRNA have been used to build the regulons. Resulting regulons are still quite big, with some counting more than 800 genes. Out of 462 miRNAs, only the 430 ones with more than $\geq$ 30 targets have been considered.

Mode of regulation of MRs Generally speaking, a molecule can have either an enhancing or repressive regulatory effect on a target. The magnitude of such effect is exquisitely context-specific, and is bound, among other things, to complex post-transcriptional and post-transcriptional interactions. This fact, henceforth referred to as "mode of regulation", must be taken into account in the MRA, and directly influences the calculation of regulon ESs (line 8 in Algorithm 1). Briefly, genes within a TF regulon are divided in two subsets: targets being repressed, and targets being enhanced by the TF. For a TF to be differentially active, repressed and enhanced targets are supposed to be enriched on the opposite tails of the GES. To take this fact in consideration, in the analysis of human B-cells a two-tail version of GSEA was used, as described in [101]. Ignoring this aspect would significantly descrease the accuracy of MRA. MiRNAs, however, usually act as repressors of mRNA translation, by interacting with the RISC complex [94]. Even though it has been shown that some miRNAs enhance the mRNA translation, the percentage of these instances is low. Thus, in this work miRNAs are assumed to always repress their targets. According to this model, to be differentially active a miRNA regulon should be enriched on a single side of the GES. A one-tail version of the Kolmogorov-Smirnov statistic has been used in MARINa, instead of the two-tail version, to evaluate the ESs. Further detail are available in [79].

Preliminary results At first, Proneural GCMP+ and Proneural GCMP- phenotypes were compared. The uncorrected p-values of all the miRNAs are ranked in Figure 3.2. Only 2 miRNAs obtained a p-value ≤ 0.01. This is a rather negative result, since a multiple hypothesis correction would raise all the p-values above 0.1. More importantly, there is no correlation between the expression of a miRNA

Algorithm 2: MRA (first version)

```
Data: $G$: regulatory network
      $M$: list of potential master regulators
      $E$: expression values $E_{i,j}$ of gene $i \in G$ in sample $j \in S$
      $E_{mir}$: expression values $E_{m,j}$ of miRNA $m \in G$ in sample $j \in S$
      $C_{real}$: classification of samples $j \in S$ in two phenotypes
Result: $P$: Pvalues for each regulator $m$
begin
    $R = \emptyset$
    forall the $m \in M$ do
        $R_m \leftarrow$ regulon of $m$ (built exploiting $G$)
        $R \leftarrow R \bigcup \{R_m\}$
    end
    $P \leftarrow$ MARINa($R$,$E$,$C_{real}$) ; // See Algorithm 1
end
```

and its activity, as outlined in Figure 3.3. The scatter plot shows that the activity of the regulons is almost independent from miRNAs' differential expression. Even though a perfect correlation is not expected, a moderate concordance between the expression level of a miRNA and the global expression level of its targets should be present.

3.4.4 Second MRA pipeline: contex-specific miRNA-mRNA networks

A possible explanation for the negative results obtained by the direct application of MARINa is that regulons are inaccurate. It is indeed possible that some predicted targets do not respond to the miRNA in the context of GBM. Indeed, there are several factors that might modulate the action of miRNAs on targets. For instance, there are evidences of modulators that act as "sponges", and interfere with the effects of miRNAs [110]. In other words, the miRNA-mRNA interaction network used to define the regulons is not disease specific, with regulons including genes

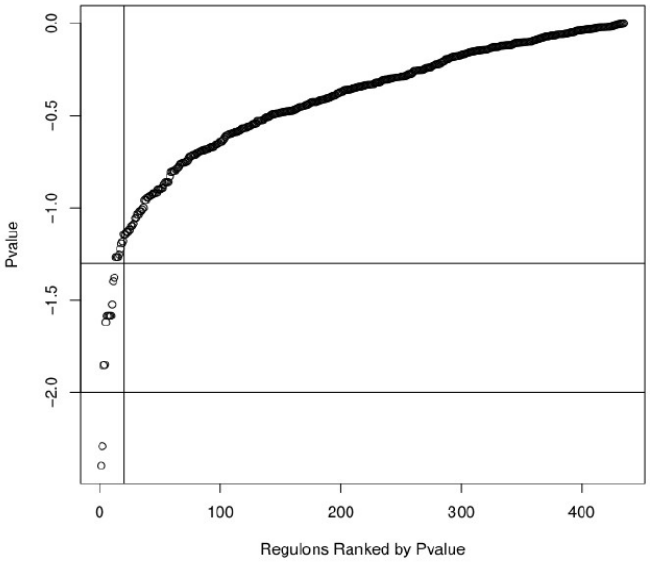


Figure 3.2: Uncorrected p-values of the differential activity of miRNAs between GCMP+ and GCMP- subtypes. Results are obtained with the first MRA pipeline, using the original regulons inferred from the regulatory network. Only two miRNAs have p-values ≤ 0.01.

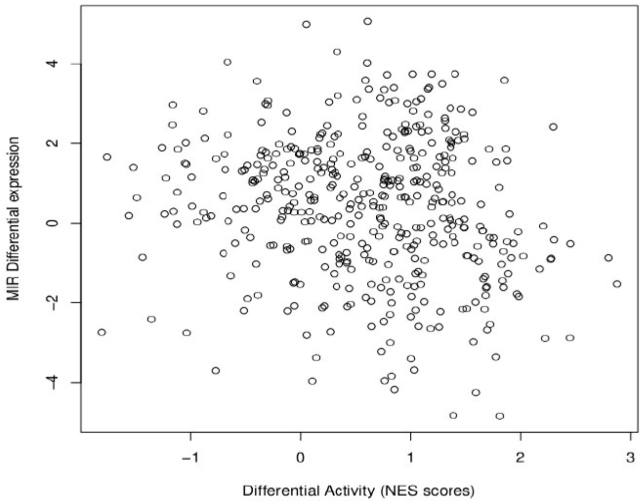


Figure 3.3: Scatter plot of the Differential Activity vs Differential Expression of miRNAs, between GCMP+ and GCMP- subtypes. Analysis performed with the first MRA pipeline, using the original regulons inferred from the regulatory network. There is no correlation between miRNA expression and activity (Pearson's correlation: -0.15).

that are not affected by the corresponding miRNA in GBM.

To solve this problem, regulons can be pruned by removing those targets that are not likely to respond to the regulator, at least in the context of the disease. Under the hypothesis that miRNAs exercit only a repressive effect, the pruning consists of verifying whether a potential target's expression is lower when the regulator is more expressed, and vice versa. The corresponding miRMA pipeline is outlined in Algorithm 3, where an additional pruning step has been added (line 5). The pruning strategy devised in this work is reported in Algorithm 4. Given the original regulon R_m relative to miRNA m, all the samples $j \in S$ are divided in two groups P and N. Let $\bar{m}$ be the mean of the expression levels of m across all the samples. A sample j is assigned to group P if the expression level of m in j is $\geq \bar{m}$, and to N otherwise (lines $2-4$). If a target $i \in R_m$ is effectively repressed my m, then its expression levels in samples $\in P$ should tend to be lower than in samples $\in N$. This can be verified, for instance, by evaluating the difference of the mean expression of i between P and N. If the difference is not negative, i is removed from the regulon (lines $7-9$).

On pruned regulons MARINa produced promising results. In particular, as shown in Figure 3.4, there is a significant correlation between the expression of a miRNA and its regulon activity. More detailed results are presented in the next section (Results). In the following, some additional considerations about the pruning are presented.

A critical aspect is avoid overfitting in the pruning step. Indeed, if only the samples of the compared subtypes are considered, almost all the miRNAs result differentially active. GBM samples are classified in more than two subgroups. This means that there are always some samples that do not belong to the com-

Algorithm 3: MRA (final version)

```
Data: $G$: regulatory network
      $M$: list of potential master regulators
      $E$: expression values $E_{i,j}$ of gene $i \in G$ in sample $j \in S$
      $E_{mir}$: expression values $E_{m,j}$ of miRNA $m \in G$ in sample $j \in S$
      $C_{real}$: classification of samples $j \in S$ in two phenotypes
Result: $P$: Pvalues for each regulator $m$
begin
    $R = \emptyset$
    forall the $m \in M$ do
        $R_m \leftarrow$ regulon of $m$ (built exploiting $G$)
        $R \leftarrow R \bigcup \{$PruneRegulon($R_m$, $E$, $E_{mir}$)$\}$ ; // See Algorithm 4
    end
    $P \leftarrow$ MARINa($R$,$E$,$C_{real}$) ; // See Algorithm 1
end
```

Algorithm 4: PruneRegulon

```
Data: $R_m$: regulon of miRNA $m$
      $E$: expression values $E_{i,j}$ of gene $i \in G$ in sample $j \in S$
      $E^{mir}$: expression values $E_j^{mir}$ of miRNA $m \in G$ in sample $j \in S$
Result: $PR_m$: Pruned regulon
begin
    $thresh \leftarrow$ mean($E_{m,j}$) ; // mean of miRNA expression across all
    the samples
    $P \leftarrow \{$ samples $j \in S$ $|E_j^{mir} \geq thresh$ $\}$
    $N \leftarrow \{$ samples $j \in S$ $|E_j^{mir} < thresh$ $\}$
    $PR_m = \emptyset$
    forall the $i \in R_m$ do
        score $\leftarrow$ T-test($E_{i,P}$, $E_{i,N}$) ; // $E_{i,P}$ is the set of expression
        values of $i$ across samples $\in P$. $E_{i,N}$ is the set of
        expression values of $i$ across samples $\in N$
        if $score < 0$ then
            $PR_m \leftarrow PR_m \bigcup \{i\}$ ; // Accept target $i$
        end
    end
end
```

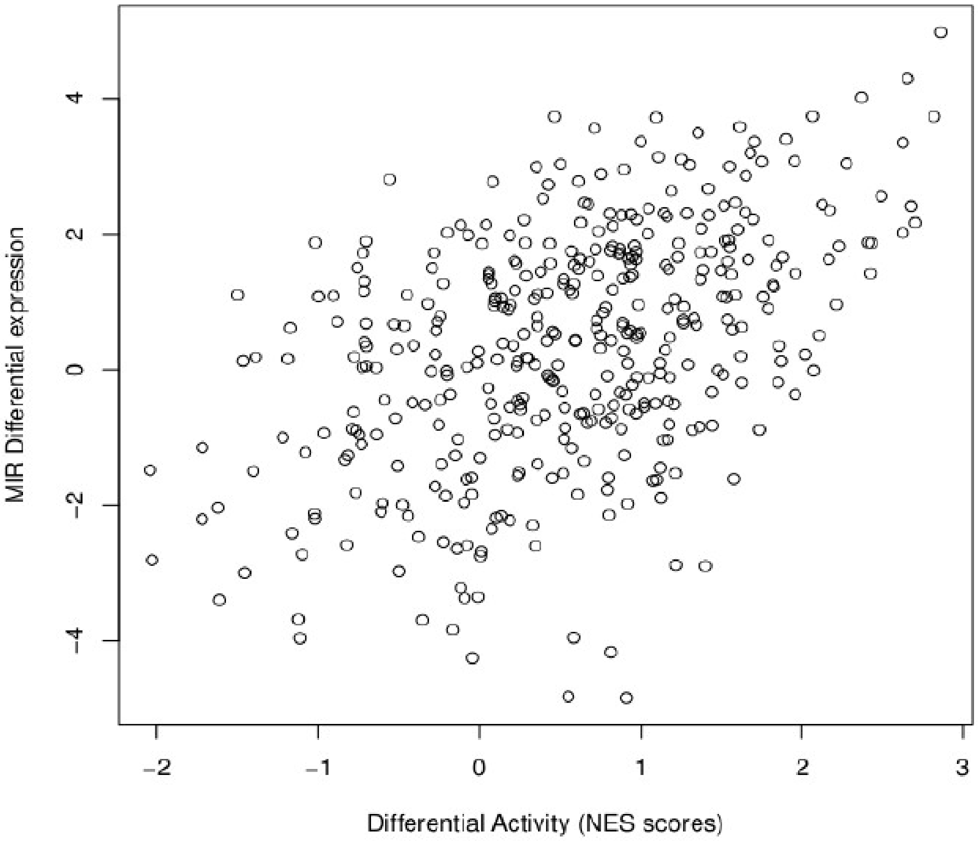


Figure 3.4: Scatter plot of the Differential Activity vs Differential Expression of miRNAs, between GCMP+ and GCMP- subtypes. Analysis performed with the second MRA pipeline, using the pruned regulons. miRNAs' expression and activity show an observable level of correlation (Pearson's correlation: 0.468, p-value < 0.001).

pared classes, and can be exploited to prune the regulons avoiding overfitting. In this work a single pruned interactome has been obtained, by considering all the samples in Algorithm 4. An alternative consists in removing, from the set of samples S used in the pruning step, those that belong to the phenotypes to be compared. Even though this strategy avoids any overfitting, it generates different regulons for each pair of compared phenotypes (for the case study, there would be more than 10 interactomes). In this scenario, genes that are considered as targets of a miRNA in a comparison, are not in others. This fact is not easy to justify. Moreover, when removing some classes, such as the Mesenchymal, a lot of samples are discarded, and the number of remaining samples is significantly lower than in the case of comparisons not involving the Mesenchymal phenotype. Less samples means lower observation power, and as a consequence, pruning quality. It was verified that best results were indeed obtained by using the same regulatory network in all the comparisons, built considering all the samples.

It might also be objected that the measure used to compare the expression levels of the targets is naive. More refined strategies, such as a statistics based on the Wilcoxon Rank-Sum Test, could be used instead. The simple T-test is nevertheless appropriate for verifying whether a gene shows any degree of inverse correlation respect to a miRNA.

3.5 Results

The final pipeline has been applied to compare the GBM subtypes. A positive and significant correlation between the expression and the activity of miRNAs was observed in all the comparisons (Figures 3.5 and 3.6). The biggest differences were observed between Mesenchymal and GCMP+ subtype: not only several miRNAs are differentially expressed between the two classes, but many are also differentially active. As expected, instead, weaker differences were registered between GCMP+ and GCMP-, both subclasses of Proneural subtype.

Patients affected by GCMP+ subtype have a significantly higher survival rate than the others [109]. Thus, it might be interesting to identify the miRNAs differentially active between this and the other classes of GBM. In Figure 3.7, each miRNA is ranked by its MRA p-value. In line with the general comparison, few miRNAs are differentially active between GCMP+ and GCMP-. More and more miRNAs become differentially active as the comparison moves toward the Mesenchymal phenotype. A list of the top differentially active miRNAs in each comparison is presented in Table 3.5. Some miRNAs are differentially active in more than one comparison. In general, both common and specific miRNAs are significant. A combined overview of top differentially active miRNAs is proposed in Figure 3.8. Biggest differences arise from the comparison of Mesenchymal and GCMP+ subtypes, with many miRNAs differentially active. Only miR-101, instead, is differentially active in the comparison between GCMP+ and Proneural GCMP-.

Literature mining confirms the implication of miRNAs in GBM. A detailed comparison of activity and expression of a small selection of miRNAs is shown in Figure 3.9. In [118], hsa-miR-181a/b were reported to be down-regulated in all grades of glioma. In particular, the expression level of hsa-miR-181a was negatively correlated with tumor grade. This is in agreement with the results of the MRA. Indeed, the expressions of mir-181a/b/c/d are higher in GCMP+ respect to all the other subclasses, and their regulons are differentially active (see Figure

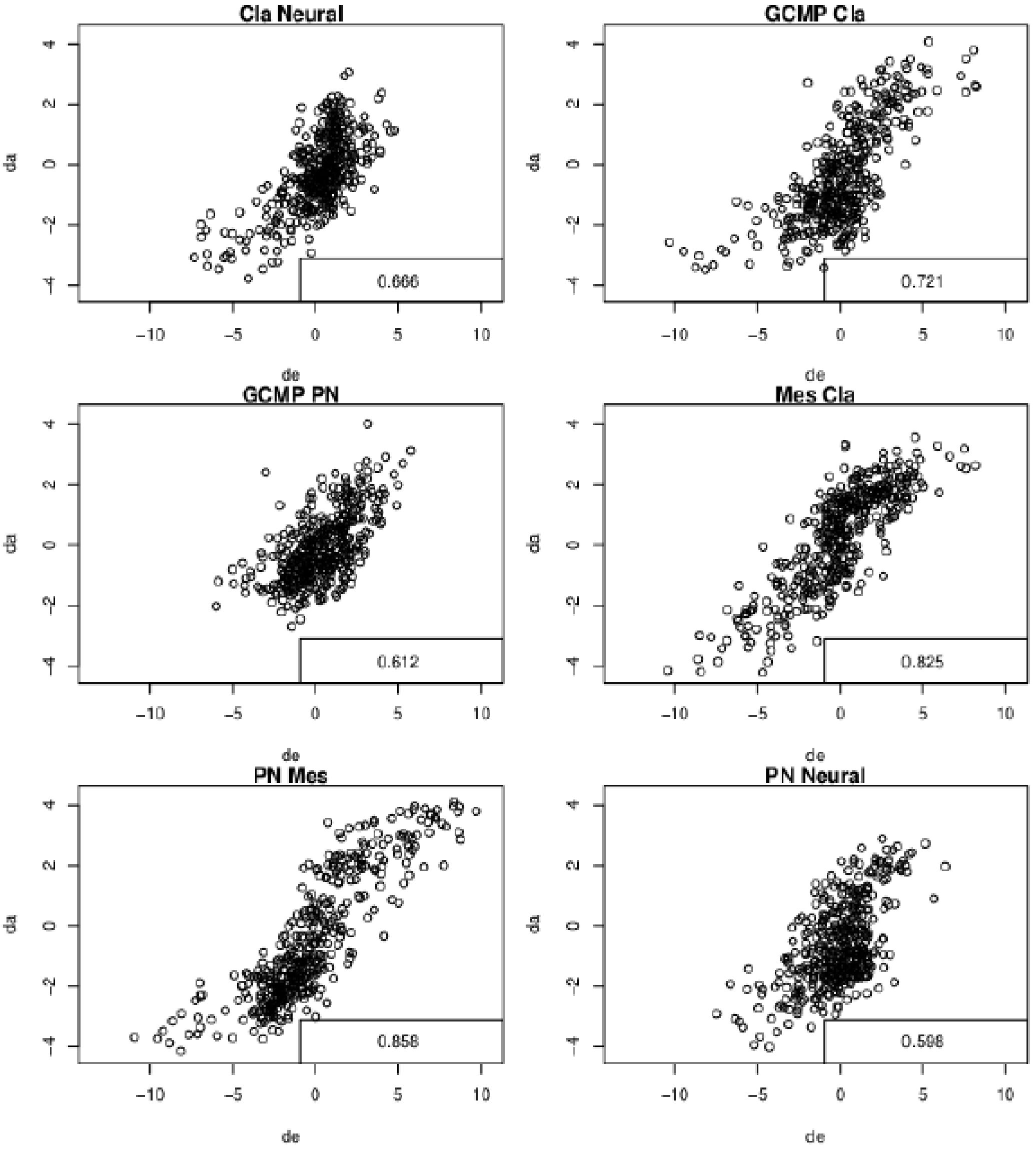


Figure 3.5: Scatter plots of the Differential Activity vs Differential Expression of miRNAs. Each plot represents a different comparison (**Mes**:Mesenchymal, **Cla**:Classical, **PN**:Proneural GCMP-). The analysis has been performed with the second MRA pipeline, using the pruned regulons. The Pearson's correlations between activity and expression of miRNAs are reported in the lower right corners of each plot. A significant correlation between miRNAs' expression and activity is observable. Continues in Figure 3.6.

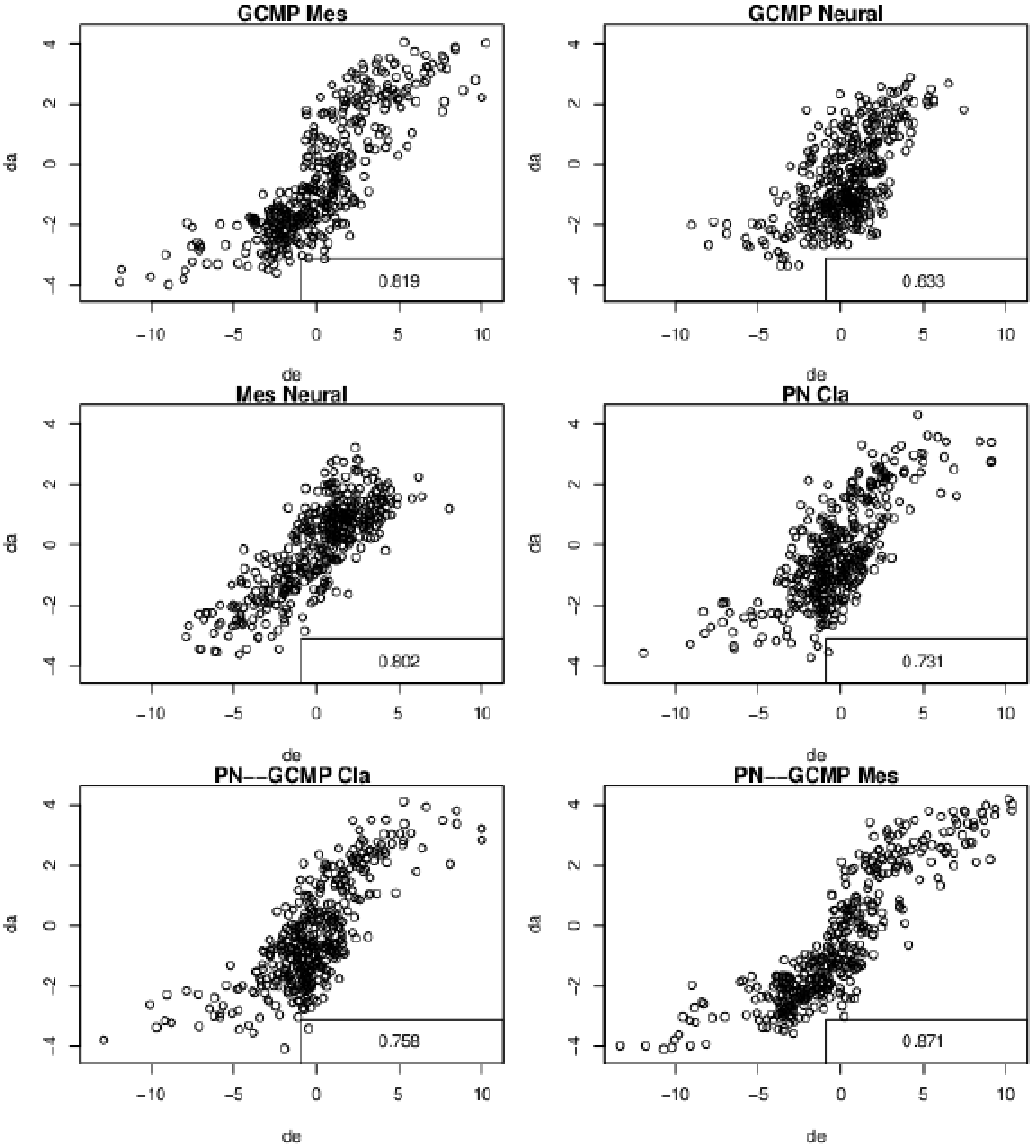


Figure 3.6: Continues from Figure 3.5.

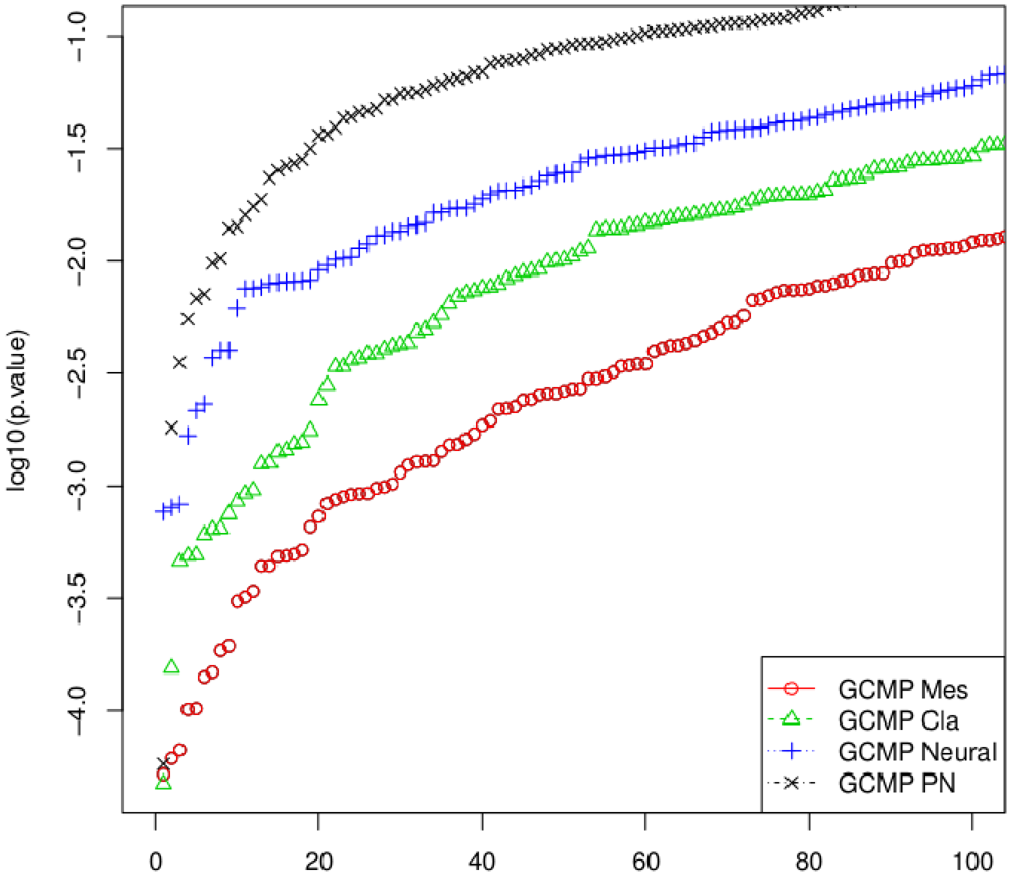


Figure 3.7: Uncorrected p-values of the differential activity of miRNAs in different comparisons. Results obtained with the second MRA pipeline, using the pruned regulons. GCMP+ subtype has been compared to all the other ones (**Mes**:Mesenchymal, **Cla**:Classical, **PN**:Proneural GCMP-). As expected, the Mesenchymal phenotype is the farthest from GCMP+, and little differences are instead observable in the comparison with GCMP-.

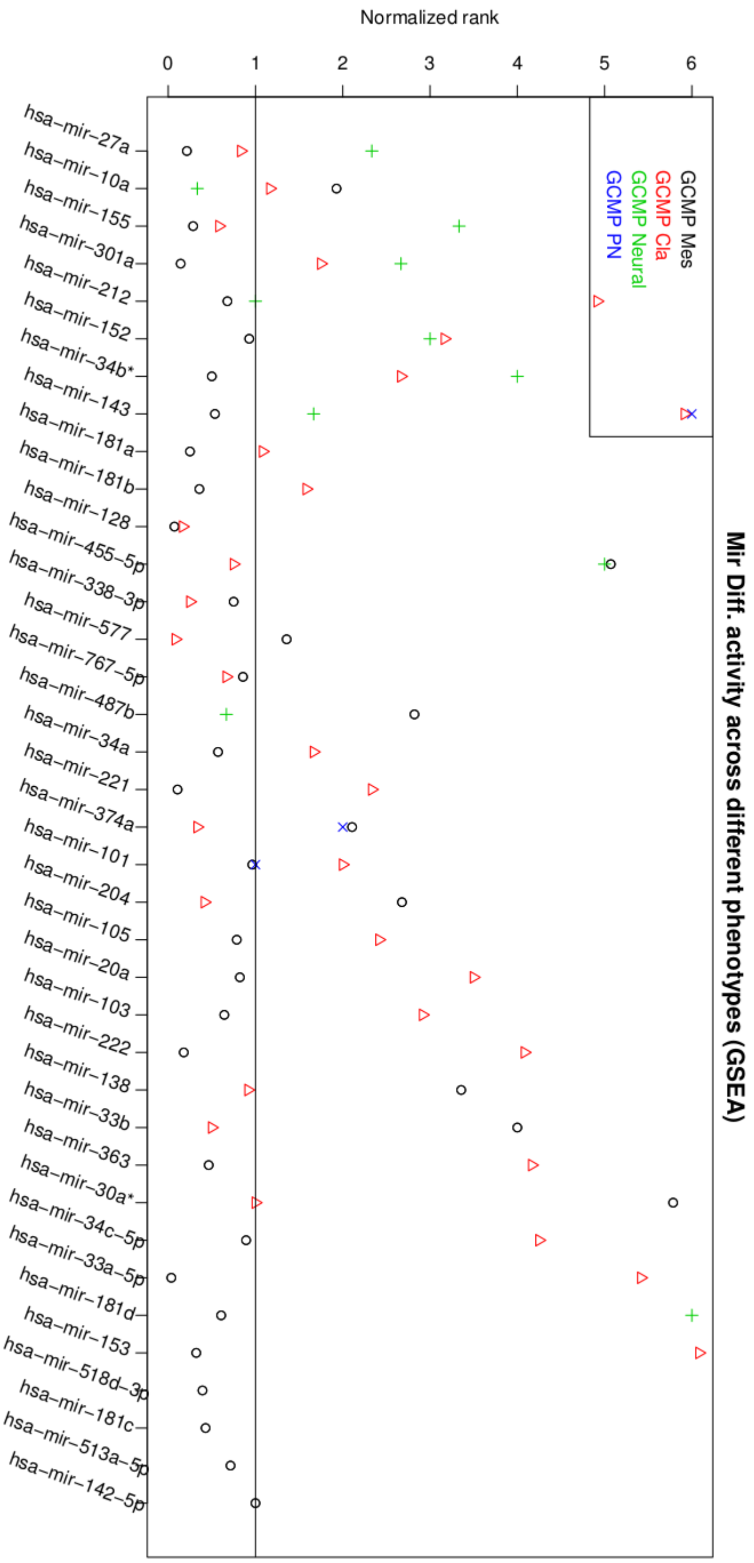


Figure 3.8: Combined plot of the most differentially active miRNAs (p-value $\leq 10^{-4}$) in the comparisons of GCMP+ with the other 4 subtypes (**Mes**:Mesenchymal, **Cla**:Classical, **PN**:Proneural GCMP-).

GCMP Neural		GCMP PN		GCMP Mes		GCMP Cla	
mir	p.value	mir	p.value	mir	p.value	mir	p.value
10a	7.67e-04	101	5.79e-05	33a	5.22e-05	577	4.74e-05
487b	8.00e-04	374a	1.81e-03	**128**	6.14e-05	**128**	1.55e-04
212	8.27e-04	30e	3.53e-03	221	6.69e-05	338	4.59e-04
31	1.67e-03	641	5.51e-03	301a	1.01e-04	374a	4.89e-04
143	2.16e-03	30e*	6.80e-03	222	1.02e-04	204	4.93e-04
29a	2.31e-03	143	7.12e-03	27a	1.41e-04	33b	6.07e-04
27a	3.70e-03	**181a**	9.78e-03	**181a**	1.49e-04	**155**	6.41e-04
301a	4.00e-03			**155**	1.86e-04	767	6.46e-04
152	4.01e-03			153	1.95e-04	455	7.48e-04
155	6.16e-03			181b	3.07e-04	27a	8.54e-04
30a	7.54e-03			518d	3.20e-04	138	9.25e-04
34b*	7.55e-03			181c	3.40e-04	30a*	9.58e-04

Table 3.2: Lists of top differentially Active miRNAs in the different comparisons of GCMP+ (**Mes**:Mesenchymal, **Cla**:Classical, **PN**:Proneural GCMP-). MiRNAs in bold appear in more than one comparison.

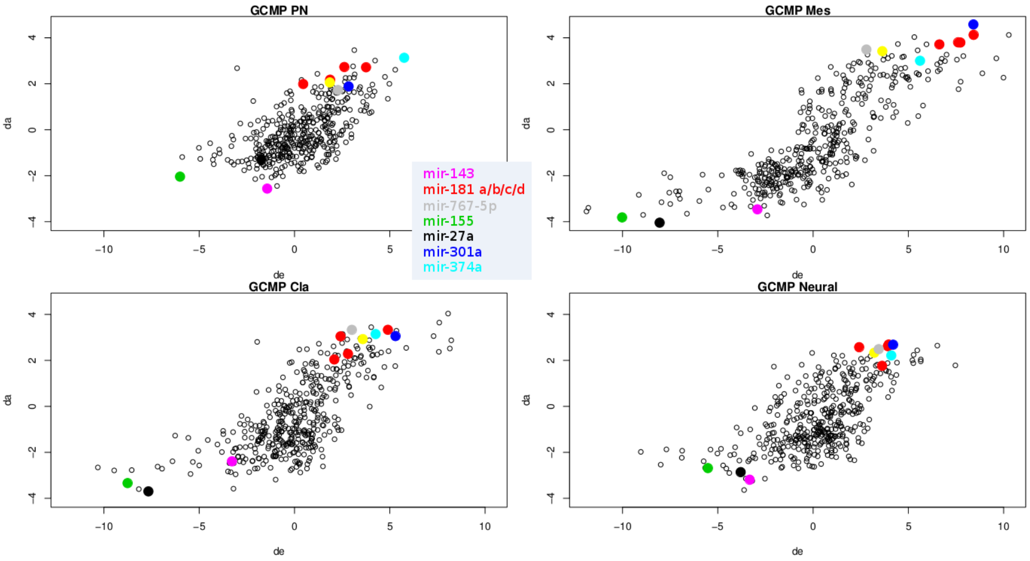


Figure 3.9: Combined view of selected miRNAs in the four comparisons of GCMP+ against the other phenotypes (**Mes**:Mesenchymal, **Cla**:Classical, **PN**:Proneural GCMP-).

3.9).

It has been reported that hsa-miR-155 is significantly elevated in GBM, and in a recent study [119] human GBM cells have been treated with a miR-155 inhibitor. Results showed a significant increase in growth inibition. Indeed, miR-155 regulates multiple genes associated with cancer cell proliferation, appoptosis, and invasiveness. Interestingly, miR-155 is significantly down-regulated in GCMP+, and results significantly active in the comparison of GCMP+ with Mesenchymal and Classical subtypes (as shown in the second and third plots of Figure 3.9).

3.6 Conclusions and future directions

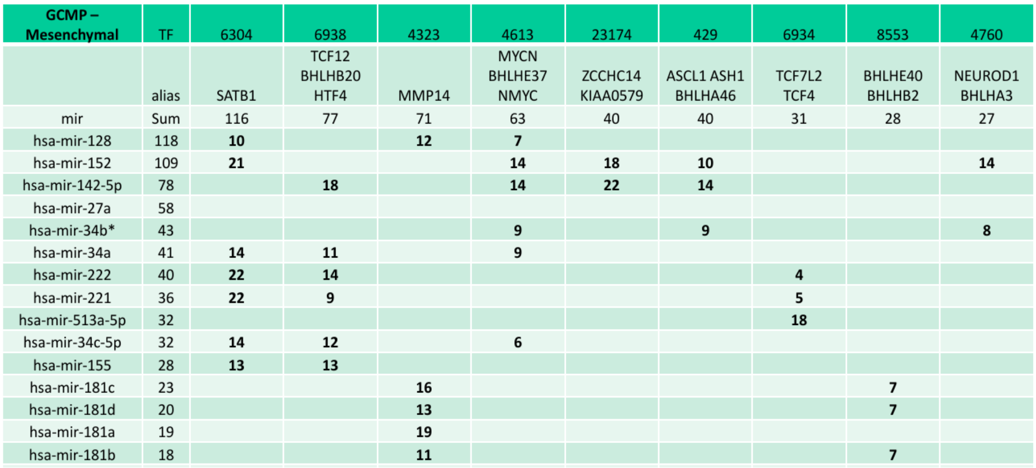
The comparison of GBM phenotypes highlighted a set of miRNAs potentially responsible for mantaining a specific phenotype. The analysis was performed by adapting MARINa, a FCS pipeline for GSEA, to the repressive action of miRNAs. A pruning strategy was introduced to narrow down all the predicted miRNA-mRNA interactions to a disease-specific network.

Selected miRNAs should be now be studied from a biological perspective, and validated. MiRNA interference experiments, where the action of a miRNA is inhibited through the treatment with specific molecules, might determine whether predicted miRNAs are in fact responsible for mantaining a specific phenotype. Such analysis, however, is beyond the purposes of this thesis and the competences of the author.

From a computational perspective the pipeline could be further validated by analyzing different datasets. In particular, in would be interesting to determine whether the list of differentially active miRNAs obtained in this work is similar to the results on other datasets. It is also possible to select different sample subsets from the current dataset, and verify whether the results are similar. As a preliminary step in this sense, MARINa was tested on different sample classifications. Verhaak classification of GBM samples was repeated (data not yet published), and MARINa was tested both on this new classification, and on the intersection of the two. The results were very similar. In particular, the same miRNAs were selected as significant in all the three studies.

Only the direct targets of miRNAs have been taken into account to build the regulons. This is sound, in the context of the noisy regulatory network currently available. A more sophisticated heuristic might extend the regulons beyond the first level of regulation. This is appealing, considering that transcriptional and post-transcriptional networks will become more and more reliable.

Finally, an important consideration regards the combinations of effects of single master regulators. Indeed, a phenotype is not usually mantained by the action of a single master regulator. Several MRs are generally selected in GSEA studies (as in this study, or in [99]), and many of them act synergistically to drive the analyzed cell phenotypes. There are many forms of “collaboration” between MRs. For instance, they usually share some common targets. Synergistic studies, aimed at identifying pairs or groups of MRs working together, have already been proposed [99, 100]. Briefly, in their analysis the shared components of the overlapping regulons of two differentially active TFs is selected as common regulon, and the MRA is repeated on it. If the common regulon is more differentially active than the single original ones, then a synergistic pair of TFs is identified. Other types of collaboration can be studied as well. For instance, the integrated transcriptional and post-transcriptional network of GBM built in this thesis enables the identification of shared targets between different types of regulators, such as TFs and miRNAs. It might be interesting to focus on Feed Forward Loops (FFLs) involving two MRs and a shared target. FFLs are topological motifs broadly studied in Biology. There is an extensive literature about their origin and their importance, with precise mathematical models of their behaviour. As a preliminary study, the regulons of TFs and miRNAs identified as MRs in the comparison of GCMP+ and Mesenchymal subtypes were interesected. Only consistent FFLs has been considered in this context. A FFL is consistent if the expression of the three nodes involved is in agreement with the expected activity of the regulators. For example, let's suppose that in a FFL the miRNA downregulates the TF and the common target, and that the TF promotes the transcription of the common target. Then, if the miRNA's expression is higher in the first phenotype, and the expression of the TF is lower, then the expected expression of the common target should be lower as well. FFLs that do not satisfy such requirement are filtered out. Figure 3.10 shows the number of FFLs identified in the integrated regulatory network. The integration of FFLs identify a subgraph, represented in Figure 3.11, where miRNAs and TFs act together to consistently regulate a set of shared targets.



GCMP – Mesenchymal	TF	6304	6938	4323	4613	23174	429	6934	8553	4760
	alias	SATB1	TCF12 BHLHB20 HTF4	MMP14	MYCN BHLHE37 NMYC	ZCCHC14 KIAA0579	ASCL1 ASH1 BHLHA46	TCF7L2 TCF4	BHLHE40 BHLHB2	NEUROD1 BHLHA3
mir	Sum	116	77	71	63	40	40	31	28	27
hsa-mir-128	118	**10**		**12**	**7**					
hsa-mir-152	109	**21**			**14**	**18**	**10**			**14**
hsa-mir-142-5p	78		**18**		**14**	**22**	**14**			
hsa-mir-27a	58									
hsa-mir-34b*	43				**9**		**9**			**8**
hsa-mir-34a	41	**14**	**11**		**9**					
hsa-mir-222	40	**22**	**14**					**4**		
hsa-mir-221	36	**22**	**9**					**5**		
hsa-mir-513a-5p	32							**18**		
hsa-mir-34c-5p	32	**14**	**12**		**6**					
hsa-mir-155	28	**13**	**13**							
hsa-mir-181c	23			**16**					**7**	
hsa-mir-181d	20			**13**					**7**	
hsa-mir-181a	19			**19**						
hsa-mir-181b	18			**11**					**7**	

Figure 3.10: Consistent FFLs between differentially active miRNAs and TFs, in the GCMP+ vs Mesenchymal comparison.

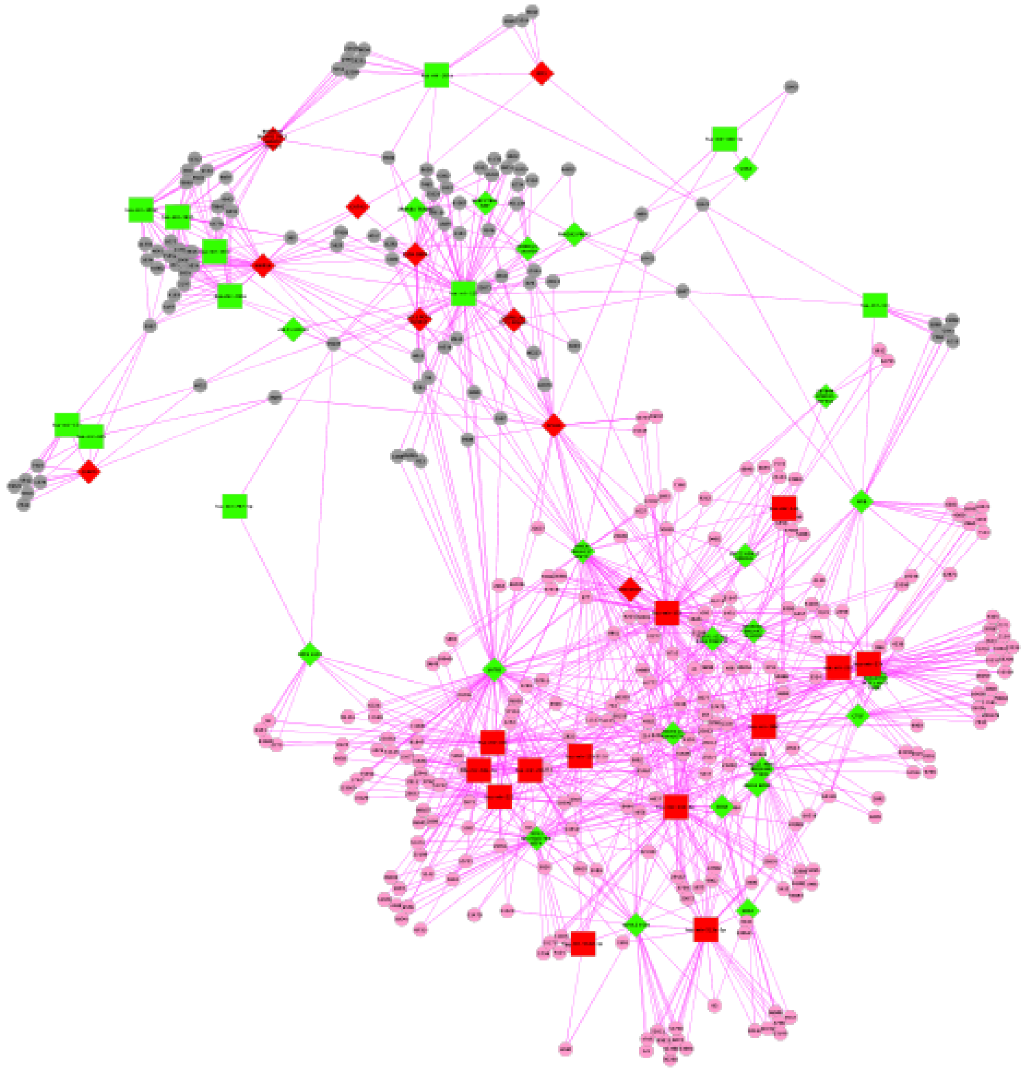


Figure 3.11: Regulatory network composed exclusively by the consistent FFLs involving differentially active miRNAs and TFs between GCMP+ and Mesenchymal subtypes. Small circles represent common target genes, with pink and gray ones being respectively up- and down- regulated. Green squares and rhombus represent TFs and miRNAs, respectively, more expressed in GCMP+. Red squares and rhombus, instead, represet TFs and miRNAs higher expressed in Mesenchymal subtype. Two mildly connected components are observable by visual inspection of the graph.

Conclusion

In this thesis computational techniques based on biological networks have been applied in two different contexts.

AlignNemo and AlignMCL, two algorithms that perform the local alignment of PPI networks and detect conserved complexes between species, have been presented in Chapter 2. The former introduces a sophisticate model of alignment graph, while the latter replaces the mining strategy of AlignNemo with the more efficient and generic Markov Clustering. A series of extensive assessments proved the quality of the proposed solutions. A significant contribution of this first work has been the improvement of previous techniques for filtering the noise out of PPI networks. Indeed, the proposed model of alignment graph exploits path redundancy to estimate the likelihood of potentially conserved interactions. This feature is particularly useful with data currently available, dominated by high levels of wrong or missing interactions. Indeed, it is easy to foresee the application of approaches based on path redundancy to other biological problems. On the other side, between the possible mining strategies, MCL has been selected for its scalability and ability of uncovering conserved modules without topological constraints.

In Chapter 3, the Master Regulator Analysis of miRNAs has been applied to the comparison of different classes of Glioblastoma Multiforme. Networks have been exploited to define the regulons of each miRNA. Beyond the adaptation of

MARINa, a pruning strategy to refine the first network into a more specific one has been presented. In this second work networks are only used to build regulons, with the core of the pipeline represented by the GSEA. As pointed out at the end of the chapter, however, possible extensions might stem from this preliminary work, potentially exploiting networks to an higher level. In any case, the idea behind the MRA is that a disease is a complex condition that affects different components of the cellular machinery. Such ability to influence different modules, potentially not overlapping, can be explained only through a network that links them together.

In a hopefully near future, when high-quality context-specific interactomes will be available, the explanatory power of network-based analysis techniques will increase considerably, with a significant impact on our understranding of life.

Bibliography

[1] Mario Cannataro, Pietro H. Guzzi, and Pierangelo Veltri. Protein-to-protein interactions. *ACM Computing Surveys*, 43(1):1–36, November 2010.

[2] Peter Uetz and Russell L Finley. From protein networks to biological systems. *FEBS letters*, 579(8):1821–7, March 2005.

[3] Yuen Ho, Albrecht Gruhler, Adrian Heilbut, Gary D Bader, Lynda Moore, Sally-lin Adams, Anna Millar, Paul Taylor, Keiryn Bennett, Kelly Boutilier, Lingyun Yang, Cheryl Wolting, Ian Donaldson, Sùren Schandorff, Juanita Shewnarane, Mai Vo, Joanne Taggart, Marilyn Goudreault, Brenda Muskat, Cris Alfarano, Danielle Dewar, Zhen Lin, Katerina Michalickova, Andrew R Willems, Holly Sassi, Peter A Nielsen, Lykke H Hansen, Hans Jespersen, Alexandre Podtelejnikov, Jesper Matthiesen, Ronald C Hendrickson, Frank Gleeson, Tony Pawson, Michael F Moran, Daniel Durocher, Matthias Mann, Christopher W V Hogue, Daniel Figeys, and Mike Tyers. Systematic identification of protein complexes in Saccharomyces cerevisiae by mass spectrometry. *Nature*, 415(January):2–5, 2002.

[4] Anne-Claude Gavin, Markus Bösche, Roland Krause, Paola Grandi, Martina Marzioch, Andreas Bauer, Jörg Schultz, Jens M Rick, Anne-Marie Michon, Cristina-Maria Cruciat, Marita Remor, Christian Höfert, Malgorzata Schelder, Miro Brajenovic, Heinz Ruffner, Alejandro Merino, Karin Klein, Manuela Hudak, David Dickson, Tatjana Rudi, Volker Gnau, Angela Bauch, Sonja Bastuck, Bettina Huhse, Christina Leutwein, Marie-Anne

Heurtier, Richard R Copley, Angela Edelmann, Erich Querfurth, Vladimir Rybin, Gerard Drewes, Manfred Raida, Tewis Bouwmeester, Peer Bork, Bertrand Seraphin, Bernhard Kuster, Gitte Neubauer, and Giulio Superti-Furga. Functional organization of the yeast proteome by systematic analysis of protein complexes. *Nature*, 415(6868):141–7, January 2002.

[5] T Ito, T Chiba, R Ozawa, M Yoshida, M Hattori, and Y Sakaki. A comprehensive two-hybrid analysis to explore the yeast protein interactome. *Proceedings of the National Academy of Sciences of the United States of America*, 98(8):4569–74, April 2001.

[6] P Uetz, L Giot, G Cagney, T a Mansfield, R S Judson, J R Knight, D Lockshon, V Narayan, M Srinivasan, P Pochart, a Qureshi-Emili, Y Li, B Godwin, D Conover, T Kalbfleisch, G Vijayadamodar, M Yang, M Johnston, S Fields, and J M Rothberg. A comprehensive analysis of protein-protein interactions in Saccharomyces cerevisiae. *Nature*, 403(6770):623–7, February 2000.

[7] Jason Flannick, Antal Novak, Balaji S Srinivasan, Harley H McAdams, and Serafim Batzoglou. Graemlin: general and robust alignment of multiple large interaction networks. *Genome research*, 16(9):1169–81, September 2006.

[8] Elena Zotenko, Katia S Guimarães, Raja Jothi, and Teresa M Przytycka. Decomposition of overlapping protein complexes: a graph theoretical method for analyzing static and dynamic protein associations. *Algorithms for molecular biology : AMB*, 1(1):7, January 2006.

[9] Andrew Douglas King. *Graph Clustering with Restricted Neighbourhood Search*. PhD thesis, 2004.

[10] GD Bader and CWV Hogue. An automated method for finding molecular complexes in large protein interaction networks. *BMC bioinformatics*, 27:1–27, 2003.

[11] Eric de Silva, Thomas Thorne, Piers Ingram, Ino Agrafioti, Jonathan Swire, Carsten Wiuf, and Michael P H Stumpf. The effects of incomplete protein interaction data on structural and evolutionary inferences. *BMC biology*, 4:39, January 2006.

[12] Luke Hakes, John W Pinney, David L Robertson, and Simon C Lovell. Protein-protein interaction networks and biology–what's the connection? *Nature biotechnology*, 26(1):69–72, January 2008.

[13] Roded Sharan, T. Ideker, B. Kelley, R. Shamir, and R.M. Karp. Identification of protein complexes by comparative analysis of yeast and bacterial protein interaction data. *Journal of Computational Biology*, 12(6):835–846, 2005.

[14] Mehmet Koyutürk, Yohan Kim, Umut Topkara, Shankar Subramaniam, Wojciech Szpankowski, and Ananth Grama. Pairwise alignment of protein interaction networks. *Journal of computational biology : a journal of computational molecular cell biology*, 13(2):182–99, March 2006.

[15] B.P. Kelley, Roded Sharan, R.M. Karp, T. Sittler, D.E. Root, B.R. Stockwell, and T. Ideker. Conserved pathways within bacteria and yeast as revealed by global protein network alignment. *PNAS*, 100(20):11394, 2003.

[16] Roded Sharan, S Suthram, RM Kelley, and T. Conserved patterns of protein interaction in multiple species. *PNAS*, 102:1974–1979, 2005.

[17] Sourav Bandyopadhyay, Roded Sharan, and Trey Ideker. Systematic identification of functional orthologs based on protein network comparison. *Genome research*, 16:428–435, 2006.

[18] Jason Flannick, Antal Novak, Chuong B Do, Balaji S Srinivasan, and Serafim Batzoglou. Automatic parameter learning for multiple local network alignment. *Journal of computational biology*, 16(8):1001–22, August 2009.

[19] Rohit Singh, Jinbo Xu, and Bonnie Berger. Pairwise global alignment of protein interaction networks by matching neighborhood topology. In *Research in Computational Molecular Biology*, pages 16–31. 2007.

[20] Rohit Singh, Jinbo Xu, and Bonnie Berger. Global alignment of multiple protein interaction networks with application to functional orthology detection. *PNAS*, 105(35):12763–12768, 2008.

[21] Chung-Shou Liao, Kanghao Lu, Michael Baym, Rohit Singh, and Bonnie Berger. IsoRankN: spectral methods for global alignment of multiple protein networks. *Bioinformatics*, 25(12):i253–8, June 2009.

[22] Oleksii Kuchaiev and Nataša Pržulj. Integrative network alignment reveals large regions of global network similarity in yeast and human. *Bioinformatics (Oxford, England)*, 27(10):1390–6, May 2011.

[23] Tijana Milenković, Weng Leong, and Nataša Pržulj. Optimal network Alignment with Graphlet Degree Vectors. *Cancer Informatics*, 9:121–137, 2010.

[24] Vesna Memišević and Nataša Pržulj. C-GRAAL: Common-neighbors-based global GRAph ALignment of biological networks. *Integrative biology : quantitative biosciences from nano to macro*, January 2012.

[25] Wenhong Tian and Nagiza F Samatova. Pairwise Alignment of Interaction Networks by Fast Identification of Maximal Conserved Patterns. *Symposium A Quarterly Journal In Modern Foreign Literatures*, 110:99–110, 2009.

[26] Adrian P Cootes, Stephen H Muggleton, and Michael J E Sternberg. The identification of similarities between biological networks: application to the metabolome and interactome. *Journal of molecular biology*, 369(4):1126–39, June 2007.

[27] Roland a. Pache and Patrick Aloy. A Novel Framework for the Comparative Analysis of Biological Networks. *PLoS ONE*, 7(2):e31220, February 2012.

[28] Johannes Berg and M Lässig. Local graph alignment and motif search in biological networks. *Proceedings of the National Academy of ...*, 101(41):14689–14694, 2004.

[29] Michal Kolár, Michael Lässig, and Johannes Berg. From protein interactions to functional annotation: graph alignment in Herpes. *BMC systems biology*, 2:90, January 2008.

[30] Giovanni Ciriello, Marco Mina, Pietro H Guzzi, Mario Cannataro, and Concettina Guerra. AlignNemo: A Local Network Alignment Method to Integrate Homology and Topology. *PloS one*, 7(6):e38107, January 2012.

[31] Kevin P O'Brien, Maido Remm, and Erik L L Sonnhammer. Inparanoid: a comprehensive database of eukaryotic orthologs. *Nucleic acids research*, 33(Database issue):D476–80, January 2005.

[32] Paul Jaccard. Étude comparative de la distribution florale dans une portion des Alpes et des Jura. *Bulletin de la Société Vaudoise des Sciences Naturelles*, pages 547–579, 1901.

[33] Ioannis Xenarios, Danny W. Rice, Lukasz Salwinski, Marisa K. Baron, Edward M. Marcotte, and David Eisenberg. DIP: the database of interacting proteins. *Nucleic Acids Research*, 28(1):289–291, January 2000.

[34] Martin H. Schaefer, Jean-Fred Fontaine, Arunachalam Vinayagam, Pablo Porras, Erich E. Wanker, and Miguel a. Andrade-Navarro. HIPPIE: Integrating Protein Interaction Networks with Experiment Based Quality Scores. *PLoS ONE*, 7(2):e31826, February 2012.

[35] Haiyuan Yu, Nicholas M Luscombe, Hao Xin Lu, Xiaowei Zhu, Yu Xia, Jing-Dong J Han, Nicolas Bertin, Sambath Chung, Marc Vidal, and Mark Gerstein. Annotation transfer between genomes: protein-protein interologs and protein-DNA regulogs. *Genome research*, 14(6):1107–18, June 2004.

[36] Minghua Deng, Fengzhu Sun, and Ting Chen. Assessment of the reliability of protein-protein interactions and protein function prediction. *Pacific Sym-*

posium on Biocomputing. Pacific Symposium on Biocomputing, 151:140–51, January 2003.

[37] J Michael Cherry, Caroline Adler, Catherine Ball, Stephen A. Chervitz, Selina S. Dwight, and Erich T. Hester. SGD: Saccharomyces Genome Database. *Nucleic acids research*, 38(Database issue):D433–6, January 2010.

[38] Shuye Pu, Jessica Wong, Brian Turner, Emerson Cho, and Shoshana J Wodak. Up-to-date catalogues of yeast protein complexes. *Nucleic acids research*, 37(3):825–31, February 2009.

[39] Andreas Ruepp, Barbara Brauner, Irmtraud Dunger-Kaltenbach, Goar Frishman, Corinna Montrone, Michael Stransky, Brigitte Waegele, Thorsten Schmidt, Octave Noubibou Doudieu, Volker Stümpflen, and H Werner Mewes. CORUM: the comprehensive resource of mammalian protein complexes. *Nucleic acids research*, 36(Database issue):D646–50, January 2008.

[40] PH Guzzi, Marco Mina, Concettina Guerra, and Mario Cannataro. Semantic similarity analysis of protein data: assessment with biological features and issues. *Briefings in bioinformatics*, 13(5):569–585, 2012.

[41] Philip Resnik. Using information content to evaluate semantic similarity in a taxonomy. *Proceedings of the 14th International Joint Conference on Artificial Intelligence*, 1, 1995.

[42] Philip Resnik. Semantic similarity in a taxonomy: An information-based measure and its application to problems of ambiguity in natural language. *Journal of Artificial Intelligence Research*, 11:95–130, 1999.

[43] Daniel Faria, Catia Pesquita, F.M. Couto, and A. Falcão. Proteinon: A web tool for protein semantic similarity. Technical report, 2007.

[44] Marco Mina. Fastsemsim, http://sourceforge.net/p/fastsemsim/home/.

[45] Elizabeth I Boyle, Shuai Weng, Jeremy Gollub, Heng Jin, David Botstein, J Michael Cherry, and Gavin Sherlock. GO::TermFinder–open source software for accessing Gene Ontology information and finding significantly enriched Gene Ontology terms associated with a list of genes. *Bioinformatics (Oxford, England)*, 20(18):3710–5, December 2004.

[46] S. K. Burley and R. G. Roeder. Biochemistry and structural biology of transcription factor iid (tfiid). *Annu Rev Biochem*, 65:769–99, 1996.

[47] Insall Robert H. Veltman, Douwe M. Wasp family proteins: Their evolution and its physiological implications. *Molecular Biology of the Cell*, 21:2880–2893, 2010.

[48] a J Enright, S Van Dongen, and C a Ouzounis. An efficient algorithm for large-scale detection of protein families. *Nucleic acids research*, 30(7):1575–84, April 2002.

[49] Sylvain Brohée and Jacques van Helden. Evaluation of clustering algorithms for protein-protein interaction networks. *BMC bioinformatics*, 7:488, January 2006.

[50] a D King, N Przulj, and I Jurisica. Protein complex prediction via cost-based clustering. *Bioinformatics (Oxford, England)*, 20(17):3013–20, November 2004.

[51] Igor V Tetko, Axel Facius, Andreas Ruepp, and Hans-Werner Mewes. Super paramagnetic clustering of protein sequences. *BMC bioinformatics*, 6:82, January 2005.

[52] Kevin R Brown and Igor Jurisica. Online predicted human interaction database. *Bioinformatics (Oxford, England)*, 21(9):2076–82, May 2005.

[53] Jingkai Yu, Svetlana Pacifico, Guozhen Liu, and Russell Finley. DroID: the drosophila interactions database, a comprehensive resource for annotated gene and protein interactions. *BMC Genomics*, 9(1):461+, October 2008.

[54] Yanhui Hu, Ian Flockhart, Arunachalam Vinayagam, Clemens Bergwitz, Bonnie Berger, Norbert Perrimon, and Stephanie E Mohr. An integrative approach to ortholog prediction for disease-focused and other functional studies. *BMC bioinformatics*, 12:357, January 2011.

[55] Christiam Camacho, George Coulouris, Vahram Avagyan, Ning Ma, Jason Papadopoulos, Kevin Bealer, and Thomas L Madden. BLAST+: architecture and applications. *BMC bioinformatics*, 10(421), January 2009.

[56] NCBI. National center for biotechnology information, http://www.ncbi.nlm.nih.gov/.

[57] K.G. Guruharsha, Jean-François Rual, Bo Zhai, Julian Mintseris, Pujita Vaidya, Namita Vaidya, Chapman Beekman, Christina Wong, DavidÂ Y.

Rhee, Odise Cenaj, Emily McKillip, Saumini Shah, Mark Stapleton, KennethÂ H. Wan, Charles Yu, Bayan Parsa, JosephÂ W. Carlson, Xiao Chen, Bhaveen Kapadia, K. VijayRaghavan, StevenÂ P. Gygi, SusanÂ E. Celniker, RobertÂ A. Obar, and Spyros Artavanis-Tsakonas. A Protein Complex Network of Drosophila melanogaster. *Cell*, 147(3):690–703, October 2011.

[58] Matthias E. Futschik, Gautam Chaurasia, and Hanspeter Herzel. Comparison of human protein-protein interaction maps. *Bioinformatics*, 23(5):605–611, 2007.

[59] Da Wei Huang, Brad T Sherman, and Richard a Lempicki. Bioinformatics enrichment tools: paths toward the comprehensive functional analysis of large gene lists. *Nucleic acids research*, 37(1):1–13, January 2009.

[60] Xiang Guo, Rongxiang Liu, Craig D Shriver, Hai Hu, and Michael N Liebman. Assessing semantic similarity measures for the characterization of human regulatory pathways. *Bioinformatics (Oxford, England)*, 22(8):967–73, April 2006.

[61] Roded Sharan and Trey Ideker. Modeling cellular machinery through biological network comparison. *Nature biotechnology*, 24(4):427–33, April 2006.

[62] Nakamura Ashwini, Patil Haruki. Hint: a database of annotated protein-protein interactions and their homologs.

[63] Nicolas Simonis, Jean-François F. Rual, Anne-Ruxandra R. Carvunis, Murat Tasan, Irma Lemmens, Tomoko Hirozane-Kishikawa, Tong Hao,

Julie M. Sahalie, Kavitha Venkatesan, Fana Gebreab, Sebiha Cevik, Niels Klitgord, Changyu Fan, Pascal Braun, Ning Li, Nono Ayivi-Guedehoussou, Elizabeth Dann, Nicolas Bertin, David Szeto, Amélie Dricot, Muhammed A. Yildirim, Chenwei Lin, Anne-Sophie S. de Smet, Huey-Ling L. Kao, Christophe Simon, Alex Smolyar, Jin Sook S. Ahn, Muneesh Tewari, Mike Boxem, Stuart Milstein, Haiyuan Yu, Matija Dreze, Jean Vandenhaute, Kristin C. Gunsalus, Michael E. Cusick, David E. Hill, Jan Tavernier, Frederick P. Roth, and Marc Vidal. Empirically controlled mapping of the caenorhabditis elegans protein-protein interactome network. *Nature methods*, 6(1):47–54, January 2009.

[64] John H Morris, Leonard Apeltsin, Aaron M Newman, Jan Baumbach, Tobias Wittkop, Gang Su, Gary D Bader, and Thomas E Ferrin. clusterMaker: a multi-algorithm clustering plugin for Cytoscape. *BMC bioinformatics*, 12(1):436, January 2011.

[65] Catia Pesquita, Daniel Faria, Hugo Bastos, António E N Ferreira, André O Falcão, and Francisco M Couto. Metrics for GO based protein semantic similarity: a systematic evaluation. *BMC bioinformatics*, 9 Suppl 5:S4, January 2008.

[66] YK Shih and Srinivasan Parthasarathy. Scalable global alignment for multiple biological networks. *BMC Bioinformatics*, 13(Suppl 3):S11, 2012.

[67] Purvesh Khatri, Marina Sirota, and Atul J Butte. Ten years of pathway analysis: current approaches and outstanding challenges. *PLoS computational biology*, 8(2):e1002375, January 2012.

[68] Jui-Hung Hung, Tun-Hsiang Yang, Zhenjun Hu, Zhiping Weng, and Charles DeLisi. Gene set enrichment analysis: performance evaluation and usage guidelines. *Briefings in bioinformatics*, 13(3):281–91, May 2012.

[69] Jelle J Goeman and Peter Bühlmann. Analyzing gene expression data in terms of gene sets: methodological issues. *Bioinformatics (Oxford, England)*, 23(8):980–7, April 2007.

[70] Dougu Nam and Seon-Young Kim. Gene-set approach for expression pattern analysis. *Briefings in bioinformatics*, 9(3):189–97, May 2008.

[71] Marit Ackermann and Korbinian Strimmer. A general modular framework for gene set enrichment analysis. *BMC bioinformatics*, 10:47, January 2009.

[72] Lily Wang, Peilin Jia, Russell D Wolfinger, Xi Chen, and Zhongming Zhao. Gene set analysis of genome-wide association studies: methodological issues and perspectives. *Genomics*, 98(1):1–8, July 2011.

[73] Pablo Tamayo, George Steinhardt, Arthur Liberzon, and Jill P Mesirov. The limitations of simple gene set enrichment analysis assuming gene independence. *Statistical methods in medical research*, October 2012.

[74] Lin S Chen, Carolyn M Hutter, John D Potter, Yan Liu, Ross L Prentice, Ulrike Peters, and Li Hsu. Insights into colon cancer etiology via a regularized approach to gene set analysis of GWAS data. *American journal of human genetics*, 86(6):860–71, June 2010.

[75] Qi Liu, Irina Dinu, Adeniyi J Adewale, John D Potter, and Yutaka Yasui. Comparative evaluation of gene-set analysis methods. *BMC bioinformatics*, 8:431, January 2007.

[76] Luca Abatangelo, Rosalia Maglietta, Angela Distaso, Annarita D'Addabbo, Teresa Maria Creanza, Sayan Mukherjee, and Nicola Ancona. Comparative study of gene set enrichment methods. *BMC bioinformatics*, 10:275, January 2009.

[77] Galina V Glazko and Frank Emmert-Streib. Unite and conquer: univariate and multivariate approaches for finding differentially expressed gene sets. *Bioinformatics (Oxford, England)*, 25(18):2348–54, September 2009.

[78] Haroon Naeem, Ralf Zimmer, Pegah Tavakkolkhah, and Robert Küffner. Rigorous assessment of gene set enrichment tests. *Bioinformatics (Oxford, England)*, 28(11):1480–6, June 2012.

[79] Aravind Subramanian, Pablo Tamayo, Vamsi K Mootha, Sayan Mukherjee, Benjamin L Ebert, Michael a Gillette, Amanda Paulovich, Scott L Pomeroy, Todd R Golub, Eric S Lander, and Jill P Mesirov. Gene set enrichment analysis: a knowledge-based approach for interpreting genome-wide expression profiles. *Proceedings of the National Academy of Sciences of the United States of America*, 102(43):15545–50, October 2005.

[80] Christine Steinhoff and Martin Vingron. Normalization and quantification of differential expression in gene expression microarrays. *Briefings in bioinformatics*, 7(2):166–77, June 2006.

[81] Rafael a Irizarry, Bridget Hobbs, Francois Collin, Yasmin D Beazer-Barclay, Kristen J Antonellis, Uwe Scherf, and Terence P Speed. Exploration, normalization, and summaries of high density oligonucleotide array probe level data. *Biostatistics (Oxford, England)*, 4(2):249–64, April 2003.

[82] Tero Aittokallio. Dealing with missing values in large-scale studies: microarray data imputation and beyond. *Briefings in bioinformatics*, 11(2):253–64, March 2010.

[83] Guy N Brock, John R Shaffer, Richard E Blakesley, Meredith J Lotz, and George C Tseng. Which missing value imputation method to use in expression profiles: a comparative study and two selection schemes. *BMC bioinformatics*, 9:12, January 2008.

[84] Fangxin Hong and Rainer Breitling. A comparison of meta-analysis methods for detecting differentially expressed genes in microarray experiments. *Bioinformatics (Oxford, England)*, 24(3):374–82, February 2008.

[85] Fisher R. *Statistical Methods for ResearchWorkers.* 1932.

[86] Thomas Aigner Ralf Zimmer Katrin Fundel, Robert Küffner. Normalization and gene p-value estimation: issues in microarray data processing. *Bioinformatics and biology insights*, (2):291–305, 2008.

[87] Vishal Saxena, Dennis Orgill, and Isaac Kohane. Absolute enrichment: gene set enrichment analysis for homeostatic systems. *Nucleic acids research*, 34(22):e151, January 2006.

[88] Vamsi K Mootha, Cecilia M Lindgren, Karl-Fredrik Eriksson, Aravind Subramanian, Smita Sihag, Joseph Lehar, Pere Puigserver, Emma Carlsson, Martin Ridderstrå le, Esa Laurila, Nicholas Houstis, Mark J Daly, Nick Patterson, Jill P Mesirov, Todd R Golub, Pablo Tamayo, Bruce Spiegelman, Eric S Lander, Joel N Hirschhorn, David Altshuler, and Leif C Groop. PGC-1alpha-responsive genes involved in oxidative phosphorylation are coordinately downregulated in human diabetes. *Nature genetics*, 34(3):267–73, July 2003.

[89] Zhen Jiang and Robert Gentleman. Extensions to gene set enrichment. *Bioinformatics (Oxford, England)*, 23(3):306–13, February 2007.

[90] William T Barry, Andrew B Nobel, and Fred a Wright. Significance analysis of functional categories in gene expression studies: a structured permutation approach. *Bioinformatics (Oxford, England)*, 21(9):1943–9, May 2005.

[91] Bradley Efron and Robert Tibshirani. On testing the significance of sets of genes. *Annals of Applied Statistics*, 1(1):107–129, June 2007.

[92] Lu Tian, Steven a Greenberg, Sek Won Kong, Josiah Altschuler, Isaac S Kohane, and Peter J Park. Discovering statistically significant pathways in expression profiling studies. *Proceedings of the National Academy of Sciences of the United States of America*, 102(38):13544–9, September 2005.

[93] Irina Dinu, John D Potter, Thomas Mueller, Qi Liu, Adeniyi J Adewale, Gian S Jhangri, Gunilla Einecke, Konrad S Famulski, Philip Halloran, and

Yutaka Yasui. Gene-set analysis and reduction. *Briefings in bioinformatics*, 10(1):24–34, January 2009.

[94] Ning-yi Shao, Hai Yang Hu, Zheng Yan, Ying Xu, Hao Hu, Corinna Menzel, Na Li, Wei Chen, and Philipp Khaitovich. Comprehensive survey of human brain microRNA by deep sequencing. *BMC Genomics*, 2010.

[95] André Boorsma, Xiang-Jun Lu, Anna Zakrzewska, Frans M Klis, and Harmen J Bussemaker. Inferring condition-specific modulation of transcription factor activity in yeast through regulon-based analysis of genomewide expression. *PloS one*, 3(9):e3112, January 2008.

[96] MahavisnoV Barrette TR Ghosh D Chinnaiyan AM Rhodes DR, Kalyana-Sundaram S. Mining for regulatory programs in the cancer transcriptome. *Nature genetics*, 37(4):579–583, 2005.

[97] Adam a Margolin, Ilya Nemenman, Katia Basso, Chris Wiggins, Gustavo Stolovitzky, Riccardo Dalla Favera, and Andrea Califano. ARACNE: an algorithm for the reconstruction of gene regulatory networks in a mammalian cellular context. *BMC bioinformatics*, 7 Suppl 1:S7, January 2006.

[98] Margolin AA Nemenman I Califano A Wang K, Banerjee N. Genome-wide discovery of modulators of transcriptional interactions in human B lymphocytes. 3909:348e62, 2006.

[99] Celine Lefebvre, Presha Rajbhandari, Mariano J Alvarez, Pradeep Bandaru, Wei Keat Lim, Mai Sato, Kai Wang, Pavel Sumazin, Manjunath Kustagi, Brygida C Bisikirska, Katia Basso, Pedro Beltrao, Nevan Krogan, Jean

Gautier, Riccardo Dalla-Favera, and Andrea Califano. A human B-cell interactome identifies MYB and FOXM1 as master regulators of proliferation in germinal centers. *Molecular systems biology*, 6(377):377, June 2010.

[100] Maria Stella Carro, Wei Keat Lim, Mariano Javier Alvarez, Robert J Bollo, Xudong Zhao, Evan Y Snyder, Erik P Sulman, Sandrine L Anne, Fiona Doetsch, Howard Colman, Anna Lasorella, Ken Aldape, Andrea Califano, and Antonio Iavarone. The transcriptional network for mesenchymal transformation of brain tumours. *Nature*, 463(7279):318–25, January 2010.

[101] WK Lim, E Lyashenko, and Andrea Califano. Master regulators used as breast cancer metastasis classifier. *Pacific Symposium on Biocomputing*, pages 504–515, 2009.

[102] Marc J. van de Vijver. A gene-expression signature as a predictor of survival in breast cancer. *…England Journal of …*, 347(25):1999–2009, 2002.

[103] Yixin Wang, Jan G M Klijn, Yi Zhang, Anieta M Sieuwerts, Maxime P Look, Fei Yang, Dmitri Talantov, Mieke Timmermans, Marion E Meijer-van Gelder, Jack Yu, Tim Jatkoe, Els M J J Berns, David Atkins, and John a Foekens. Gene-expression profiles to predict distant metastasis of lymph-node-negative primary breast cancer. *Lancet*, 365(9460):671–9, 2005.

[104] Heidi S Phillips, Samir Kharbanda, Ruihuan Chen, William F Forrest, Robert H Soriano, Thomas D Wu, Anjan Misra, Janice M Nigro, Howard Colman, Liliana Soroceanu, P Mickey Williams, Zora Modrusan, Burt G Feuerstein, and Ken Aldape. Molecular subclasses of high-grade glioma

predict prognosis, delineate a pattern of disease progression, and resemble stages in neurogenesis. *Cancer cell*, 9(3):157–73, March 2006.

[105] Katia Basso, Adam a Margolin, Gustavo Stolovitzky, Ulf Klein, Riccardo Dalla-Favera, and Andrea Califano. Reverse engineering of regulatory networks in human B cells. *Nature genetics*, 37(4):382–90, April 2005.

[106] Ramamurthy Mani, Robert P St Onge, John L Hartman, Guri Giaever, and Frederick P Roth. Defining genetic interaction. *Proceedings of the National Academy of Sciences of the United States of America*, 105(9):3461–6, March 2008.

[107] H Zhang, Y Li, and M Lai. The microRNA network and tumor metastasis. *Oncogene*, 29(7):937–48, February 2010.

[108] Roel G W Verhaak, Katherine a Hoadley, Elizabeth Purdom, Victoria Wang, Yuan Qi, Matthew D Wilkerson, C Ryan Miller, Li Ding, Todd Golub, Jill P Mesirov, Gabriele Alexe, Michael Lawrence, Michael O'Kelly, Pablo Tamayo, Barbara a Weir, Stacey Gabriel, Wendy Winckler, Supriya Gupta, Lakshmi Jakkula, Heidi S Feiler, J Graeme Hodgson, C David James, Jann N Sarkaria, Cameron Brennan, Ari Kahn, Paul T Spellman, Richard K Wilson, Terence P Speed, Joe W Gray, Matthew Meyerson, Gad Getz, Charles M Perou, and D Neil Hayes. Integrated genomic analysis identifies clinically relevant subtypes of glioblastoma characterized by abnormalities in PDGFRA, IDH1, EGFR, and NF1. *Cancer cell*, 17(1):98–110, January 2010.

[109] Houtan Noushmehr, Daniel J Weisenberger, Kristin Diefes, Heidi S Phillips, Kanan Pujara, Benjamin P Berman, Fei Pan, Christopher E Pelloski, Erik P Sulman, Krishna P Bhat, Roel G W Verhaak, Katherine a Hoadley, D Neil Hayes, Charles M Perou, Heather K Schmidt, Li Ding, Richard K Wilson, David Van Den Berg, Hui Shen, Henrik Bengtsson, Pierre Neuvial, Leslie M Cope, Jonathan Buckley, James G Herman, Stephen B Baylin, Peter W Laird, and Kenneth Aldape. Identification of a CpG island methylator phenotype that defines a distinct subgroup of glioma. *Cancer cell*, 17(5):510–22, May 2010.

[110] Pavel Sumazin, Xuerui Yang, Hua-Sheng Chiu, Wei-Jen Chung, Archana Iyer, David Llobet-Navas, Presha Rajbhandari, Mukesh Bansal, Paolo Guarnieri, Jose Silva, and Andrea Califano. An Extensive MicroRNA-Mediated Network of RNA-RNA Interactions Regulates Established Oncogenic Pathways in Glioblastoma. *Cell*, 147(2):370–381, October 2011.

[111] Hyunsoo Kim, Wei Huang, Xiuli Jiang, Brenton Pennicooke, Peter J Park, and Mark D Johnson. Integrative genome analysis reveals an oncomir/oncogene cluster regulating glioblastoma survivorship. *Proceedings of the National Academy of Sciences of the United States of America*, 107(5):2183–8, February 2010.

[112] TCGA. The cancer genome atlas, http://cancergenome.nih.gov/.

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

[114] Michael Kertesz, Nicola Iovino, Ulrich Unnerstall, Ulrike Gaul, and Eran Segal. The role of site accessibility in microRNA target recognition. *Nat Genetics*, 39(10):1278–84, 2007.

[115] Doron Betel, Manda Wilson, Aaron Gabow, Debora S Marks, and Chris Sander. The microRNA.org resource: targets and expression. *Nucleic acids research*, 36(Database issue):D149–53, January 2008.

[116] Adam Siepel, Gill Bejerano, Jakob S Pedersen, Angie S Hinrichs, Minmei Hou, Kate Rosenbloom, Hiram Clawson, John Spieth, Ladeana W Hillier, Stephen Richards, George M Weinstock, Richard K Wilson, Richard a Gibbs, W James Kent, Webb Miller, and David Haussler. Evolutionarily conserved elements in vertebrate, insect, worm, and yeast genomes. *Genome research*, 15(8):1034–50, August 2005.

[117] B. L. Welch. The generalization of "Student's" problem when several different population variances are involved. *Biometrika*, 34(1-2):28–35, 1947.

[118] Lei Shi, Zihao Cheng, Junxia Zhang, Rui Li, Peng Zhao, Zhen Fu, and Yongping You. Hsa-Mir-181a and Hsa-Mir-181B Function As Tumor Suppressors in Human Glioma Cells. *Brain research*, 1236:185–93, October 2008.

[119] Wei Meng, Ling Jiang, L Lu, H Hu, and Hailang Yu. Anti-miR-155 oligonucleotide enhances chemosensitivity of U251 cell to taxol by inducing apoptosis. *Cell biology*, 2012.