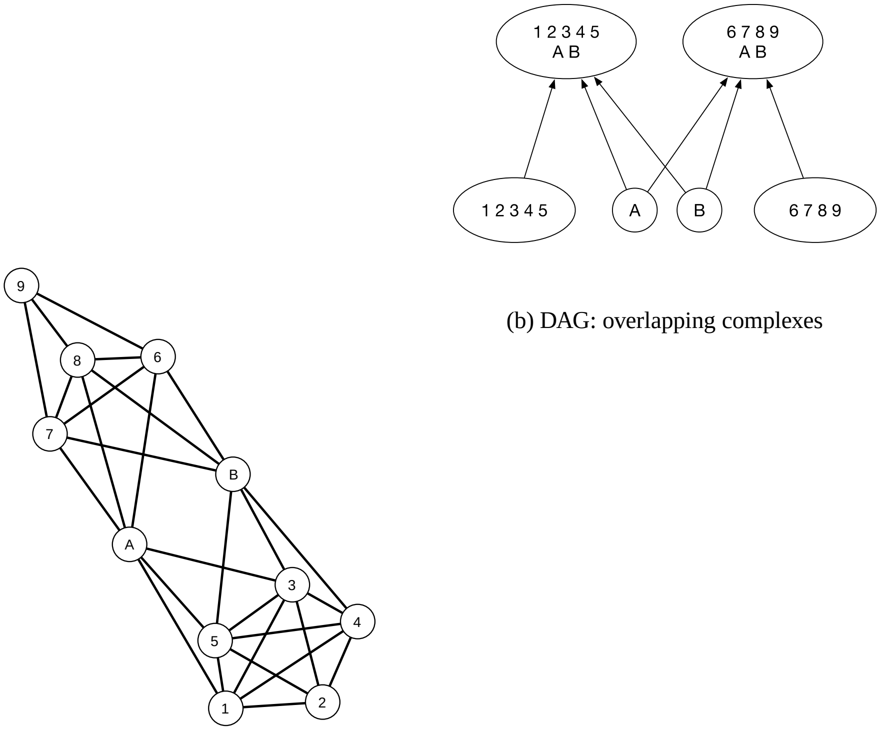
Chapter 6

Visualization of purifications

As we have seen in previous chapters, purification results contain significant observational error. An alternative to automatic methods to deal with this error is to represent the large amounts of raw observational data in a way that allows human introspection.

In particular, highly sensitive mass spectrometers identify many proteins that are only present in substoichiometric amounts and that might not be *bona fide* interactors, showing up as false positives in the observation. Typical contaminants are abundantly expressed in the cell and occur frequently in biochemical purifications. Consequently, these contribute to many complexes and may even appear as pivotal components (called hubs) in the interaction networks [25]. The approaches described in [14] and [37] try to solve the problem by considering interactions with proteins as weaker if the protein was found often. The approach investigated in this chapter is to provide a visual mapping to identify hub proteins and shared modules with respect to its purifications. This task is crucial in the interpretation of overlapping complexes from the experimental data. It is related to the problem of finding functional modules in protein-protein interaction networks [34, 41], which in turn are similar to those that were used to delineate modules in gene expression data [12] or protein sequences [6, 28].

Graph-based clustering approaches provided initial results [4, 14, 37, 42], but these methods aim to be independent of the actual experimental methods used to generate the protein-protein interaction data. Unfortunately, in doing so, detailed experimental information is eliminated, for instance, whether a given protein was actually tested as a bait or whether it was retrieved as prey. The bait-prey relation established experimentally is obviously asymmetric. Our method generates inclusion



(b) DAG: overlapping complexes

(a) Pairwise interaction graph

Figure 6.1: Two complexes interacting with shared components.

probabilities based on the purifications and therefore providing a concise representation of shared components.

Shared components, proteins that are part of several distinct complexes, make it difficult to define complex boundaries. A typical case is depicted in Fig 6.1, where two alternative sub-complexes interact with a central shared component. A successful method for generating complexes from this data needs to take these shared components into account. If it relies on the interaction graph alone, it needs to cut the graph into four components, two complexes { 1,2,3,4,5 }, { 6,7,8,9 } and two shared components { A } and { B }, assign the shared component to individual complexes or consider all three parts as a single entity.

A real world example of such complexes are the RNA polymerases, three homologous complexes that share many components with each other. Methods that aim to detect functional modules [34, 41] considered finding a cluster of all RNA polymerases a success, as all three complexes perform similar functions in the transcription of DNA into RNA. However, the three are clearly distinct in the experimental data. Most protein complexes with shared components are smaller than the RNA polymerases, which makes it even more difficult to detect these patterns.

Our framework relies on representing purifications as a directed graph, reflecting the asymmetric nature of the experiment. Inclusion probabilities are computed with a Bayesian approach to infer pair-wise posterior probabilities of inclusion of purified complexes from the pair-wise interaction probability. A simple graph algorithm identifies similar and different purifications. A resulting directed graph provides a mapping from experimental data to overlapping protein complexes. In the following we describe our approach and our evaluation in detail.

6.1 Systems and Methods

Our goal is to obtain groups of purifications representing the same protein complex or an ensemble of complexes. Our approach is motivated by a similar approach for clustering protein sequences which makes use of the asymmetry in the subsequence relationship between two proteins [6]. Here, we use the asymmetry of the bait-prey relationship. Strongly connected components represent a collection of purifications of the same complex. In addition, a purification is often a subset or subcomplex of a larger structure, which we can derive by doing a depth-first search on the graph of strongly connected components.

6.1.1 Directed-graph of purifications

Given N observable proteins, we record the collection of M observed purifications in the $M \times N$ matrix O. Let $B(i) \in \{1, \ldots, N\}$ be the bait protein of the ith purification. Note that two or more purifications can have the same bait. The entry $\hat{O}_{ik}$ is a pair (t, s): where s is the number of times the protein k observed to associate with $B(i)$ and t the number of trials testing the interaction pair $(B(i), k)$. This follows the observation model of protein interaction proposed by Gilchrist et al. [17].

If there were no noise in the experiment, the problem of identifying a complex from a collection of purifications can be reduced to finding the strongly connected component in the *unweighted* directed graph where an edge from i to j represents inclusion of purification i in purification j. In reality, all experimental methods that are currently in use to generate protein-protein interaction data also generate a substantial number of false-positive and false-negative associations. Based on the probabilistic model introduced by Gilchrist et al. [17], we formulate a probabilistic model of subset relationship that incorporates the probability of bait-prey interactions.

6.1.2 Method outline

- We begin by defining a complete weighted directed graph G whose vertex set represents a collection of M observed purifications. The node i represents the ith purification which consists of a set of proteins associated with its bait. The directed edge (i, j) is computed from our probabilistic model, $\mathbf{P}[H_i \leq H_j|\hat{O}_i, \hat{O}_j]$, the inclusion probability given the observation $\hat{O}_i, \hat{O}_j$, defined in Eq. 6.1 below. See Figure 6.2.

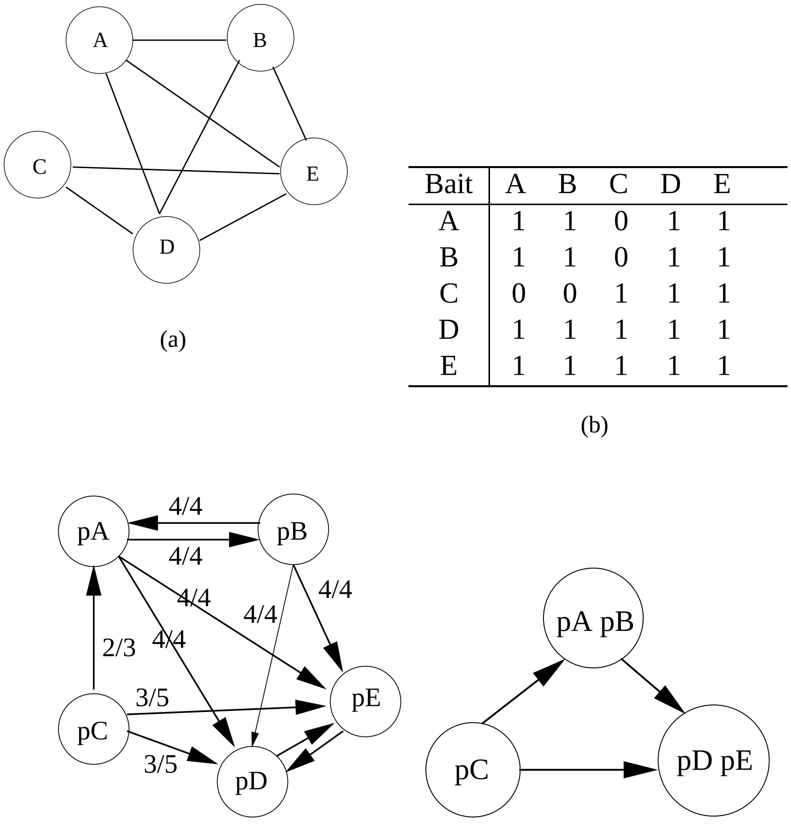
- Proceed to the threshold graph $G(\tau)$ (see Figure 6.2c) by removing all edges with the posterior probability less than τ. We tested different threshold values for high probability subset inclusions to keep.

- Compute all strongly connected components (SCCs) [39] in $G(\tau)$. The strongly connected components are maximal sets of vertices such that directed paths exist from P to Q and from Q to P for all vertices P, Q in a SCC. Define an SCC graph G_{SCC}, where each vertex corresponds to an extracted SCC of $G(\tau)$ and there is a directed edge from SCC$_A$ to SCC$_B$ iff there exists at least one edge (v, w) in $G(\tau)$ s.t. $v \in$ SCC$_A$ and $w \in$ SCC$_B$. The resulting SCC graph is a directed acyclic graph (DAG). A union of bait proteins belonging to the same SCC identify a collection of almost identical purifications. See Figure 6.2d. We then decompose the SCC graph into several subgraphs to describe possible sub-complex structure. See Figure 6.3.

6.1.3 A model of subset inclusion for purifications

In this section, we formulate the probabilistic model for the inclusion probability. Given two observed purifications $\hat{O}_i, \hat{O}_j$, we write the inclusion probability of two error-free purifications

Figure 6.2: Principles of the approach using idealized purifications. In the inclusion graph in Figure 6.2c, a weight from i to j indicates the number of similar proteins in nodes i and j over the total observed proteins associated to the node i. **2a** shows a given protein interaction graph. **2b** is a set of ideal purifications in matrix notation, each row in the matrix corresponding to one purification. The first column indicates the bait protein used in each row. **2c** shows an inclusion graph of purifications after applying a threshold. The edge weight from i to j indicates the number of similar proteins in nodes i and j over the total observed proteins associated to the node i. **2d** shows a sub-complex structure where nodes are strongly connected components in the threshold graph.



(a)

Bait	A	B	C	D	E
A	1	1	0	1	1
B	1	1	0	1	1
C	0	0	1	1	1
D	1	1	1	1	1
E	1	1	1	1	1

(b)

(c)

(d)

$H_i \leq H_j$ as follows:

$$\mathbf{P}[H_i \leq H_j|\hat{O}_i, \hat{O}_j] = \prod_{k \in \{1,\ldots,N\}} 1 - \mathbf{P}[H_{ik} > H_{jk}|\hat{O}_{ik}, \hat{O}_{jk}], \tag{6.1}$$

Assuming independence between proteins, we define $\mathbf{P}[H_{ik} > H_{jk}|\hat{O}_{ik}, \hat{O}_{jk}]$ by

$$\mathbf{P}[H_{ik} > H_{jk}|\hat{O}_{ik}, \hat{O}_{jk}] = \mathbf{P}[H_{ik} = 1, H_{jk} = 0|\hat{O}_{ik}, \hat{O}_{jk}], \tag{6.2}$$

and assuming independence between purifications H_i and H_j,

$$\mathbf{P}[H_{ik} > H_{jk}|\hat{O}_{ik}, \hat{O}_{jk}] = \mathbf{P}[H_{ik} = 1|\hat{O}_{ik}]\mathbf{P}[H_{jk} = 0|\hat{O}_{jk}]. \tag{6.3}$$

It follows from Bayes' Theorem that

$$\begin{aligned} \mathbf{P}[H_{ik} = 1|\hat{O}_{ik} = (t,s)] &= \frac{\mathbf{P}[\hat{O}_{ik} = (t,s)|H_{ik} = 1]\rho}{\mathbf{P}[\hat{O}_{ik} = (t,s)]} \qquad (6.4) \\ &= \frac{\mathbf{P}[\hat{O}_{ik} = (t,s)|H_{ik} = 1]\rho}{\mathbf{P}[\hat{O}_{ik} = (t,s)|H_{ik} = 1]\rho + \mathbf{P}[\hat{O}_{ik} = (t,s)|H_{ik} = 0](1-\rho)}, \end{aligned}$$

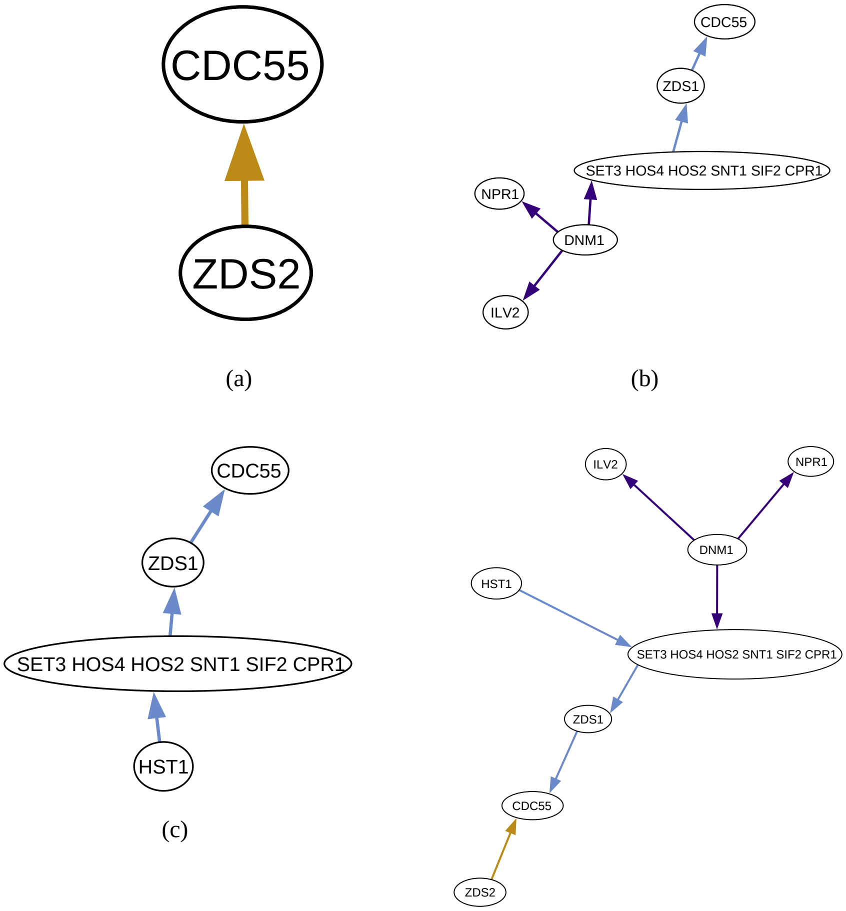
where the likelihood term $\mathbf{P}[\hat{O}_{ik}|H_{ik}]$ is defined by Eq. 2.1 and 2.2. The model depends on three parameters: ν, ϕ and ρ, which can be estimated from the data using maximum likelihood. Chapter 2 provides the mechanistic description of the statistical model for bait-prey interactions. See also Gilchrist et al. [17] for details and the numerical solver for ν, ϕ and ρ.

6.1.4 Decomposition of the SCC graph

The partial ordering encoded by the SCC graph is a model for sub-complex relationships between SCCs. Given a directed SCC graph G_{SCC} with vertices representing all SCCs in G, each path from a vertex A to a vertex B in the SCC graph provides a hypothesis on a possible sub-complex structure. We are interested in reachability structure between any two given SCCs. We decompose the SCC graph into a collection of subgraphs. For each vertex v of minimal order, which is one without incoming edges, we create a subgraph of the vertices reachable from v, $G^*(v) = (V^* \cup v, E^*(v))$, where V^* is the set of vertices reachable from v and $E^*(v)$ is the set of all edges connecting V^* to

v in the SCC graph. The graph $G^*(v)$ is computed by running a depth-first search starting at v for all minimal order vertices v. See Figure 6.3 for an illustration of decomposed components.

Figure 6.3: Decomposition of the SCC graph in **3d** into three subgraphs. The edges in the original graph **3d** are colored according to their assignment to the three subgraphs. Dashed edges indicate edges present in the transitive closure, but not in the original subgraph. Colors are assigned by matching the edges between the decomposition and the original graph.


CDC55
ZDS2
CDC55
ZDS1
SET3 HOS4 HOS2 SNT1 SIF2 CPR1
NPR1
DNM1
ILV2
CDC55
ZDS1
SET3 HOS4 HOS2 SNT1 SIF2 CPR1
HST1
ILV2
NPR1
DNM1
HST1
SET3 HOS4 HOS2 SNT1 SIF2 CPR1
ZDS1
CDC55
ZDS2

(a)

(b)

(c)

(d) Coloring of the original subgraph.

6.2 Implementation

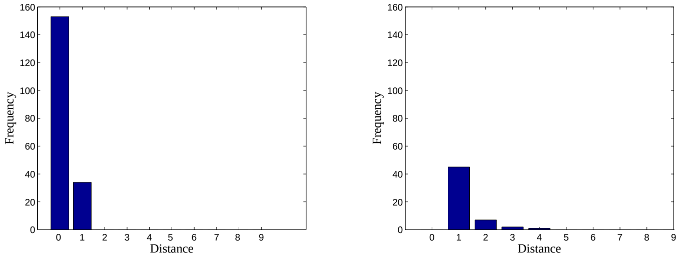
The software to perform the clustering was written in Python. It is available under the GPL at `http://algorithmics.molgen.mpg.de/Subcomplex/index.html`. The complete calculation takes about 750 seconds on an AMD Athlon 2800+. For the **Gav02** dataset (586 purifications), our implementation took 317 CPU seconds for computing the probability for all pairs and 440 CPU seconds for computing the threshold graph and the delineation of protein complexes. For the larger **Gav06** dataset (2166 purifications), the running time is about three hours for computing all pair-wise inclusion probabilities. Decomposition of the SCC graph takes about four hours.

6.3 Results

We evaluated the delineation of sub-complex structure by computing the sensitivity and specificity over all decomposed subgraphs. For the evaluation against the two benchmarks (MIPS and Reguly), we excluded those purifications that have their baits not annotated to occur in a complex in the benchmark data set. The Gav02 set is rather small and comes with a manual annotation, allowing comparison between manual curation and automated methods. For computing the interaction probability, we must estimate three parameters ν, ϕ and ρ. For the **Gav02** data set, we refer to the parameters published in Gilchrist et al. [17]: $\nu = 0.346$,$\phi = 0.00107$ and $\rho = 0.00188$. For the **Gav06** data set, the parameters are $\nu = 0.40699$,$\phi = 0.00135$ and $\rho = 0.00389$ estimated with 2760 proteins (3579150 possible interaction pairs) by counting proteins occurred in all 2166 purifications. Although we applied a different model from Scholtens et al. [37] to estimate the experimental error, we agree with their recommendation to perform more repeated purifications using well-characterized proteins as baits to better estimate the false negative and false positive error rate [37]. We also investigated the distribution of all pair shortest path lengths in the SCC graph. After annotating SCCs with the corresponding MIPS annotation by majority voting, we found that purifications of the same MIPS complex are always adjacent nodes in the SCC graph (having the path length of one). Purifications of different MIPS complexes result in a longer path. See Figure 6.4.

Several proteins that occur in many purifications (> 10) can be considered systematic errors as they are typically abundant or display a general protein binding affinity. Purifications of these proteins are typically linked to many purifications. Our approach assigns these purifications to many subgraphs due to their asymmetric nature without creating large clusters that solely rely on these false positive identifications. However, for display purposes of the complete interaction graph, we recommend removing them as in Figure 6.5.

Figure 6.4: Histogram of the shortest distance among SCCs (**Gav02**). This figure shows histogram of the shortest distance among SCCs having MIPS annotation, computed for **Gav02**.



(a) Within complexes

(b) Between complexes

Given the directed representation, we can derive overlapping clusters from the decomposed path computed by depth-first search. As an example, we compare the overlapping paths with the manual curation (overlapping) on the two benchmarks. We compute the specificity and sensitivity and prediction error for all protein interaction pairs of a reduced set of proteins from the **Gav02** set which are also annotated in the benchmark, **Cellzome** and **Krause**. This results in 176 proteins for the MIPS benchmark and 136 for **Reguly**. We do not provide comparison for the **Gav06** data set due to lack of manual curation. In our solution, a cluster is a union of baits from a decomposed subgraph and all pairs of interaction in the cluster are considered. Therefore our resulting clusters overlap because proteins can be assigned to multiple clusters. Similar to ours, the **Cellzome** solution also assign proteins to several clusters. We selected two thresholds for comparison with manual curation at $\log(\tau) = -8$ and -5. The sensitivity and specificity is tabulated in Table 1 and prediction error is Table 2. We will refer to an (unordered) pair of baits from the same complex as a *true* pair, and to a pair of baits from the same cluster as a *predicted* pair. We will call a true predicted pair *true positive* (TP), a true pair which has not been predicted *false negative* (FN), a false pair predicted to be from a complex *false positive* (FP) and a false, but not predicted pair *true negative* (TN). The following quantities summarize the performance: *Sensitivity*, $sens = \frac{\#TP}{\#TP+\#FN}$ and *Specificity*, $spec = \frac{\#TP}{\#TP+\#FP}$. A perfect clustering method would have $sens = spec = 1$, which implies no errors, neither FP nor FN. The prediction error is the percentage of predicted interaction pairs that are different from the true assignment.

Table 6.1: This table shows different cluster solutions of the baits against the two benchmarks. SN: Sensitivity. SP: Specificity. For our model, the parameters are $\nu = 0.346$, $\phi = 0.00107$, $\rho = 0.00188$. We select two thresholds at $\log(\tau) = -8$ (sensitive setting) and $\log(\tau) = -5$ (specific setting).

	MIPS		Reguly	
Solution	SN	SP	SN	SP
Cellzome [15]	0.64	0.64	0.42	0.86
Krause [24]	0.37	0.85	0.20	0.91
Our model, sensitive setting	0.53	0.24	0.39	0.50
Our model, specific setting	0.40	0.40	0.23	0.75

Table 6.2: This table shows prediction error from different cluster solutions on the two benchmarks. Prediction error is the percentage of predicted interaction pairs that are different from the true assignment. For our model, the parameters are $\nu = 0.346$, $\phi = 0.00107$, $\rho = 0.00188$. We select two thresholds at $\log(\tau) = -8$ (sensitive setting) and $\log(\tau) = -5$ (specific setting).

	MIPS	Reguly
Solution	Err(%)	Err(%)
Cellzome [15]	1.1	2.7
Krause [24]	1.5	3.4
Our model, sensitive setting	3.4	4.0
Our model, specific setting	1.9	3.4

6.3.1 Examples

The SCC represents the interacting *core* of protein complexes and usually includes well-characterized proteins. For example, we group six purifications of Arp2/3 complex into the same SCC: Arc15, Arc18, Arc35, Arc40, Arp2 and Arp3.

The RNA polymerase complexes (RNAPI, RNAPII and RNAPIII) are a good example. We detected nine decomposed subgraphs to make up the RNA polymerase complexes with three distinct subgraphs corresponding to three distinct Pol complexes (shown in Figure 6.7 with different colors in the supplementary materials). More importantly, the purification of Rpc40 which creates an overlapping component between PolI and PolIII shows the expected connecting structure of these two complexes.

For illustration purposes, we show the SCCs containing the same bait multiple times due to repeated experiments as square boxes, for example in Figure 6.5 and 6.6, to distinguish them from SCCs consisting of several baits or a single bait.

In Figure 6.5, the SCC graph depicts purification pairs with inclusion probability higher than a threshold τ or its logarithm greater than $\log(\tau)$. Parameters are $\nu = 0.40699$, $\phi = 0.00135$, $\rho = 0.00389$ and $\log(\tau) = -8$. SCCs with in-degree larger than 8 are removed from the figure.

6.4 Conclusion

We have presented a model for mapping overlapping complexes to purification experiments based on subset inclusion. The model results in a directed graph that shows inclusion structure among the bait proteins in the whole experiment. The model is based on a probabilistic model of protein interaction presented in the previous chapter.

Overlapping protein complexes can be obtained from such a directed graph; however, the poor prediction accuracy shows it to be inadequate for this task. As a result, we conclude that our tool provides an overview of possible overlapping structures, yet this view is subject to significant error. Given the current experimental data, it is unlikely that a general computational method for finding the directed graph of overlapping structures is attainable. The number of directed graphs grows exponentially with the number of proteins. Therefore, a general model to describe such a structure will require too many parameters. The simplest algorithm to discover a part of this structure is to compute frequent itemsets from purification data, as explored in Chapters 3 and 4. By interpreting frequent itemsets as protein complexes, we can directly derive their overlap. However, as shown in previous chapters, the prediction accuracy of an overlapping model is still worse than assuming a partitioning model of protein interactions.

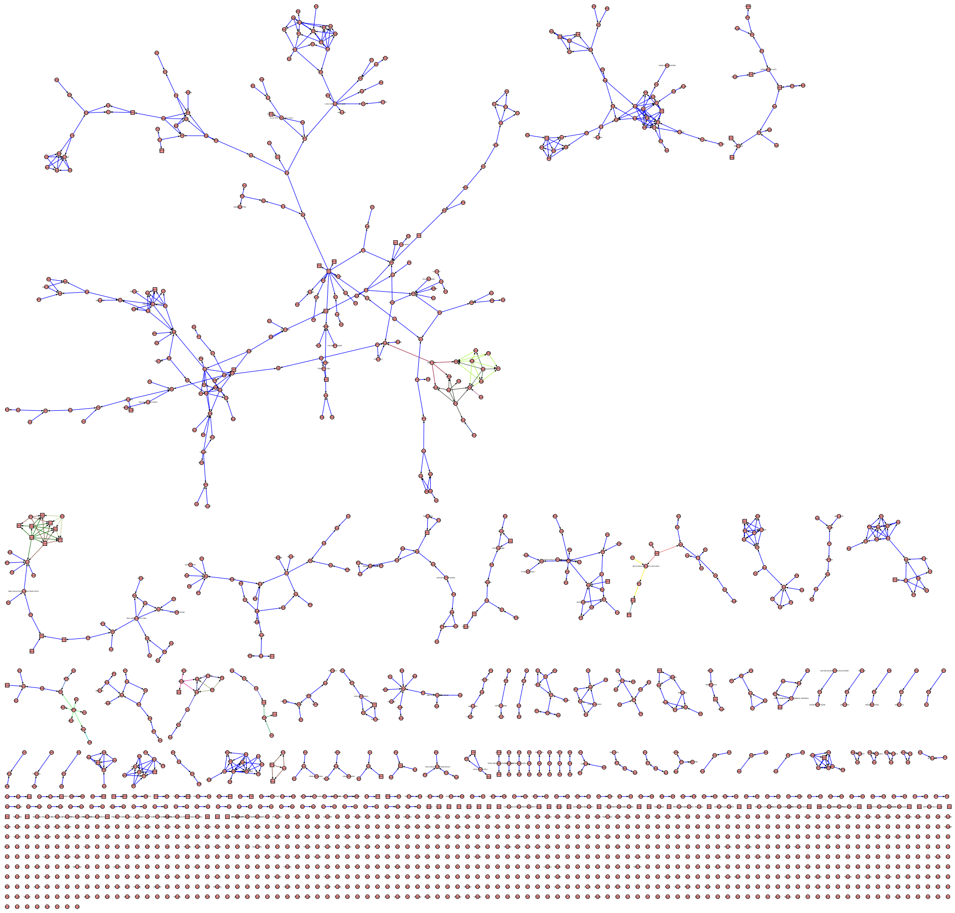
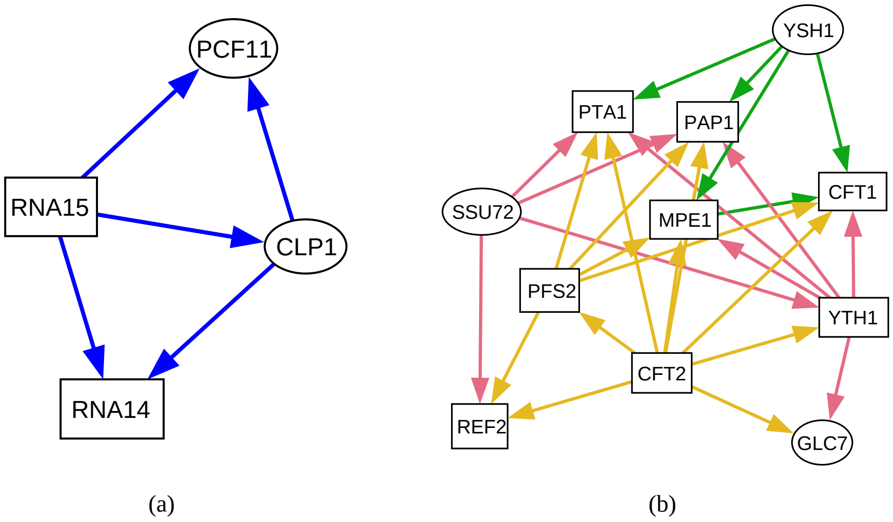
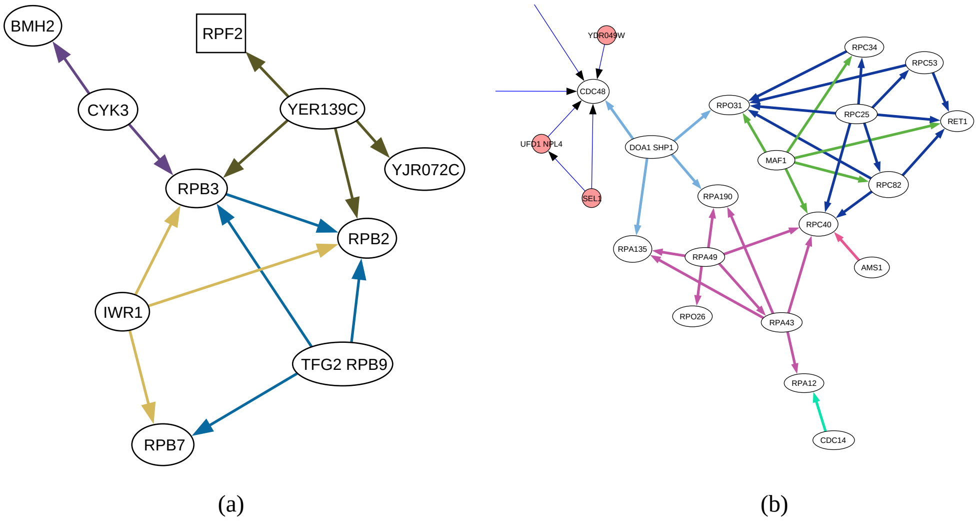


Figure 6.5: The SCC graph of **Gav06** data set. The SCC graph depicts purification pairs with inclusion probability higher than a threshold τ or its logarithm greater than $\log(\tau)$.



(a) (b)

Figure 6.6: Subcomplex structure of the PolyA complex.



(a) (b)

Figure 6.7: Subcomplex structure of the Rna polymerases I, II and III. (a) Rna polymerase II and (b) Rna polymerases I and III.