Clustering Coefficients in Protein Interaction Hypernetworks

Suzanne Renick Gallagher
and Debra S. Goldberg
University of Colorado, Boulder
Department of Computer Science
Email: {suzanneg,debra}@colorado.edu

***Abstract*—Protein interaction data is usually modeled using graphs where nodes represent proteins and there is an edge between two proteins if they have been shown to interact in some study. However, this model is insufficient for some types of data, such as affinity purification data, which captures the interaction between many proteins rather than just two. To model this data, an extension of graphs known as hypergraphs has been proposed. However, due to the relative newness of the study of large complex hypergraphs, many of the statistics we have used to study protein interaction graphs have not been well defined.**

In this paper, we look at clustering coefficient, one of the statistics commonly used to study protein interaction networks. We examine some of the previous suggestions on how to extend this statistic to hypernetworks and look at the physical meaning of these in terms of protein interactions. We evaluated these various definitions to see how well they help to predict complexes.

I. Introduction

Protein interaction data is usually modeled using graphs where nodes represent proteins and there is an edge between two proteins if they have been shown to interact in some study. However, this model is insufficient for some types of interaction data, such as affinity purification (AP) data. In an affinity purification experiment, a particular protein, the "bait," is tagged in such a way that it will make it easy to remove from the cell, and that the tag does not interfere with the natural level of protein expression in the cell. Affinity purification removes this bait protein from the cell along with anything directly or indirectly attached to the bait. A subsequent assay, usually mass spectrometry, determines which proteins have been brought out [1]. However, mass spectrometry only determines which proteins are in the sample, not the binary interactions within it. Therefore, interactions from affinity purification are usually modeled using a "hub-and-spoke" pattern where an edge is placed between the bait protein and each protein pulled out with it, or a clique, where an edge is placed between every pair of proteins pulled out [2]. Both methods are shown in Fig. 1. However, neither method completely captures the information from the interaction data. Drawing the graph as a clique implies binary interactions between proteins that may not physically interact. In Fig. 1, for example, there may be no direct interaction between proteins B and E. Representing this interaction as a clique loses the information that B and E interacted only in the presence of proteins A, C, and D. However, representing the interaction using the hub-and-spoke model loses the information that there is *some* connection between B and E, though it may need to be mediated by other proteins.

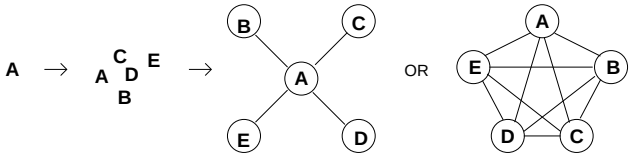

Fig. 1. An example of affinity purification. Protein A is the bait. When it is pulled out, proteins B, C, D, and E are brought out with it. In the PPI network, this is modeled as either the "hub-and-spoke" graph on the left or the 5-clique on the right.

An alternative method that has been proposed is to represent AP data using hypergraphs rather than graphs [3]. Unlike graphs, where an edge can only connect two vertices, the hyperedges of a hypergraph can connect an arbitrary set of vertices. Thus, the interaction in Fig. 1 could be represented with a single hyperedge connecting all five proteins. This would represent the fact that all five proteins interact as a unit without implying anything about whether or not any particular binary relationship might exist or not.

One of the drawbacks to modeling protein interactions as hypergraphs rather than graphs is that many of the statistics used to study graphs do not have well-defined analogs in hypergraphs. In some cases, this is because the statistic has not been defined. With other statistics, such as clustering coefficient, the problem is not that no one has attempted to define an analogous statistic for hypergraphs, but that many different authors have created distinct definitions with little attempt to compare the various definitions.

In this paper, we examine the various proposed definitions for the clustering coefficient of a hypergraph, as well as some new proposed definitions, in the context of protein interaction hypergraphs created from AP data. We will look at the physical meaning for each definition. We will also examine how well each of these clustering coefficient definitions performs in practice at the task of determining protein complexes. We will see which clustering coefficients are best at determining which proteins are part of complexes as well as which pairs of proteins are co-complexed.

A. Definitions

In this paper, all equations will refer to a hypergraph $H = (V, E)$ where V is a set of vertices and $E \subseteq 2^V$ is a set of hyperedges. Each hyperedge $e \in E$ is a subset of V. Let $u, v, w \in V$ and $e_i, e_j, e_k \in E$. For a vertex v, let $M(v)$ be the edges adjacent to v, i.e. $M(v) = \{e_i \in E : v \in e_i\}$. Let $N(v)$ be the neighbors of v, i.e. $N(v) = \{u \in V : \exists e \in E, u, v \in e\}$.

To emphasize the difference between hypergraphs and ordinary graphs, we will occasionally refer to ordinary graphs, where every edge connects only two vertices, as *binary graphs*.

B. Existing Clustering Coefficient Definitions

When the clustering coefficient was first proposed for graphs, it was a measure of the neighborhood density of a single node [4]. However, the initial definitions of clustering coefficient in hypergraphs have instead focused on clustering between pairs of nodes, more analogous to the mutual clustering coefficients of Goldberg and Roth [5]. Under these definitions, the clustering coefficient is a measure of how many common edges a pair of nodes share. One definition for this clustering coefficient that is given in multiple references ([3], [6], [7]) is:

$$CC_{union}(u, v) = \frac{|M(u) \cap M(v)|}{|M(u) \cup M(v)|} \tag{1}$$

There are a few additional definitions for the clustering coefficient between a pair of nodes. These are obtained by varying the denominator [6]:

$$CC_{max}(u, v) = \frac{|M(u) \cap M(v)|}{max\{|M(u)|, |M(v)|\}} \tag{2}$$

$$CC_{min}(u, v) = \frac{|M(u) \cap M(v)|}{min\{|M(u)|, |M(v)|\}} \tag{3}$$

All of these definitions are trying to get at the same physical meaning: given a pair of proteins, what percent of the times that they were pulled out were they pulled out together? The varying definitions come from the fact that it is not obvious what base set should be considered here, but all are roughly the same in terms of the physical quantity being measured.

Using these two-node clustering coefficient definitions, the clustering coefficient of a single vertex is then defined as the average of the clustering coefficient of the vertex and with each its neighbors [6]:

$$CC(u) = \frac{\sum_{v \in N(u)} CC(u, v)}{|N(u)|} \tag{4}$$

This clustering coefficient could be calculated using any of the methods of calculating the clustering coefficient of two vertices.

There have also been several methods proposed to calculate the clustering coefficient on an entire hypergraph. These range from taking the average of the single-node clustering coefficient above over all vertices in the graph to attempts find a hypergraph analogy to the "counting triangles" method of determining the clustering coefficient of a graph [8]. A further discussion of these clustering coefficients is beyond the scope of this paper, however, because these clustering coefficients cannot give us any information about individual proteins.

C. New Clustering Coefficient Definitions

In addition to the previous definitions of clustering coefficients given above, we would like to propose a few new definitions.

1) Clustering Coefficients Defined On Pairs of Vertices: In graph theory, the closest analog to clustering coefficient that deals with the relationships between two nodes is the mutual clustering coefficient of Goldberg and Roth [5]. Goldberg and Roth proposed four possible definitions for the mutual clustering coefficient. Two of these were similar to $CC_{union}(u, v)$ and $CC_{min}(u, v)$. In addition to these, however, there were two additional definitions for mutual clustering coefficient given. These were the geometric and hypergeometric definitions. The geometric definition is something of a compromise between the meet/max and the meet/min standards:

$$CC_{geo}(u, v) = \frac{|M(u) \cap M(v)|}{\sqrt{|M(u)||M(v)|}} \tag{5}$$

The cumulative hypergeometric mutual clustering coefficient, on the other hand, is based on a p-value. The hypergeometric mutual clustering coefficient is designed to answer the question, given the number of nodes in the network and the degrees of two nodes, how likely is it that the overlap of the neighborhoods of those two nodes would be due strictly to chance? The value is defined as:

$$CC_{hgeo}(u, v) = -\log \sum_{i=|M(u) \cap M(v)|}^{min\{|M(u)|, |M(v)|\}} \frac{\binom{|M(u)|}{i}\binom{Total - |M(u)|}{|M(v)| - i}}{\binom{Total}{|M(v)|}} \tag{6}$$

Both of these definitions could also be used to give a mutual clustering coefficient between two nodes in a hypergraph. As in the previous definitions, we will let $M(v)$ be the edges of which v is a member, $M(v) = \{e_i \in E : v \in e_i\}$, and calculate using the formulas above.

Goldberg and Roth tested these mutual clustering coefficients in graphs to measure their ability to predict interactions. In these tests, the meet/min and hypergeometric clustering coefficients were the most effective.

As with the previous definitions, these look at the number of times for a pair of proteins have been pulled out together. $CC_{geo}(v, w)$ is most similar to the previous definitions as also a measure of the percent of the times a pair of proteins has been pulled out together. $CC_{hgeo}(v, w)$, on the other hand, is a measure of the probability that these proteins have been pulled out together this often by chance.

2) Clustering Coefficients Defined On a Single Vertex: The original definition of a clustering coefficient on a vertex of a graph is as the density of the vertex's neighborhood: what percent of a vertices neighbors are neighbors of each other?

It's possible to define a clustering coefficient on a vertex of hypergraph the same way; in fact, this is the same as the clustering coefficient on the graph obtained using the clique model. By doing so, however, we do not take advantage of the extra information in the hypergraph. One way that we might take advantage of this information is by looking at all vertices present in an edge and only "counting" connections between vertices if the edges meet certain conditions. For example, we could look at the number of adjacent nodes that have connections not facilitated by the original node:

$$CC_{ind}(u) = \frac{2\sum_{v,w \in N(u)} I_E(v, w, \neg u)}{|N(u)|(|N(u)| - 1)} \tag{7}$$

where $I_E(v, w, \neg u) = 1$ if there exists $e_i \in E$ such that $v, w \in e_i$ but $u \notin e_i$ and 0 otherwise.

Conversely, we could decide that what interested us was the number of adjacent nodes that have connections that *are* facilitated by the original node, because those nodes might be more likely to share a function with each other and the original node:

$$CC_{dep}(u) = \frac{2\sum_{v,w \in N(u)} I_E(v, w, u)}{|N(u)|(|N(u)| - 1)} \tag{8}$$

where $I_E(v, w, u) = 1$ if there exists $e_i \in E$ such that $u, v, w \in e_i$ and 0 otherwise.

$CC_{ind}(u)$ looks for connections between neighbors of u that do not include u, which has the advantage that any interactions found in this set likely represent a real connection between the neighbors and not an artifact of the data due to the fact that both interact with u. However, it may focus too much on neighbors that have secondary shared functions unrelated to their interactions with u. Conversely, $CC_{dep}(u)$ looks for connections between the neighbors that do include u: this would make it more likely that an edge found this way would represent a true clustering including u and the neighbors, but caution would have to be taken that interactions found this way might simply be an artifact of the data due to the shared interaction with u.

A different definition we could use for the density of a vertices neighborhood would be the amount of overlap between it's adjacent hyperedges. This can be calculated using the following equation:

$$CC_{share}(u) = \frac{(\sum_{e \in M(u)} |e| - 1) - |N(u)|}{|N(u)|(|M(u)| - 1)} \tag{9}$$

In this case, the numerator of the fraction, the difference between $\sum_{e \in M(u)} |e| - 1$ and $|N(u)|$, represents the number of vertices in multiple hyperedges incident to u, where each vertex v is weighted by the number of hyperedges $e \in M(u)$ past the first where $v \in e$. The denominator represents the possible number of such overlaps.

$CC_{share}(u)$ is meant to represent the number of shared edges among u and its neighbors. In terms of protein interactions, it is meant to answer the question: given that a protein was pulled out once with u, what are the chances that it would be in an arbitrary set pulled out with u?

II. Methods

A. Data

For our hypergraphs, we used two different sets of affinity purification data: the filtered data from the Gavin et al. [9] and Ho et al. [10] studies. We used these to create a unified hypergraph.

To determine whether or not a protein is in a complex, as well as which pairs of proteins are co-complexed, we used the Munich Information Center for Protein Sequences (MIPS) database of yeast protein interactions [11]. In order to avoid study bias, we excluded high-throughput complexes determined by the Gavin et al. and Ho et al. studies.

B. Testing for correlations with protein complexes

For each vertex in the hypergraph, we calculated ten single vertex hypergraph clustering coefficients: the eight hypergraph clustering coefficients discussed here along with the clustering coefficients in the binary graphs created by both the clique and hub-and-spoke methods. We then determined the the Pearson correlation coefficient between the value of the clustering coefficients and whether a protein was complexed. We also ran a Student's T-test to determine the probability that the clustering coefficient values for proteins in complexes and those for proteins not in complexes come from the same distribution.

We ran similar tests for the two-node clustering coefficients. For each pair of vertices in the hypergraphs, we calculated all five of the two-node clustering coefficients as well as determining whether or not the pair was co-complexed. For each clustering coefficient, we then determined correlation between pairs with high values of the clustering coefficient and whether or not the pair was co-complexed using the same three methods as we did for the single vertex test: a Pearson correlation coefficient and a Student's T-test.

Similarly, for each pair of vertices the hypergraph, we calculated all five of the two-node clustering coefficients. For each clustering coefficient, we then computed the Pearson correlation coefficient between the clustering coefficient and whether or not the pair is co-complexed and the Student's T-test for known co-complexed protein pairs vs. other pairs of proteins.

III. Results and Discussion

A. Single-node clustering coefficients

Many proteins did not have valid values for all clustering coefficients (Table I). Therefore, so that clustering coefficients could be compared fairly, we restrict our analysis to those proteins for which all clustering coefficients had a valid value. Results are in Table II.

CC_{hgeo} performed best here, outperforming not only all other hypergraph clustering coefficients but also the clustering coefficients in the graphs obtained using both methods of modeling the AP data. CC_{share} also performed well, essentially

TABLE I
PERCENT OF NODES THAT HAD A VALID VALUE FOR EACH OF THE CLUSTERING COEFFICIENTS.

Clustering Coefficient	Fraction of nodes with a valid value
$CC_{Min/Max/Union/Geo/Hgeo}$	0.98
CC_{Share}	0.52
$CC_{Dep/Ind}$/Clique CC	0.90
Hub/Spoke CC	0.61

TABLE II
CORRELATION BETWEEN WHETHER A PROTEIN IS PART OF A COMPLEX AND ITS SINGLE-NODE CLUSTERING COEFFICIENTS.

Clustering Coefficient	T-test	Pearson
CC_{hgeo}	2.5e-20	.2521
Hub/Spoke CC	1.3e-16	.23318
CC_{share}	1.7e-16	.26515
CC_{geo}	8.6e-8	.19111
CC_{union}	4.1e-7	.18524
CC_{max}	3.3e-6	.17206
CC_{min}	.00025	.11506
CC_{dep}	.15348	.04646
Clique CC	.15775	.04374
CC_{ind}	.69689	.00032

tied with the clustering coefficient on the graph obtained from the hub-and-spoke model. Other than CC_{hgeo}, the other single node measures that average a two-node clustering coefficient performed moderately well, significantly less well than the top three, but better than the clustering coefficients on the graph from the clique model.

It is worth noting that CC_{hgeo} is scaled differently than the other clustering coefficients. While the other clustering coefficients give a value between 0 and 1, CC_{hgeo} is scaled by a negative logarithm and can take any positive value. This may affect the correlation values.

CC_{dep}, the clustering coefficient on the graph obtained from the clique model, and CC_{ind} did not perform well as an indicator of complexes. This is hardly surprising in the case of CC_{ind} given the way it was defined: because it requires two neighbors of a node to share an edge that does not also contain the original node, it is not unlikely that these additional edges do not represent a shared module with the original node. It is somewhat surprising, perhaps, that the dependent clustering coefficient performed no better than the projection graph, but this may be due to the limited amount of data. Many nodes were part of only a single hyperedge, meaning that their dependent clustering coefficient and projection clustering coefficients were the same.

B. Two-node clustering coefficients

We considered only pairs of vertices where the clustering coefficient was non-zero because there are so many pairs that share no edges, and two-node clustering coefficients cannot give any information to differentiate among these pairs. Even with this restriction, there were so many pairs that all T-tests returned a value of 0. Thus, the only way to compare the different clustering coefficients was by looking at the Pearson correlation coefficients. See Table III.

TABLE III
CORRELATION BETWEEN WHETHER A PAIR IS CO-COMPLEXED AND THE TWO-NODE CLUSTERING COEFFICIENTS BETWEEN THAT PAIR.

Clustering Coefficient	Pearson
CC_{geo}	0.336581
CC_{union}	0.335149
CC_{max}	0.331244
CC_{hgeo}	0.302062
CC_{min}	0.232638

TABLE IV
CORRELATION BETWEEN WHETHER A PAIR IS CO-COMPLEXED AND THE MUTUAL CLUSTERING COEFFICIENTS BETWEEN THAT PAIR IN THE HUB-AND-SPOKE GRAPH.

Mutual Clustering Coefficient	Pearson
Max	0.17588
Union	0.16905
Hyper	0.16415
Geo	0.16326
Min	0.13442

TABLE V
CORRELATION BETWEEN WHETHER A PAIR IS CO-COMPLEXED AND THE MUTUAL CLUSTERING COEFFICIENTS BETWEEN THAT PAIR IN THE CLIQUE GRAPH.

Mutual Clustering Coefficient	Pearson
Geo	0.29001
Union	0.28294
Hyper	0.24858
Max	0.24416
Min	0.19241

For comparison purposes, we also calculated the Pearson correlation between the mutual clustering coefficient and whether or not a pair was co-complexed in both the hub-and-spoke graph (Table IV) and clique graph (Table V). In the clique graph, due to the large number of pairs which had non-zero mutual clustering coefficients, we only examined a sample including roughly half the pairs. Although the mutual clustering coefficients and the hypergraph 2-node clustering coefficients are not precisely analogous to each other (the mutual clustering coefficient measures shared neighbors while the hypergraph clustering coefficients represent shared interactions), the mutual clustering coefficient is the closest measure in binary graphs. As can be see from the two tables, most hypergraph clustering coefficients outperform the mutual clustering coefficients at determining whether or not a pair of proteins is co-complexed in both types of graphs.

It is interesting to note that the hub-and-spoke graph model for the data performs much better than the clique model at determining complexed proteins using clustering coefficients, but the clique model performs much better than the hub-and-spoke model at determining co-complexed proteins using mutual clustering coefficient. Upon further examination,

however, this should not be surprising. The ways in which information is distorted in the different models leave them vulnerable in different ways. By including so many edges from each hyperedge, proteins in the clique graph appear to have more more binary interactions between their neighbors than actually exist, causing even non-complexed proteins to have high clustering coefficients. The hub-and-spoke model, on the other hand, leaves out potential interactions between non-bait proteins, making it difficult to detect relationships between these nodes. Only the hypergraph model retains enough information to perform well at both tasks.

Another noteworthy result is the fact that CC_{min} performed the worst out of all the two-node-based clustering coefficients both at finding complexed nodes and co-complexed nodes. The minimum mutual clustering coefficient also performed the worst at finding co-complexed nodes in both the clique and hub-and-spoke graphs. However, in their paper introducing the mutual clustering coefficient, Goldberg and Roth recommended the minimum as the best performing of the ratio methods for calculating the mutual clustering coefficient. This difference may be due to the fact that Goldberg and Roth were using mutual clustering coefficients for a different application (providing evidence for possible interactions rather than finding complexes) and looking at a different type of protein interaction data (yeast 2-hybrid rather than AP). The fact that this discrepancy exists, however, suggests that we need to be cautious about labeling one possible clustering coefficient as "the best hypergraph clustering coefficient" and instead should consider both the nature of the hypergraph and the problem we are trying to solve.

IV. Conclusion

We have looked at 8 single-node and 5 two-node definitions of hypergraph clustering coefficients in the context of protein interaction hypernetworks. We examined both existing and new definitions. For all of these definitions, we considered both their physical meaning in the context of protein interactions and how they perform at predicting complexed or co-complexed proteins. For predicting complexed proteins, we found that two of our new hypergraph clustering coefficients perform as well or better than the clustering coefficient in the graph generated by the hub-and-spoke model, while virtually all of the hypergraph clustering coefficients performed better than the clustering coefficient in the graph generated by the clique method. For predicting pairs of co-complexed proteins, 4 of 5 hypergraph clustering coefficients performed better than the mutual clustering coefficients in graphs generated using either method. We conclude that hypergraph clustering coefficients are more suitable to predicting proteins that are components of a common protein complex than for predicting proteins that are in a protein complex. This may be because there are biological functions other than being a component of a complex that could cause a protein to have high clustering coefficient (e.g., proteins within a pathway or cellular process, or "linker" proteins that interact with different processes).

Our results demonstrate the potential use of hypergraphs in modeling protein interaction data. Although they are not clearly better, there are single-node hypergraph clustering coefficients that are as good as (or possibly better than) the graph clustering coefficients using the current methods of modeling AP data as a graph. Further, the two-node hypergraph clustering coefficients are significantly better at predicting proteins that are within a common complex than previous methods. We hope that this work will inspire others to use hypergraphs to model protein interaction data, and to create new hypergraph methods to analyze this data.

References

[1] G. Rigaut, A. Shevchenko, B. Rutz, M. Wilm, M. Mann, and B. Seraphin, "A generic protein purification method for protein complex characterization and proteome exploration," *NATURE BIOTECHNOLOGY*, vol. 17, no. 10, pp. 1030–1032, OCT 1999.

[2] G. Bader and C. Hogue, "Analyzing yeast protein-protein interaction data obtained from different sources," *NATURE BIOTECHNOLOGY*, vol. 20, no. 10, pp. 991–997, OCT 2002.

[3] S. Klamt, U.-U. Haus, and F. Theis, "Hypergraphs and Cellular Networks," *PLOS COMPUTATIONAL BIOLOGY*, vol. 5, no. 5, MAY 2009.

[4] D. Watts and S. Strogatz, "Collective dynamics of 'small-world' networks," *NATURE*, vol. 393, no. 6684, pp. 440–442, JUN 4 1998.

[5] D. Goldberg and F. Roth, "Assessing experimentally derived interactions in a small world," *PROCEEDINGS OF THE NATIONAL ACADEMY OF SCIENCES OF THE UNITED STATES OF AMERICA*, vol. 100, no. 8, pp. 4372–4376, APR 15 2003.

[6] M. Latapy, C. Magnien, and N. Del Vecchio, "Basic notions for the analysis of large two-mode networks," *SOCIAL NETWORKS*, vol. 30, no. 1, pp. 31–48, JAN 2008.

[7] S. Le Blond, J.-L. Guillaume, and M. Latapy, "Clustering in p2p exchanges and consequences on performances," in *Peer-to-Peer Systems IV*, ser. Lecture Notes in Computer Science, M. Castro and R. van Renesse, Eds. Springer Berlin / Heidelberg, 2005, vol. 3640, pp. 193–204.

[8] E. Estrada and J. Rodriguez-Velazquez, "Subgraph centrality and clustering in complex hyper-networks," *PHYSICA A-STATISTICAL MECHANICS AND ITS APPLICATIONS*, vol. 364, pp. 581–594, MAY 15 2006.

[9] A. Gavin, M. Bosche, R. Krause, P. Grandi, M. Marzioch, A. Bauer, J. Schultz, J. Rick, A. Michon, C. Cruciat, M. Remor, C. Hofert, M. Schelder, M. Brajenovic, H. Ruffner, A. Merino, K. Klein, M. Hudak, D. Dickson, T. Rudi, V. Gnau, A. Bauch, S. Bastuck, B. Huhse, C. Leutwein, M. Heurtier, R. Copley, A. Edelmann, E. Querfurth, V. Rybin, G. Drewes, M. Raida, T. Bouwmeester, P. Bork, B. Seraphin, B. Kuster, G. Neubauer, and G. Superti-Furga, "Functional organization of the yeast proteome by systematic analysis of protein complexes," *NATURE*, vol. 415, no. 6868, pp. 141–147, JAN 10 2002.

[10] Y. Ho, A. Gruhler, A. Heilbut, G. Bader, L. Moore, S. Adams, A. Millar, P. Taylor, K. Bennett, K. Boutilier, L. Yang, C. Wolting, I. Donaldson, S. Schandorff, J. Shewnarane, M. Vo, J. Taggart, M. Goudreault, B. Muskat, C. Alfarano, D. Dewar, Z. Lin, K. Michalickova, A. Willems, H. Sassi, P. Nielsen, K. Rasmussen, J. Andersen, L. Johansen, L. Hansen, H. Jespersen, A. Podtelejnikov, E. Nielsen, J. Crawford, V. Poulsen, B. Sorensen, J. Matthiesen, R. Hendrickson, F. Gleeson, T. Pawson, M. Moran, D. Durocher, M. Mann, C. Hogue, D. Figeys, and M. Tyers, "Systematic identification of protein complexes in Saccharomyces cerevisiae by mass spectrometry," *NATURE*, vol. 415, no. 6868, pp. 180–183, JAN 10 2002.

[11] H. W. Mewes, D. Frishman, K. F. X. Mayer, M. Münsterkötter, O. Noubibou, P. Pagel, T. Rattei, M. Oesterheld, A. Ruepp, and V. Stümpflen, "Mips: analysis and annotation of proteins from whole genomes in 2005," *NUCLEIC ACIDS RES*, vol. 34, pp. D169–D172, 2006.