<table>
<thead>
<tr>
<th>Title</th>
<th>Identifying protein complexes from heterogeneous biological data (Main Article)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Author(s)</td>
<td>Wu, Min; Xie, Zhipeng; Li, Xiaoli; Kwoh, Chee Keong; Zheng, Jie</td>
</tr>
<tr>
<td>Date</td>
<td>2013</td>
</tr>
<tr>
<td>URL</td>
<td><a href="http://hdl.handle.net/10220/18179">http://hdl.handle.net/10220/18179</a></td>
</tr>
<tr>
<td>Rights</td>
<td>© 2013 Wiley Periodicals, Inc. This paper was published in Proteins: Structure, Function, and Bioinformatics and is made available as an electronic reprint (preprint) with permission of Wiley Periodicals, Inc. The paper can be found at the following official DOI: <a href="http://dx.doi.org/10.1002/prot.24365">http://dx.doi.org/10.1002/prot.24365</a>. One print or electronic copy may be made for personal use only. Systematic or multiple reproduction, distribution to multiple locations via electronic or other means, duplication of any material in this paper for a fee or for commercial purposes, or modification of the content of the paper is prohibited and is subject to penalties under law.</td>
</tr>
</tbody>
</table>
Identifying protein complexes from heterogeneous biological data

Min Wu,1,2* Zhipeng Xie,3 Xiaoli Li,1,2 Chee-Keong Kwoh,1 and Jie Zheng1,4*

1 School of Computer Engineering, Nanyang Technological University, Singapore
2 Institute for Infocomm Research, A*STAR, 1 Fusionopolis Way, Singapore
3 School of Computer Science, Fudan University, China
4 Genome Institute of Singapore, A*STAR, Biopolis, Singapore

ABSTRACT

With the increasing availability of diverse biological information for proteins, integration of heterogeneous data becomes more useful for many problems in proteomics, such as annotating protein functions, predicting novel protein–protein interactions and so on. In this paper, we present an integrative approach called InteHC (Integrative Hierarchical Clustering) to identify protein complexes from multiple data sources. Although integrating multiple sources could effectively improve the coverage of current insufficient protein interactome (the false negative issue), it could also introduce potential false-positive interactions that could hurt the performance of protein complex prediction. Our proposed InteHC method can effectively address these issues to facilitate accurate protein complex prediction and it is summarized into the following three steps. First, for each individual source/feature, InteHC computes the matrices to store the affinity scores between a protein pair that indicate their propensity to interact or co-complex relationship. Second, InteHC computes a final score matrix, which is the weighted sum of affinity scores from individual sources. In particular, the weights indicating the reliability of individual sources are learned from a supervised model (i.e., a linear ranking SVM). Finally, a hierarchical clustering algorithm is performed on the final score matrix to generate clusters as predicted protein complexes. In our experiments, we compared the results collected by our hierarchical clustering on each individual feature with those predicted by InteHC on the combined matrix. We observed that integration of heterogeneous data significantly benefits the identification of protein complexes. Moreover, a comprehensive comparison demonstrates that InteHC performs much better than 14 state-of-the-art approaches. All the experimental data and results can be downloaded from http://www.ntu.edu.sg/home/zhengjie/data/InteHC.

© 2013 Wiley Periodicals, Inc.

Key words: protein complexes; ranking SVM; data integration; hierarchical clustering; protein; protein interactions.

INTRODUCTION

Protein complexes are of great importance for understanding cellular organization and functions. For example, RNA-induced silencing complex (RISC complex)1 plays a fundamental role in gene regulation by micro RNAs (miRNA) and in defence against viral infections by incorporating one strand of a small interfering RNA (siRNA) or miRNA. Another example is RNA polymerase II complex,2 which transcribes genetic information into messages for ribosomes to produce proteins. Although protein complexes are crucial for many cellular processes in cell and molecular biology, they are still largely obtained through small-scale experimental techniques, which are time-consuming and tedious. In addition, many important protein complexes have not been detected by current wet-lab experiments. Therefore, computational methods for predicting protein complexes are highly desired.

Additional Supporting Information may be found in the online version of this article.
Grant sponsor: Ministry of Education, Singapore (Tier 1 AcRF Grant RG32/11, Tier 2 AcRF Grant); Grant numbers: M4010977.020, MOE2008-T2-1-074.
This work was done when Min Wu was a research fellow in School of Computer Engineering, Nanyang Technological University.
*Correspondence to: Min Wu, Institute for Infocomm Research, A*STAR, 1 Fusionopolis Way, Singapore. E-mail: wumin@i2r.a-star.edu.sg or Jie Zheng, School of Computer Engineering, Nanyang Technological University, Singapore. E-mail: zhengjie@ntu.edu.sg
Received 3 January 2013; Revised 3 June 2013; Accepted 17 June 2013
Published online 15 July 2013 in Wiley Online Library (wileyonlinelibrary.com).
DOI: 10.1002/prot.24365
As a large amount of binary protein–protein interaction (PPI) data are now available, it becomes more prevalent to detect protein complexes in PPI networks where nodes are proteins and edges are protein interactions. There is an observation that protein complexes generally correspond to dense subgraphs or cliques in PPI networks. On the basis of this observation, a number of computational methods are proposed for identifying protein complexes, such as MCODE, MCL, DPClus, IPCA, COACH, CMC, and so on.

As two large-scale tandem affinity purification with mass spectrometry (TAP-MS) data were released in 2006, another batch of algorithms were proposed to detect protein complexes from these TAP-MS data. These methods can be classified into the following two categories. The methods in the first category compute the affinity scores between proteins (e.g., the probability of two proteins to be co-complex members) and convert the TAP-MS data to a PPI network based on the calculated affinity scores. Traditional graph clustering methods are then applied to detect protein complexes in the obtained PPI networks. The second category is to model TAP-MS data as bipartite graphs and then detect dense bipartite subgraphs as protein complexes.

Given the fact that PPI data are inherently noisy (e.g., false positives and false negatives), several methods had integrated other non-PPI sources to assess the reliability of PPI data for more accurate detection of protein complexes. For example, the above mentioned PCP and CMC algorithms utilized the topological weights of interactions as their reliability for detecting protein complexes. DECAFF and STM first exploited functional information of proteins (e.g., Gene Ontology annotations) to assess the reliability of PPI data and then detected protein complexes from the refined PPI data. Meanwhile, MATISSE integrated gene expression data with PPI data to increase the confidence of interactions for the same purpose. More details for the above computational prediction of protein complexes can be found in this survey. However, each of the above integration methods generally combines a single data source (e.g., gene expression profiles or functional annotations) with PPI data to refine the quality of the PPI data. Currently, a highly diverse collection of sources for proteins is available, e.g., binary PPI data, gene expression profiles, functional Gene Ontology (GO) terms, TAP-MS data, and so on. Therefore, it is highly motivated to integrate these multiple heterogeneous sources for predicting protein complexes.

Although there are many studies on integrating multiple data sources to predict PPI integrative methods for predicting complexes are still rather limited. To the best of our knowledge, there are two such existing works as follows. Xia et al. integrated multiple data sources to create a database named IntNetDB for more confident PPI data and then predicted protein complexes by using MCODE algorithm on IntNetDB. However, they did not predict PPI or protein complexes in the model organism yeast. Another method named CMBI integrated PPI data, gene expression profiles and gene essentiality data to predict protein complexes.

To address the above issue, we proposed an approach called InteHC (Integrative Hierarchical Clustering) to predict protein complexes by integrating multiple data sources. The overall framework of InteHC is shown in Figure 1. First, we use individual sources/features to compute the score matrices that store various protein affinities between all the pair-wise proteins. Second, we construct a final score matrix, which is the weighted sum of the matrices from individual features. In particular, these weights, indicating the reliability of each individual data sources, are learned by a supervised model (i.e., a linear ranking SVM) on a known set of positive and negative protein–protein interactions. This final score matrix not only integrates all the individual sources to address the false-negative issue but also takes the reliability of each individual sources into consideration to address the potential false positive issue. Finally, a hierarchical clustering algorithm is then applied to identify protein complexes from the final score matrix. We have conducted comprehensive experiments to evaluate our predicted complexes based on several metrics, such as recall, coverage rate, and accuracy. The comparison with other methods shows that the integration of heterogeneous sources.

![Figure 1](image-url)

The overall framework of our proposed InteHC method.
significantly improves the coverage and accuracy for protein complex prediction.

**MATERIALS AND METHODS**

In this section, we first introduce various data sources used in this study. And then we provide detailed description of our proposed InteHC method as well as the evaluation metrics for predicted protein complexes.

**Data source**

Our InteHC integrated four data sources for identifying protein complexes, namely, PPI data, Gene Ontology data, gene expression profiles, and AP-MS data. DIP database for PPI data was downloaded from Ref. 29. GO data were downloaded from Ref. 30, while gene expression data were downloaded from Ref. 31. AP-MS data were downloaded from Refs. 11 and 12. In summary, DIP data consist of 4930 proteins and 17,262 interactions. Gene Ontology data consist of 2996 bait proteins and 5405 prey proteins. In addition, 13,424 positive protein–protein interactions for training the linear ranking SVM were downloaded from Refs. 11 and 12. In summary, individual data sources for protein affinities were downloaded from Ref. 31. AP-MS data consist of 6498 purifications involving 2996 bait proteins and 5405 prey proteins.

**Individual data sources for protein affinities**

Estimation of the affinity scores between proteins is a crucial step for protein complex prediction. In the following, we introduce various protein affinities from heterogeneous data sources.

\[
S_{FS}(p, q) = \frac{2 \times |N_p \cap N_q|}{|N_p - N_q| + 2 \times |N_p \cap N_q| + 1}
\]

where \( N_p \) includes direct neighbors of the protein \( p \) as well as \( p \) itself and \( N_q \cap N_q \) includes the common neighbors between \( p \) and \( q \). Thereafter, FS is short for FSweight in PPI networks.

**Gene expression profiles**

Given a protein pair \((p, q)\), the proteins’ propensity to interact can be measured by using the Pearson Correlation Coefficient between their encoded genes’ expression profiles \( G_p \) and \( G_q \) as follows,

\[
S_{GE}(p, q) = \frac{\sqrt{\sum_{i=1}^{n} (p_i - \overline{p})(q_i - \overline{q})}}{\sqrt{\sum_{i=1}^{n} (p_i - \overline{p})^2} \sqrt{\sum_{i=1}^{n} (q_i - \overline{q})^2}}
\]

where \( n \) is the number of time points for the expression profiles and \( p_i \) is the \( i \)th expression value of protein \( p \)’s expression profiles. \( \overline{p} \) is the average expression value across different time points.

**Protein function profiles**

Given two proteins \( p \) and \( q \) annotated by GO terms \( \{g_{i1}, \ldots, g_{im}\} \) and \( \{g_{i1}, \ldots, g_{jn}\} \) respectively, their functional similarity as defined in Ref. 38, \( S_{GO}(p, q) \), is calculated as follows.

\[
S_{GO}(p, q) = \frac{\sum_{i=1}^{m} \max_{1 \leq j \leq n} \text{sim}(g_{i1}, g_{j1}) + \sum_{i=1}^{n} \max_{1 \leq j \leq m} \text{sim}(g_{i1}, g_{j1})}{m + n}
\]

where \( \text{sim}(g_{i1}, g_{j1}) \) is the semantic similarity between GO terms \( g_{i1} \) and \( g_{j1} \) defined in Ref. 37. As we know, GO has three sub-ontologies, namely, biological process (BP), molecular function (MF), and cellular component (CC). BP sub-ontology contains the most number of GO terms and is the most informative, and thus we utilize GO terms in BP sub-ontology to calculate the functional similarities between the pair-wise proteins in this article.

**TAP-MS data profiles**

With two large-scale TAP-MS data released, additional computational methods have been proposed to assess the protein affinities based on the purification.
Integration of evidences

In the aforementioned subsection, we have introduced how to calculate the protein affinity scores based on four heterogeneous data sources. Now we are ready to integrate four score matrices together. Generally, a typical approach to integrating these affinity scores into a final score matrix is the linear weighted sum. This leads us to address the next question, “How to assign appropriate weights to the respective data sources given that they have different reliability?”

We first collected a set of known positive and negative protein–protein interactions.

Classification models can be learned from these known positive and negative PPIs to predict novel PPIs and assess protein affinities. In particular, each interaction can be represented as a feature vector $x_i = (s_{i1}, s_{i2}, ..., s_{ik})$ where $s_{il}$ ($1 \leq l \leq k$, where $k$ is the number of data sources and $k=4$ in this work) is the affinity score from the $l$th data source (i.e., $S_{FS}$, $S_{GC}$, $S_{GO}$ and $S_{TAP}$). In addition, a linear ranking support vector machine (SVM) would be used to learn the weight vector $w$. The linear ranking SVM in Eq. (4) will try to rank a positive interaction $x_i$ before a negative interaction $x_j$ by maximizing the area under ROC curve (AUC),

$$w^T \sum_{i,j} I(y_i = 1 \land y_j = -1) x_i - \sum_{i,j} I(y_i = -1 \land y_j = 1) x_j$$

where $y_i$ is the label of the interaction $x_i$ (i.e., $y_i = 1$ means that $x_i$ is a positive interaction and $-1$ otherwise), $I(y_i \neq y_j)$ is the indicator function and $c$ is a parameter.

Now let us focus on the loss function in Equation (4). Given that two interactions $x_i$ and $x_j$ have different labels, we assume that $x_i$ is positive (i.e., $y_i = 1$) and $x_j$ is negative (i.e., $y_j = -1$) without loss of generality. In this case, it is obvious that ranking $x_j$ before $x_i$ (i.e., $w \cdot x_j > w \cdot x_i$) will have a larger loss than ranking $x_i$ before $x_j$. In other words, the linear ranking SVM in Eq. (4) will try to rank a positive interaction $x_i$ before a negative interaction $x_j$ to reduce the loss and maximize the AUC. With the weight vector $w = (w_1, w_2, ..., w_k)$ learned from the known positive and negative PPIs, we can calculate the final affinity score for a protein pair $(p, q)$, $S(p, q)$ in Equation (5).

$$S(p, q) = \sum_{i=1}^{4} w_i \cdot \mathcal{S}_E(p, q), \text{ where } E_i \in \{FS, GE, GO, TAP\}$$

Obviously, protein pairs with higher final affinity scores are more likely to be true protein–protein interactions. In addition, the affinity scores in the matrix $S$ in Eq. (5) are able to estimate the quality of protein interactions more accurately than those from individual data sources, because they combine multiple data sources and also take the reliability of each source into account.

Hierarchical clustering for protein complexes

The hierarchical clustering algorithm is applied to detect protein complexes on the final score matrix in Eq. (5). First, it considers all singleton proteins as initial clusters. Second, it iteratively merges two clusters with the highest similarity in each iteration. The algorithm terminates when the quality of the detected complexes in the merging process has become maximal. The detailed procedure for the hierarchical clustering is illustrated in Algorithm 1. Given two clusters $c_i$ and $c_j$, their similarity, $\text{sim}(c_i, c_j)$, is the average affinity score of all protein pairs between these two clusters as shown in Eq. (6). Next, we will introduce the quality function in Line 2.

$$\text{sim}(c_i, c_j) = \frac{1}{|c_i| \cdot |c_j|} \sum_{p \in c_i, q \in c_j} S(p, q)$$

Given a cluster $c_i$, we define its confidence score as the average affinity score of all possible protein pairs within it as shown in Eq. (7). Given a clustering
C = \{c_1, \ldots, c_n\}, we proposed the following two functions to measure the quality of this clustering C based on the affinity scores for protein pairs within these clusters. The first quality function $Q_1(C)$ in Eq. (8) is the average confidence score for all the clusters while the second quality function $Q_2(C)$ in Eq. (9) is the average affinity score for all the protein pairs within the clusters.

\[
Conf(c_i) = \frac{\sum_{p,q \in C} S(p,q)}{|C| \times (|C|-1)}
\]

\[
Q_1(C) = \frac{\sum_{i=1}^{n} Conf(c_i)}{n}
\]

\[
Q_2(C) = \frac{\sum_{i=1}^{n} \sum_{p,q \in C} S(p,q)}{\sum_{i=1}^{n} |C| \times (|C|-1)}
\]

In general, $Q_2(C)$ will be dominated by those large clusters as they contain many more protein pairs than those small clusters. Therefore, we can upgrade $Q_2(C)$ to the third quality function $Q_3(C)$ in Eq. (10) by dividing a factor (i.e., the square root of the cluster size, $\sqrt{|C|}$) in both denominator and numerator to eliminate the bias from those large clusters (this factor has also been utilized to eliminate the bias of cluster size in a previous study\(^{41}\)). As such, we have three functions to evaluate the quality of a set of clusters.

\[
Q_3(C) = \frac{\sum_{i=1}^{n} \frac{1}{\sqrt{|C|}} \sum_{p,q \in C} S(p,q)}{\sum_{i=1}^{n} \frac{1}{\sqrt{|C|}} \times (|C|-1)}
\]

In fact, the hierarchical clustering here does not stop, i.e., it runs from the start with all individual proteins as clusters to the end with all the proteins as a whole cluster. During this process, the values for a given quality function can be monitored at each iteration. Finally, we output the clusters in a certain iteration where the given quality function achieves the highest value.

**Evaluation metrics**

We will utilize the sensitivity ($Sn$, i.e., the coverage rate) and positive predictive value (PPV) to evaluate the quality of predicted protein complexes. In particular, the Sensitivity and revised PPV between a benchmark complex $b_i$ and a predicted complex $c_j$ are defined in Eq. (11).\(^9\)\(^4\)\(^2\). Accuracy is the geometric mean of sensitivity and PPV.

\[
Sn = \frac{\max_{i,j} T_{ij}}{|[b_i]|}, \quad PPV = \frac{\max_{i,j} T_{ij}}{\sum_{j} [\bigcup_{i} (b_i \cap c_j)]}
\]

and $\text{Accuracy} = \sqrt{Sn \times PPV}$

where $T_{ij}$ is the number of proteins shared by $b_i$ and $c_j$, i.e., $|b_i \cap c_j|$. We also use another metric, Recall, to evaluate the predicted complexes. Given a predicted complex $c_j$ and a real complex $b$, their neighborhood affinity score, $NA(c, b)$ in Eq. (12), can be used to determine how well they match with each other. We consider them to be matching if $NA(c, b) \geq \omega$ ($\omega$ is a parameter which is normally set as 0.2 in previous studies\(^6\)\(^9\)\(^8\)\(^19\) and we also set it as 0.2 in our experiments). Recall is defined in Eq. (13) as the fraction of benchmark complexes that are matched by at least one predicted complex. In Eq. (13), $B$ is the set of benchmark complexes. Here, we used CYC2008 catalog\(^43\) as benchmark protein complexes for calculating both Accuracy and Recall.

\[
NA(c, b) = \frac{|c \cap b|^2}{|c| \times |b|},
\]

\[
\text{Recall} = \frac{|\{b \in B, \exists c \in C, NA(c,b) \geq \omega\}|}{|B|}
\]

With the Accuracy and Recall defined earlier, their average and harmonic average (HAverage) in Eq. (14) are utilized to evaluate the overall quality of predicted complexes. Now we have a comprehensive set of evaluation metrics for the quality of predicted complexes. For example, the sensitivity in Eq. (11) shows how many proteins in benchmark complexes are covered by the predicted complexes and thus measures protein-level coverage of predicted complexes. Meanwhile, Recall measures complex-level coverage of predicted complexes.

\[
\text{Average} = \frac{\text{Accuracy} + \text{Recall}}{2},
\]

\[
\text{HAverage} = \frac{2 \times \text{Accuracy} \times \text{Recall}}{\text{Accuracy} + \text{Recall}}
\]

**RESULTS**

In this section, we demonstrate a comprehensive comparison between InteHC and various state-of-the-art approaches for predicting protein complexes. Moreover, we also introduce the experimental setting for a common parameter involved in ranking SVM as well as our choice for quality functions.

**Linear ranking SVM**

SVM\(^{perf}\)\(^3\)\(^9\) is utilized in this article to solve the optimization problem in Eq. (4). The parameter $c$ in Eq. (4) is the trade-off between training error and margin. In general, the classifier (i.e., SVM) becomes more and more over-fitting and the generalization error increases as $c$ becomes larger. Therefore, an appropriate $c$ should not be too large while maintaining a good ROC area (AUC).

To select an appropriate value for $c$, we performed five-folder cross-validation on 26,848 positive and negative protein interactions. Figure 2 shows the average
The AUC of our trained SVM using different values for $c$. We can observe that the AUC becomes more and more stable as $c$ increases. We finally set $c$ as 15 in our experiments as it satisfies the above two requirements—$c$ is relatively small and its AUC (0.879) is also good. After learning, the weights for four data sources (FS, GO, GE, and TAP) in Eq. (5) are 8.49, 2.25, 0.84, and 2.89 respectively. On the basis of these learned weights for different biological sources, we are able to compute the final score matrix using Eq. (5).

**Quality functions for a clustering**

Once we get the final score matrix, the hierarchical clustering will run on it and output the predicted clusters as protein complexes when a specific quality function for the clustering is maximized. As such, the hierarchical clustering may predict different sets of protein complexes when we utilize different quality functions. Next, we will show the Recall and Accuracy of protein complexes predicted by our proposed quality functions.

Note that the aforementioned three quality functions for clusters are based solely on the properties of these clusters themselves (e.g., the affinity scores of protein pairs within clusters and the cluster size). Meanwhile, we can have another quality function for clusters, which is the sum of Recall and Accuracy collected by mapping them to known protein complexes (e.g., CYC2008 complexes). This quality function will output the optimal set of complexes which achieves higher sum of Recall and Accuracy. We thus denote it as $Q^*$ and utilize it to verify the goodness of the quality functions $Q_1$, $Q_2$ and $Q_3$. As shown in Figure 3, it is obvious that the clusters generated by the quality function $Q_3$ achieve higher Recall and Accuracy than those by $Q_1$ and $Q_2$. $Q_3$ is also much closer to the optimal solution $Q^*$ than $Q_1$ and $Q_2$. This indicates that the quality of a set of clusters is well evaluated by considering the cluster size in a reasonable manner in our Equation (10). Thereafter, we will only show the results generated by $Q_3$ for comparisons.

**The benefit of data integration**

To demonstrate the power of data integration for identifying protein complexes, we will show the performance of hierarchical clustering algorithm using each individual data sources, as well as the combined set of all features in Figure 4.
For the results using single data sources, we can observe that PPI network with FSweight (FS) covers the most benchmark complexes and achieves the highest Recall (0.708), demonstrating that traditional PPI networks are indeed informative and reliable for protein complex detection. GO can also achieve a good Recall (0.679). Meanwhile, complexes generated from TAP data have the highest sensitivity (0.679) and accuracy (0.752), indicating that TAP data are also a high quality source for predicting protein complexes. However, it is obvious that FS has a low Accuracy while TAP has a relatively low Recall as shown in Figure 4, indicating that using individual sources alone will not produce very good results.

In Figure 4, it is demonstrated that data integration achieves a higher overall performance than using individual data sources, illustrating that our proposed InteHC can effectively integrate multiple sources for protein complex prediction. As mentioned above, FS has a high Recall and a low Accuracy while TAP has a low Recall and a high Accuracy. Here, InteHC manages to overcome the limitations of individual sources. In particular, InteHC has a comparable Recall with FS and has a significantly higher Accuracy than FS (0.769 vs. 0.627). Similarly, it has much higher Recall than TAP (0.711 vs. 0.625).

For completeness of our evaluation, we have 15 combinations with four data sources available (FS, GE, GO, and TAP). The performance of hierarchical clustering algorithm using all these 15 combinations is shown in Supporting Information Figure S1. For example, the combinations FS+GO and FS+GO+TAP also achieve high overall performance. This indicates once again that data integration indeed benefits the accurate prediction of protein complexes.

**Comparison with methods on PPI networks**

In this subsection, we compare InteHC with eight existing state-of-the-art approaches that detect protein complexes from PPI networks (e.g., the DIP data in our experiments), which include MCODE, MCL, DPClus, IPCA, DECAFF, COACH, HC-PIN, HC-wPIN, and ProRank. HC-PIN and HC-wPIN represent the complexes predicted from unweighted and weighted DIP data, respectively. Here, FSweight is utilized to compute the weights for protein interactions in HC-wPIN. We can observe that HC-PIN predicts a small number of complexes and achieves low Recall and Accuracy on both unweighted and weighted DIP data (similarly, a recent method ProRank also generates a small number of complexes). However, we must point out that HC-PIN has many good merits. For example, it has a low computational complexity. In addition, HC-PIN has a high fraction of predicted complexes (e.g., 86 of 147 complexes predicted by HC-wPIN) that can match with known complexes. ProRank is also similar to HC-PIN with 58 of 110 complexes matching with known complexes.

**Comparison with methods on TAP-MS data**

Here, we compare InteHC with five existing methods proposed for TAP-MS data, including C2S, CACHET, BT, Pu, and Hart. Similar to InteHC-FS in Table I, InteHC-TAP in Table II represents the set of complexes that are predicted by InteHC.

### Table I
**Comparisons between InteHC and Various Existing Methods on PPI Data**

<table>
<thead>
<tr>
<th>Methods</th>
<th>No. of complexes</th>
<th>Recall</th>
<th>Sn</th>
<th>PPV</th>
<th>Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>MCODE</td>
<td>182</td>
<td>0.240</td>
<td>0.403</td>
<td>0.624</td>
<td>0.501</td>
</tr>
<tr>
<td>MCL</td>
<td>1116</td>
<td>0.586</td>
<td>0.541</td>
<td>0.763</td>
<td>0.643</td>
</tr>
<tr>
<td>DPClus</td>
<td>1140</td>
<td>0.642</td>
<td>0.511</td>
<td>0.836</td>
<td>0.654</td>
</tr>
<tr>
<td>IPCA</td>
<td>1242</td>
<td>0.539</td>
<td>0.501</td>
<td>0.766</td>
<td>0.620</td>
</tr>
<tr>
<td>DECAFF</td>
<td>2190</td>
<td>0.525</td>
<td>0.454</td>
<td>0.795</td>
<td>0.601</td>
</tr>
<tr>
<td>CDACH</td>
<td>746</td>
<td>0.527</td>
<td>0.545</td>
<td>0.698</td>
<td>0.617</td>
</tr>
<tr>
<td>HC-PIN</td>
<td>99</td>
<td>0.169</td>
<td>0.573</td>
<td>0.347</td>
<td>0.446</td>
</tr>
<tr>
<td>HC-wPIN</td>
<td>147</td>
<td>0.289</td>
<td>0.516</td>
<td>0.673</td>
<td>0.589</td>
</tr>
<tr>
<td>ProRank</td>
<td>110</td>
<td>0.162</td>
<td>0.199</td>
<td>0.897</td>
<td>0.422</td>
</tr>
<tr>
<td>InteHC-FS</td>
<td>1307</td>
<td>0.708</td>
<td>0.434</td>
<td>0.906</td>
<td>0.627</td>
</tr>
<tr>
<td>InteHC</td>
<td>860</td>
<td>0.711</td>
<td>0.701</td>
<td>0.845</td>
<td>0.769</td>
</tr>
</tbody>
</table>

### Table II
**Comparisons between InteHC and Various Existing Methods on TAP Data**

<table>
<thead>
<tr>
<th>Methods</th>
<th>No. of complexes</th>
<th>Recall</th>
<th>Sn</th>
<th>PPV</th>
<th>Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>C2S</td>
<td>1035</td>
<td>0.630</td>
<td>0.676</td>
<td>0.847</td>
<td>0.757</td>
</tr>
<tr>
<td>CACHET</td>
<td>449</td>
<td>0.515</td>
<td>0.492</td>
<td>0.901</td>
<td>0.665</td>
</tr>
<tr>
<td>BT</td>
<td>409</td>
<td>0.598</td>
<td>0.629</td>
<td>0.848</td>
<td>0.730</td>
</tr>
<tr>
<td>Pu</td>
<td>400</td>
<td>0.591</td>
<td>0.691</td>
<td>0.789</td>
<td>0.738</td>
</tr>
<tr>
<td>Hart</td>
<td>390</td>
<td>0.593</td>
<td>0.610</td>
<td>0.863</td>
<td>0.725</td>
</tr>
<tr>
<td>InteHC-TAP</td>
<td>974</td>
<td>0.622</td>
<td>0.679</td>
<td>0.833</td>
<td>0.752</td>
</tr>
<tr>
<td>InteHC</td>
<td>860</td>
<td>0.711</td>
<td>0.701</td>
<td>0.845</td>
<td>0.769</td>
</tr>
</tbody>
</table>
of complexes predicted by our hierarchical algorithm on TAP data.

As shown in Table II, InteHC achieves the highest Recall, sensitivity, and Accuracy. For example, InteHC has a Recall 71.1%, which is 8.1% higher than the second best method (C2S, 63.0%). Also, InteHC-TAP and C2S achieve very similar results. In fact, InteHC-TAP and C2S are regarded as operating on the same score matrix and have different forms of stop criteria for the hierarchical clustering. Meanwhile, C2S18 was demonstrated to have an effective stop criteria for the hierarchical clustering (however, it cannot be used in the framework of our InteHC) and outperformed existing methods including BT,13 Pu,16 and Hart.15 Therefore, we can confidently claim that our quality function in Eq. (10) also provides us with an effective stop condition for the hierarchical clustering.

Comparison with other integrative methods

Xia et al.26 constructed a database named IntNetDB for PPI data predicted by integrating multiple data sources. They then predicted protein complexes by using MCODE algorithm on IntNetDB. However, they did not predict PPI or protein complexes for the model organism yeast. Therefore, we are not able to compare MCODE on IntNetDB with InteHC.

Another method named CMBI28 was recently proposed to detect protein complexes by integrating multiple biological resources including PPI data, gene expression profiles and essential protein information. Essential proteins are first selected as seeds. Proteins, which have high topological similarity and gene expression correlation with those seeds, will be included to form protein complexes. In this subsection, we will briefly compare the results between InteHC and CMBI.

Using the data described in Ref. 28, CMBI predicted 760 protein complexes. 160 of 408 benchmark complexes will be covered, that is, CMBI has a Recall 0.392. The sensitivity, PPV, and Accuracy of CMBI are 0.508, 0.482, and 0.495, respectively. Using the same PPI and gene expression data, InteHC predicted 1469 protein complexes and achieves a Recall 0.720, sensitivity 0.464, PPV 0.900, and Accuracy 0.646. Thus, InteHC performs significantly better than CMBI in terms of these measures. As shown in Supp. Info. Table S1, different data sources have different importance for estimating protein affinities and predicting protein complexes. For example, the PPI data (FS scores) have the highest importance based on the leave-one-out AUC while gene expression profiles have the lowest. Correspondingly, our supervised model [i.e., the ranking SVM in Eq. (4)] assigns each data source with different weights, e.g., FS scores have the highest weight 8.49 while the gene expression profiles have the lowest 0.84.47 However, CMBI treats equally the topological weights (ECC, edge clustering coefficient) and gene expression correlations and thus overrates the importance of the gene expression profiles. This may be one of the reasons why InteHC performs better than CMBI.

Protein complexes more accurately detected by InteHC

In previous sections, we demonstrated that InteHC outperformed 14 methods in terms of various evaluation measures (e.g., Recall and Accuracy). Next, we introduce two example protein complexes that are more accurately detected by InteHC.

Figure 5
Two examples of known protein complexes.
several aspects of mRNA metabolism, including repression and activation of mRNA initiation, control of mRNA elongation, deadenylation, and degradation. The complex (ID: 17) predicted by InteHC also has nine proteins and it can exactly match with the CCR4-NOT complex, that is, covering all the nine proteins in the CCR4-NOT complex. In Figure 5(B), the known transcription factor TFIID complex 49 in CYC2008 catalogue contains 15 proteins, while the complex (ID: 186) predicted by InteHC has 12 proteins. All these 12 proteins (black circles) are involved in the benchmark complex while three remaining proteins (white circles) are not covered by the complex 186. More examples can be found in the Supporting Information.

From the set of complexes predicted by a specific approach (e.g., InteHC), we may pick up a complex which has the highest similarity score [i.e., NA score in Eq. (12)] to a given benchmark complex and denote it as the mapped complex for this benchmark complex. For instance, the complex 186 predicted by InteHC is the mapped complex for the TFIID complex. Table III shows the mapped complexes for CCR4-NOT complex and TFIID complex predicted by various methods. Take the TFIID complex for example, its mapped complex predicted by InteHC (i.e., the complex 186) covers 12 of its 15 proteins and has the highest NA score 0.8, while its mapped complexes predicted by CACHET, COACH, DPClus, and IPCA have NA scores 0.738, 0.533, 0.491, and 0.471, respectively. Meanwhile, the complex 17 predicted by InteHC is exactly the benchmark CCR4-NOT complex and has the optimal NA score (i.e., 1.0). Two example complexes more accurately detected by InteHC show that a proper integration of multiple data sources indeed improves the computational prediction of protein complexes.

**CONCLUSIONS AND DISCUSSIONS**

In this article, we have proposed an integrative approach (InteHC, Integrative Hierarchical Clustering) for identifying protein complexes from heterogeneous sources, including PPI data, gene expression profiles, GO terms, and TAP-MS data. For each individual source/feature, we calculated the affinities between proteins to show their propensity to interact. Subsequently, a supervised model (i.e., the ranking SVM) was utilized to learn the affinities between proteins to show their propensity to interact. Subsequently, a supervised model (i.e., the ranking SVM) was utilized to learn the weight for each feature. The weighted sum of affinity scores from individual features resulted in a final score matrix. A hierarchical clustering algorithm on this final score matrix will generate a set of non-overlapping clusters as predicted protein complexes. In particular, we proposed a novel quality function [i.e., Q3 in Eq. (10)] for capturing the high-quality protein complexes. Experimental comparisons have shown that InteHC performs much better than 14 existing approaches in terms of several evaluation metrics, e.g., InteHC significantly improves the coverage for known protein complexes in both a protein-level (Recall) and complex-level (sensitivity). In addition, our InteHC is a flexible and generic framework to integrate multiple data sources for predicting protein complexes. It allows us to include new data sources by simple matrix operations to achieve even better results.
As discussed in our previous study,\textsuperscript{18} we plan to design new data-integration mechanisms for identifying protein complexes. For this purpose, we already managed to integrate multiple TAP-MS datasets in Ref. \textsuperscript{18} and various heterogeneous datasets in this article. These techniques are characterized as raw dataset-level integration. In future work, we will focus on another kind of integration—result-level integration. A specific method for detecting protein complexes working on a specific dataset will predict a set of protein complexes. We can thus collect many sets of complexes by applying various methods on multiple datasets. We may expect to achieve higher prediction accuracy by analyzing and processing those integrated complexes. Furthermore, we plan to set up a platform or web-server to demonstrate all the above integration techniques for identifying protein complexes.

\section*{REFERENCES}