Evaluating Functional Group Detection in *Saccharomyces cerevisiae* Gene Networks through Dynamic Community Detection and Metric Optimization

Carmelo Ellezandro Atienza[1] and Calvin James Maximo[2]

[1] Algorithms and Complexity Lab,
University of the Philippines Diliman
cratienza1@up.edu.ph
[2] Algorithms and Complexity Lab,
University of the Philippines Diliman
ctmaximo1@up.edu.ph

1 Background of the Study

1.1 Network Biology and Biological Networks

Biological systems are inherently complex—molecules such as genes, proteins, and metabolites intricately interact with each other to perform essential functions that regulate cellular processes. Barabási and Oltvai (2004) highlight that traditional reductionist approaches in examining biological functions is insufficient, since such methods only focus on individual cellular components. In reality, biological characteristics are seen as a product from complex interactions between multiple molecules, such as proteins, DNA, and small molecules (Barabási and Oltvai, 2004). This shift in perspective gave way for network biology: an interdisciplinary domain which integrates the computational field of network theory with biological sciences.

Girvan and Newman (2002) emphasize that many networks in different domains of study exhibit community structures, where nodes cluster into tightly connected groups. In particular, this phenomenon is prevalent in biological systems, where such community structures may pertain to functional modules—a group of genes, proteins or other biological entities that work in conjunction to perform a specific biological role. Biological networks, such as gene regulatory networks (GRNs), protein-protein interaction networks (PPIs) and metabolic pathways, are compelling topics for research as it can uncover how certain molecules are grouped within cells (Yu et al., 2013).

Dorogovtsev and Mendes (2002) explained that a crucial aspect of biological networks is that they are not static—they evolve over time, adapting to changing internal and external conditions. This dynamic nature is especially evident in gene co-expression networks, where gene interactions fluctuate throughout different stages of biological processes. Thus, studying the dynamic community structure within these networks can provide insights into how biological functions are regulated, and how brief interactions between genes may arise and disappear during key cellular events. This evolving nature of biological networks makes dynamic community detection a critical tool for revealing how these molecular interactions change over time.

1.2 Dynamic Community Detection in Networks

Building on this understanding of dynamic biological networks, dynamic community detection has emerged as a key approach for analyzing networks across different domains, including social networks and biological systems. Much of the existing research regarding community detection begins with a strict assumption that real-world phenomena are represented using static networks which are fixed and unchanging over time. However, this assumption does not align with the evolutionary world around us.

Complex networks such as gene regulatory networks are generated at a blistering pace, which spawns the problem of the difficulty of modeling these networks as static relations. Time is an important component in studying the evolution of connectivity and patterns. The dynamic nature of these real-world networks further complicates the task of community detection as the structure of these communities evolve along with the networks themselves. The need to integrate the knowledge from the temporal dimension has led to the

emergence of a new field of study: dynamic network analysis (Rossetti and Cazabet, 2018; Sattar et al., 2023).

Dynamic community detection is one of the key problems under this field. Over time, nodes and edges in a dynamic network can appear and disappear, causing significant shifts in its structure. Communities, which are the fundamental building blocks of this complex network, are heavily influenced by these local changes. While the objective of traditional community detection algorithms is to reveal hidden substructures, the goal of dynamic approaches is to monitor these local topological changes and their transformations over time (Rossetti and Cazabet, 2018). Understanding how community structures in complex networks evolve over time are vital in gaining valuable insights. The insights can be used to understand the structural dynamicity of the network. There is also great potential in the medical domain for analyzing disease progression (Redekar and Varma, 2022).

1.3 Gene Co-expression Networks and their Dynamic Nature

Gene co-expression networks (GCNs) are widely used to model interaction between genes based on their expression levels across different biological conditions or datasets, and across varying time points (Ovens et al., 2021; Lau et al., 2020). In GCNs, nodes represent genes, and edges signify co-expression relationships, where the strength of an edge correlates with the similarity in gene expression.

That said, GCNs will be the focus of this paper because of their temporal and dynamic properties. A study done by Lau et al. (2020) show the need to capture the temporal dynamics of GCNs, as gene co-expression patterns fluctuate significantly over time. Consequently, the community structure of GCNs will also fluctuate, with different clusters of genes forming or dissipating at different timepoints. These changes highlight the importance of dynamic community detection techniques in capturing the evolving structure of GCNs (Ovens et al., 2021), which can reveal crucial insights on their functional grouping.

2 Preliminaries

2.1 Gene Co-expression Networks (GCNs)

A graph-based representation of genes, where nodes represent individual genes and edges represent co-expression relationships based on their expression patterns across multiple conditions or time points. Co-expression relationship refers to the correlation between the expression levels of two or more genes across different conditions (Stuart et al., 2003). Expression levels refer to the amount of mRNA produced by a gene in a cell at a given time. To put it simply, co-expression relationship is when two or more genes show similar patterns of activity (expression levels) across different conditions or samples. If their activity goes up or down together, they are said to be co-expressed, which often means they might work together or be controlled by the same regulatory processes (Ovens et al., 2021).

2.2 Community Detection

The process of identifying clusters of groups of nodes in a network that are more densely connected internally than with nodes outside the group. There are several community detection techniques (Redekar and Varma, 2022), depending on the nature of the network, whether static or dynamic. In modern network science, community detection is an NP-Hard problem, posing as one of the most challenging problems in this field (Priya et al., 2023)

2.2.1 Static Community Detection. Static community detection assumes that the network is fixed across different time points. From there, communities are identified based on a single snapshot of the network. This method is well-suited for networks where relationships between nodes are stable and predictable, if not unchanging. Techniques such as Louvain or Girvan-Newman are prime examples of static community detection algorithms, where it can reveal hierarchical structures in the network (Redekar and Varma, 2022).

2.2.2 Dynamic Community Detection. Dynamic community detection is designed for networks that evolve over time. It is able to detect communities and how they vary at each time step, tracking the evolution, merging and splitting of nodes and edges as the network changes (Dorogovtsev and Mendes, 2002). Capturing these temporal changes are crucial for understanding time-dependent phenomena, such as gene interactions that change during biological processes.

For the purposes of this paper, dynamic community detection is employed in discovering the functional classification of *Saccharomyces cerevisiae*.

2.3 *Saccharomyces cerevisiae*

According to Parapouli et al. (2020), *Saccharomyces cerevisiae* is a unicellular fungus that is hailed as model organism—extensively used in biological research due to its well-annotated genome. In the field of bioinformatics, *Saccharomyces cerevisiae* contains around 6,000 documented genes which allows for in-depth research on cellular processes that is shared with higher-level eukaryotes. This includes gene regulation, cell cycle control, and metabolic functions. Due to its accessibility and applications, the gene expression data of this fungus is an ideal candidate for constructing a GCN through Dynamic Community Detection techniques.

For brevity, *Saccharomyces cerevisiae* is succinctly referred to as *S. cerevisiae*, or simply just *yeast* in this paper.

3 Review of Related Literature

3.1 Time-Series Gene Expression Data of *S. cerevisiae*

Cho et al. (1998) conducted a time-series laboratory experiment on *S. cerevisiae* across the different phases of the mitotic cell cycle. Of the original 6220 documented genes from the yeast, it was determined that 416 genes were actively involved in 17 different time points. These genes were functionally grouped according to their activity during the different phases of the cell cycle:

1. Early G1 Phase
2. Late G1 Phase
3. Synthesis Phase
4. G2 Phase
5. Mitosis
6. Multi-phase (occurring in different phases)

This grouping according to the cell cycle phase is denoted as the *Level 1* classification within the dataset. Another classification occurs within the dataset, denoted as *Level 2*, which represents a more specific subcategory or function within the broader classification in *Level 1*. For example, within the Early G1 Phase in *Level 1*, it can be further branched in to specific cellular processes in *Level 2*, such as Cell Cycle Regulation, Directional Growth, DNA Replication, and more. As such, one may observe the hierarchical and dynamic structure of the data, where categories and subcategories form across the 17 different time points.

The resulting data from Cho et al. (1998) laid the foundation for Yeung et al. (2001), where they further processed the data for clustering analysis. Notably, they dropped the sixth functional grouping (multi-phase) so that the dataset only shows genes that peak in only one phase. This leaves with a gene expression data of 384 genes across 17 different time points.

Though this dataset is comprehensive, it is possibly outdated, especially with the rise of modern biotechnologies. Calderon et al. (2024) attempted to reconcile the Cho dataset with the more precise and updated groupings from the Kyoto Encyclopedia of Genes and Genomes (KEGG) (Kanehisa et al., 2022). Calderon et al. (2024) undertook a detailed comparison of the gene classifications. Their goal was to align the functional groupings from the time-series data of Cho with KEGG's established pathways and gene groupings, which are based on more modern genomic studies.

The KEGG *S. cerevisiae* dataset contains three levels of classification, as opposed to the Cho et al. (1998) of two levels. To reconcile the three levels of classification from KEGG with the two levels in Cho's dataset, they performed a detailed comparison using the Adjusted Rand Index (ARI) to measure the similarity between the groupings at different levels. First, the KEGG data was filtered to include only the genes that

had both complete classification information in KEGG and were present in the Cho dataset. As a result, the 384 genes was reduced down to 149 genes. They then compared the classifications at each level of KEGG (*Levels 1, 2* and *3*) with each of the two levels from Cho's dataset (*Level 1* and *Level 2*).

The ARI was calculated for every possible combination of levels between KEGG and Cho, as shown in the table below.

Table 1. Relationship between KEGG and CHO classifications with ARI Calderon et al., 2024

KEGG Level	CHO Level	ARI
1	1	0.022699
1	2	0.083272
2	1	0.057357
2	2	0.118445
3	1	0.043423
3	2	0.156492

The highest ARI score was found between KEGG Level 3 and Cho Level 2, indicating the strongest agreement between these more detailed levels of classification. Nonetheless, this pair still exhibits a weak agreement score (≈ 0.16).

3.2 Hierarchical Community Detection and the ARI Agreement Measure in Gene Co-expression Networks

In the same study, Calderon et al. (2024) explored hierarchical community detection (HCD) in gene co-expression networks (GCNs) of *S. cerevisiae*, aiming to identify functional gene communities within the yeast's cell cycle. Their study relied on gene expression data from the Cho dataset, using community detection algorithms to detect modular and hierarchical structures in gene interactions. In particular, they used three HCDs: Girvan-Newman (GN), Paris, and Local Optimization Function Model (LFM). Additionally, as shown in the previous subsection, they updated the functional classifications using gene annotations from the Kyoto Encyclopedia of Genes and Genomes (KEGG), ensuring that each gene's classification was current and functionally relevant.

To assess the alignment between the functional classifications from Cho and KEGG, Calderon et al. (2024) employed the Adjusted Rand Index (ARI). The ARI was used twice in their study: first, to measure the degree of similarity between classifications in the Cho dataset and KEGG, and second, to evaluate the alignment of detected gene communities with both the Cho and KEGG dataset.

1. **Comparing KEGG and Cho Classifications:** The results were previously shown in Table 1, where we observe the underwhelming ARI scores between the classifications of KEGG and Cho. Ideally, since KEGG aggregates and organizes all available gene data, the Cho classifications should have shown stronger alignment, as they represent a subset of KEGG's comprehensive gene data.
2. **Evaluating Community Detection Results:** After applying three hierarchical community detection algorithms—Girvan-Newman (GN), Local Fitness Model (LFM), and Paris—to the gene co-expression network, Calderon and Ventures used ARI again to evaluate the alignment between the detected communities and the Cho dataset, as well as with the KEGG dataset. The results revealed relatively weak ARI scores across all algorithms, suggesting limited biological alignment with the known gene functions.

The weak ARI scores suggests that ARI may not fully capture the relationship between the classifications or that the hierarchical techniques used do not align closely with biological reality.

3.3 Dynamic Community Detection Algorithms

Dynamic community detection (DCD) algorithms not only uncover communities in complex networks but also the evolution of these networks over time. Several studies have probed into the performance of various

DCD algorithms on both real-world and synthetic datasets (Ma and Dong, 2017; Seifikar et al., 2020; Singh et al., 2020; Redekar and Varma, 2022; Sattar et al., 2023; Calderon et al., 2024). We will be taking a look on a selection of noteworthy algorithms from previous studies, which all offer unique methods in discovering relevant communities in dynamic networks.

3.3.1 TILES Algorithm TILES is an algorithm that detects and tracks the evolution of overlapping communities in dynamic social networks. Its approach is similar to that of a fall of a domino tile. Every time a new interaction (edge) is introduced to the network, it first updates the local communities before propagating the changes to the surrounding nodes which in turn affects their community memberships. The algorithm characterizes two types of community memberships for the nodes: *peripheral* membership and *core* membership. A node is a *core* node if it is part of a triangle within the same community. Meanwhile, it is classified as a *peripheral* node if it is connected to a *core* node but not part of any triangle within the *core* node's community. Only *core* nodes are allowed to propagate community membership to their neighboring nodes. Therefore, the TILES algorithm generates overlapping communities, where a node can belong to different communities at the same time which is identical to real-world networks (Rossetti et al., 2017).

The online nature of this algorithm is advantageous. The updating process is expedited because the computation of network substructures is localized and involves a minimal number of nodes and communities. Two evolutionary behaviors can also be observed due to this algorithm: (1) the persistence of individuals' ties to the communities, and (2) the gradual evolution of communities through interactions. Compositionality is an interesting property of TILES, which makes its execution parallelizable, thus allowing for faster computations of communities (Rossetti et al., 2017).

Rossetti et al. (2017) evaluated the performance of TILES versus other CD algorithms (DEMON, CFinder, and iLCD) both on synthetic and real-world networks, which included social media and call network data. The results showed that TILES had faster execution times and was more accurate with ground truth communities. It was also able to identify significant lifecycle events which affected the structure of these communities, which provided great insights into the dynamics of these networks. TILES could also be adapted to gene co-expression networks because they are dynamic in nature as well.

3.3.2 InfoMap Algorithm InfoMap is an algorithm that utilizes a map equation to detect community structures in networks. It views community detection as a coding problem and applies the minimum description length (MDL) in computing the optimal partition. In this algorithm, the description length of a random walker in the network is expressed by the map equation. A partition with a good modular structure is usually emphasized by a smaller description length, and it is incorporated in the objective function in uncovering better network partitions. A partition's description length can be compressed if a random walker tends to stay longer within it, so those with greater community structure will yield minimum description length. This approach groups nodes through a collective process, initially treating each node as a separate module. Then, it randomly selects nodes to combine together, which leads to a significant decrease in the map equation. In the succeeding steps, previously formed modules are treated as nodes and the process is repeated until a decrease in the map equation is no longer possible. A fundamental element of this algorithm is the utilization of flows in the graph. It is robust as long as clusters inside it flow well, which indicates the tendency of staying inside the same cluster when doing random walks. Otherwise, it fails if there is a lack of flow or if it mostly results to deadlocks (Held et al., 2016; Chejara and Godfrey, 2017; Sun et al., 2019).

3.4 Statistical Analysis of Dynamic Community Detection

3.4.1 Internal Criteria Internal criteria are used to assess the clustering results of the CD algorithms without using external information such as ground-truth data, and it only utilizes information that are inherent to the data (Liu et al., 2020).

One of the widely used metrics that we will employ in this study is Modularity, which is used to assess the quality of a network community structure. High modularity is an indication that vertices within the same community have denser connections compared to vertices across different communities which have sparser connections. It is particularly useful in assessing the strength of an algorithm in dividing a given network into communities (Zhuang et al., 2021).

Another metric that we will consider applying is Closeness Centrality. It is used to quantify the speed at which information gets transmitted to a node from all other nodes in a given network. This metric is a good measure of the overall accessibility of a node in the network and their capability to obtain information rapidly (Kherad et al., 2024).

By applying these internal criteria — modularity and closeness central — we aim to evaluate the effectiveness of DCD algorithms both in the strength and cohesion of detected communities and also the accessibility and influence of individual nodes as the network structure changes over time. It can give us valuable biological insights into the temporal nature of the *S. cerevisiae* gene co-expression network.

3.4.2 External Criteria On the other hand, external criteria are used to evaluate the similarity of the clustering results of the CD algorithms against a "ground truth" generated by prior experiments.

The Adjusted Rand Index (ARI) is a key metric in assessing the similarity between two communities. To understand it, we must first understand the Rand Index (RI). RI considers every pair of nodes and determine whether they belong in the same community. If so, it compares these pairs against another graph. For a given node pair, it gets a score of 1 if it is either in the same community for both graphs or not in the same community for both graphs. Otherwise, it gets a 0 which indicates the difference between the community structure of the graphs. The result of dividing the sum of the scores of all node pairs and the number of pairs is between 0 and 1, where the result is directly proportional to the similarity of the two graphs' community structures (Bakker et al., 2018). The equation for RI is as follows:

$$Rand(A, B) = \frac{n_{00} + n_{11}}{n_{00} + n_{11} + n_{01} + n_{10}} \tag{1}$$

where A and B are the methods which yielded communities. n_{00} is the number of node pairs from a network G which are in different communities in both A and B. n_{11} is the number of node pairs from a network G which are in the same community in A and also in the same community in B. n_{00} and n_{11} are considered to be as agreements in terms of classification. Meanwhile, n_{01} and n_{10} are disagreement quantities. n_{01} is the number of node pairs that belong in the same community in A but belong in different communities in B. Similarly, n_{10} is the number of node pairs that belong in the same community in B but belong in different communities in A. RI can be interpreted as the probability that a node pair from A and B will agree. Equation 1 can be rewritten as Equation 2 as follows:

$$\frac{n_{00} + n_{11}}{\binom{N}{2}} \tag{2}$$

where N is the total number of nodes, and $\binom{N}{2}$ is calculated as $\frac{N(N-1)}{2}$ and is equal to the number of pairs. RI is not a normalized quantity, wherein the upper bound is 1 but the lower bound is more than 0. (Rahiminejad et al., 2019).

With that in mind, ARI is simply a normalized version of RI where chance is taken into account. The equation for ARI is as follows:

$$\begin{aligned} RI &= \sum_{i=1}^{c_A} \sum_{j=1}^{c_B} \binom{N_{ij}}{2} \\ ExpectedRI &= \frac{2 \sum_{i=1}^{c_A} \binom{N_{i.}}{2} \sum_{j=1}^{c_B} \binom{N_{.j}}{2}}{N(N-1)} \\ MaxRI &= \frac{1}{2} \left[\sum_{i=1}^{c_A} \binom{N_{i.}}{2} + \sum_{j=1}^{c_B} \binom{N_{.j}}{2} \right] \\ ARI &= \frac{RI - ExpectedRI}{MaxRI - ExpectedRI} \end{aligned} \tag{3}$$

Let N be a confusion matrix that corresponds to methods A and B, whose rows are the communities in A, and columns are the communities in B. N_{ij} corresponds to the number of nodes in community i of A and

community j of B. $N_{i.}$ is the sum of all columns of row i and denotes the total number of nodes in community i of A . $N_{.j}$ is the sum of all rows of column j and denotes the total number of nodes in community j of B. Lastly, c_A and c_B are the number of communities detected by A and B, respectively. An ARI value of 1 indicates the identicality between the predicted and actual clustering, while a negative value implies that the cluster agreement is lower than expected by chance (Rahiminejad et al., 2019; Clemente et al., 2022).

Another metric that is commonly used when ground-truth information is present is Normalized Mutual Information (NMI). A confusion matrix N is defined, where the rows are the ground truth class labels, and the columns are the communities generated by a CD algorithm. An element N_{ij} is the number of nodes with class label i that are in community j. The NMI is defined as follows:

$$\mathrm{NMI}(A,C) = \frac{-2\sum_{i=1}^{n_A}\sum_{j=1}^{n_C} N_{ij}\log\left(\frac{N_{ij}N}{N_{i.}N_{.j}}\right)}{\sum_{i=1}^{n_A} N_{i.}\log\left(\frac{N_{i.}}{N}\right) + \sum_{j=1}^{n_C} N_{.j}\log\left(\frac{N_{.j}}{N}\right)} \tag{4}$$

where A is the ground-truth partition, C is the CD algorithm partition, n_A is the number of ground-truth communities, n_C is the number of communities detected by the CD algorithm, $N_{i.}$ is the number of nodes with class label i, and $N_{.j}$ is the number of nodes in community j. NMI is 1 if the partitions are equal, and 0 otherwise (Márquez and Weber, 2023).

By employing these external criteria — ARI and NMI — we will be able to understand the accuracy of DCD algorithms when tested with real-world data, particularly biological data.

4 Research Aims

4.1 Statement of the Problem

The study of gene co-expression networks (GCNs) in *S. cerevisiae* has provided significant insights into the modular and hierarchical structure of gene interactions, particularly as they relate to the cell cycle (Calderon et al., 2024; Clemente et al., 2022). Calderon et al. (2024) applied hierarchical community detection techniques and the Adjusted Rand Index (ARI) to evaluate the alignment between the detected gene communities and known biological classifications. However, their analysis yielded weak agreement scores, indicating that the hierarchical approach or the ARI metric may not capture the full structure and functional relevance of the gene communities in *S. cerevisiae*.

This gap highlights an open problem: how can the detection of gene communities in *S. cerevisiae* be improved to achieve stronger agreement with known functional classifications and more accurately reflect the biological roles of gene interactions within the cell cycle? Addressing this question requires the application of dynamic community detection techniques and the exploration of alternative metrics beyond ARI, with the aim of producing more consistent, interpretable results.

4.2 Objectives of the Study

This study aims to achieve stronger agreement with biological classifications by employing dynamic community detection and evaluating the effectiveness of alternative similarity metrics on the dataset used by Calderon et al. (2024). The objectives of this research paper is as follows:

1. **Evaluate Alternative Similarity Metrics for Improved Community Consistency**: Explore and implement alternative similarity metrics beyond the Adjusted Rand Index (ARI) to assess the alignment between gene communities and established functional groupings. The goal is to develop a Modified Rand Index (MRI) metric that offers stronger, more reliable results for comparing community structures in gene networks with multiple sources of truths.
2. **Predict Functional Classification of Genes using Dynamic Community Detection Algorithms:** Use a dynamic approach for community detection on both the Cho and KEGG dataset to predict functional groupings through the detected communities.
3. **Compare the Performance of Dynamic and Hierarchical Community Detection Algorithms**: Assess the effectiveness of dynamic community detection algorithms in comparison to hierarchical methods, as well as comparisons between different dynamic algorithms. This objective aims to determine which approach provides more accurate and biologically meaningful results in gene co-expression networks.

4.3 Scope and Limitations of the Study

The study will focus primarily on dynamic community detection and the modification of the currently existing Rand Index formula. The dataset of 384 genes will be the focus of this study, and it is the same dataset used by Calderon et al. (2024). Our goal is to approach this gene co-expression network from a different perspective through the use of dynamic community detection methods and our Modified Rand Index (MRI) [5.1].

We will not delve into the development of a dynamic novel community detection algorithm for this study. Instead, we will employ publicly available algorithms that have been widely used in different studies, especially in biological contexts. This ensures that the methodologies we will implement are proven to be advantageous in achieving our study's objectives.

Furthermore, a limitation of this study is in our choice of testing the MRI. Calderon et al. (2024) utilized the Adjusted Rand Index, an adjusted version of the Rand Index. When assessing the results of MRI, we cannot directly compare it with ARI as the former is a non-adjusted metric. Although we can attempt to formulate an adjusted version of the MRI, we leave this as a limitation in our study for time feasibility.

In evaluating the performance of our chosen algorithms, we will utilize both publicly available metrics and our proposed metric (MRI), which will be discussed further in [5.1]. The chosen metrics are critical and relevant in properly assessing the performance of dynamic community detection algorithms on a biological network. The proposed metric is also crucial to address a major weakness of an existing metric (RI). These metrics will help us in interpreting the results and applying the insights we will gain in a biological context.

By relying on existing methodologies, our aim is to efficiently allocate our time and resources to explore the dynamic nature of the *Saccharomyces cerevisiae* gene co-expression network. Our approach enables us to gain a deeper understanding of the network structure, and the biological significance in how it differs across various time points. There are valuable biological insights to be gained, especially in the use of dynamic community detection algorithms for biological networks as most existing literature revolves around static algorithms.

5 Methodology

To fulfill our research aims, we present a three-fold approach to our methodology, each addressing a specific objective.

1. **Modified Rand Index:** To evaluate an alternative approach to the Adjusted Rand Index (ARI), we propose a new modified scheme based on the definition of Rand Index according to Bakker et al. (2018). We also present a comparative analysis of our modified Rand Index similarity metric across different publicly available metrics.
2. **Dynamic Community Detection for Predicting Functional Classifications:** To predict functional groupings through detected dynamic communities, we identify and employ well-known and publicly available DCD algorithms from the CDLib library for Python: TILES, and InfoMap.
3. **Performance Comparison of Hierarchical and Dynamic Community Detection Algorithms**: To evaluate the performance of the chosen dynamic algorithms to each other and to hierarchical methods, we incorporate various evaluation metrics such as the proposed Rand Index modification, the standard Rand Index and Normalized Mutual Information (NMI). We will also look into other criteria such as Modularity and Closeness Centrality.

5.1 Modified Rand Index

To give premise to our approach in modifying the Rand Index, we make use of the definition provided by Bakker et al. (2018) as explained in our review of related literature. We give a more formal definition of the Rand Index formula in the context of Gene Co-expression Networks.

5.1.1 Limitations of Applying Rand Index to Multi-Source Datasets. Let $G = \{v_1, v_2, ..., v_i\}$ be a network of i nodes, and let A, B be two partitions of G into j and k communities, respectively. We define $P = \{(v_x, v_y) \mid x \neq y, \text{ and } v_x, v_y \in G\}$, the set of all unique node pairs in G. For any two nodes $v_x, v_y \in G$, we define the community membership relation as follows:

$$v_x \sim_X v_y \iff v_x \text{ and } v_y \text{ belong to the same community in partition } X$$

$$v_x \not\sim_X v_y \iff v_x \text{ and } v_y \text{ belong to different communities in partition } X$$

Thus, we define n_{00}, n_{11}, n_{01}, and n_{10} as follows:

- $n_{00} = |\{(v_x, v_y) \in P \mid v_x \not\sim_A v_y \text{ and } v_x \not\sim_B v_y\}|$, the number of node pairs from a network G which are in different communities in A and different communities B.
- $n_{11} = |\{(v_x, v_y) \in P \mid v_x \sim_A v_y \text{ and } v_x \sim_B v_y\}|$ is the number of node pairs from a network G which are in the same community in A and in the same community in B.
- $n_{01} = |\{(v_x, v_y) \in P \mid v_x \not\sim_A v_y \text{ and } v_x \sim_B v_y\}|$ is the number of node pairs from a network G which are in different communities in A but in the same community in B.
- $n_{10} = |\{(v_x, v_y) \in P \mid v_x \sim_A v_y \text{ and } v_x \not\sim_B v_y\}|$ is the number of node pairs from a network G which are in the same community in A but in different communities in B.

For a node pair p, if $p \in n_{00}$ or $p \in n_{11}$ then p is an "agreeing pair." Conversely, if $p \in n_{01}$ or $p \in n_{10}$, then p is a "disagreeing pair."

The Rand Index is then expressed as the proportion of agreeing pairs over all possible pairs:

$$Rand(A, B) = \frac{n_{00} + n_{11}}{n_{00} + n_{11} + n_{01} + n_{10}} = \frac{n_{00} + n_{11}}{\binom{i}{2}}$$

However, the assumption of the Bakker et al. (2018) Rand Index is that there is a single source of truth upon which the two partitions are derived. That is, a single network G is partitioned into A and B. However, in Calderon et al. (2024), the Adjusted Rand Index, which implicitly makes use of Rand Index as a subroutine, is applied to two sources of truth: the Cho dataset and the KEGG dataset.

Specifically, let C represent a partition derived from the Cho dataset and K a partition derived from the KEGG dataset. Since the Cho dataset does not contain all genes present in the KEGG dataset, inconsistencies arise when using $Rand(C, K)$. Some node pairs exist in K but not in C, creating a mismatch where:

$$p \in P_{null} = \{(v_c, v_k) \mid v_c \in C, v_k \in K, v_c \notin K\} \implies p \notin P.$$

This inconsistency violates the fundamental assumption of the Rand Index, which requires both partitions to operate over the same set of elements. Thus, when the Rand Index is calculated, the problematic pairs in P_{null} contribute to the denominator of the $Rand(C, K)$, leading to significantly weak scores. Considering that the Cho dataset is significantly more outdated than the KEGG dataset (Cho et al., 1998; Kanehisa et al., 2022), this could be a probable cause to the weak ARI scores in the testing of hierarchical community detection by Calderon et al. (2024) .

Consequently, we propose a modified approach to address this limitation and handle the node pairs that are problematic.

5.1.2 Designing the Modified Rand Index. Using the original formulation of the Rand Index as our basis, we make some redefinitions to optimize this metric for our unique multi-source dataset.

To adapt the Rand Index for comparing partitions derived from multi-source datasets, we incorporate an additional component, n_{xx}, into its formulation. The Modified Rand Index (MRI) is designed to account for new or missing pairs resulting from the differences in the scope and scale of datasets, such as the Cho dataset and the more comprehensive KEGG database.

MRI assumes two sets of communities A and B, where A and B originate from two distinct networks. Let $G = G_A \cup G_B$, where G_A is the set of nodes associated with A and G_B the set of nodes associated with B.

We define the pair set P_{MRI} which consists of all unique pairs of nodes in G_{MRI}:

$$P_{MRI} = \{(v_x, v_y) \mid x \neq y, \text{ and } v_x, v_y \in G\}$$

To ensure that n_{11}, n_{00}, n_{01}, and n_{10} account only for pairs where both nodes exist in $G_A \cap G_B$, we redefine these components as follows:

$$n_{00} = |\{(v_x, v_y) \in P_{MRI} \mid v_x, v_y \in G_A \cap G_B \text{ and } v_x \not\sim_A v_y \text{ and } v_x \not\sim_B v_y\}|$$

n_{00} represents the number of pairs where both nodes exist in $G_A \cap G_B$, and the nodes are in different communities in both partitions A and B.

$$n_{11} = |\{(v_x, v_y) \in P_{MRI} \mid v_x, v_y \in G_A \cap G_B \text{ and } v_x \sim_A v_y \text{ and } v_x \sim_B v_y\}|$$

n_{11} represents the number of pairs where both nodes exist in $G_A \cap G_B$, and the nodes are in the same community in both partitions A and B.

$$n_{01} = |\{(v_x, v_y) \in P_{MRI} \mid v_x, v_y \in G_A \cap G_B \text{ and } v_x \not\sim_A v_y \text{ and } v_x \sim_B v_y\}|$$

n_{01} represents the number of pairs where both nodes exist in $G_A \cap G_B$, and the nodes are in different communities in A but the same community in B.

$$n_{10} = |\{(v_x, v_y) \in P_{MRI} \mid v_x, v_y \in G_A \cap G_B \text{ and } v_x \sim_A v_y \text{ and } v_x \not\sim_B v_y\}|$$

n_{10} represents the number of pairs where both nodes exist in $G_A \cap G_B$, and the nodes are in the same community in A but different communities in B.

We define a new component, n_{xx}, as follows:

$$n_{xx} = |\{(v_x, v_y) \in P_{MRI} \mid v_x \notin G_A \text{ or } v_y \notin G_A \text{ or } v_x \notin G_B \text{ or } v_y \notin G_B\}|$$

n_{xx} represents the number of pairs where at least one node in the pair does not exist in one of the partitions, ensuring that pairs missing from either A or B are appropriately categorized.

This new component captures all pairs involving nodes that are missing in one of the two networks. The inclusion of n_{xx} ensures that the MRI remains valid even when comparing datasets with partially overlapping sets of nodes. The rationale for incorporating this is to allow the metric to handle emerging pairs brought about by the updated KEGG dataset. Thus, our final MRI formula is as follows:

$$MRI(A, B) = \frac{n_{00} + n_{11} + n_{xx}}{n_{00} + n_{11} + n_{01} + n_{10} + n_{xx}} = \frac{n_{00} + n_{11} + n_{xx}}{|P_{MRI}|}$$

5.1.3 Testing The Modified Rand Index on Multi-source Datasets. To validate the applicability and effectiveness of the Modified Rand Index (MRI), we conduct a series of tests and comparisons to evaluate its performance across various scenarios. This process ensures that the metric reliably captures the agreement between partitions while addressing the challenges posed by multi-source datasets.

1. Testing with Sample Partitions: To validate the scoring behavior of MRI, we create synthetic sample partitions to simulate various alignment scenarios. These tests establish baseline expectations for the MRI's scoring behavior and its ability to handle real-world data complexities:

- **Ideal Case:** Generate two partitions with perfect agreement. MRI should yield a score of 100%, as all node pairs align perfectly in both partitions.
- **Generalized Case:** Generate two random partitions, simulating real-world scenarios where datasets originate from distinct sources of truth. This case evaluates the MRI's ability to provide meaningful scores when agreement is partial or ambiguous.
- **Non-Ideal Case:** Generate two partitions with drastically varying levels of disagreement. MRI should yield scores proportionally lower, reflecting the reduced alignment between partitions.
- **Mixed Case:** Create a scenario where one partition contains missing or additional nodes not present in the other. This case tests MRI's ability to accommodate n_{xx} and assess its impact on the overall score, demonstrating the metric's robustness in handling dataset mismatches.

2. Application to Cho and KEGG Datasets: The MRI is then applied the Cho and KEGG datasets. This test evaluates how well the new metric accommodates the differences between these datasets, particularly the varying number of genes and functional groupings. For comparison, the RI is also computed for Cho and KEGG. An analysis is made between the scores of MRI and RI. It should be noted that we cannot directly compare MRI with the ARI scores as yielded by Calderon et al. (2024) in their testing of the Cho and KEGG datasets. Since ARI is an adjusted metric, a more fairer comparison would be between the standard RI and the MRI.

5.2 Dynamic Community Detection for Predicting Functional Classifications

This section outlines the methodology for applying Dynamic Community Detection (DCD) algorithms to the gene co-expression networks derived from the Cho and KEGG datasets. The goal is to explore communities that align with known functional classifications, providing insights into the organization of the gene co-expression network and predicting functional groupings where no prior classification exists.

5.2.1 Forming the Gene Co-expression Network. The first step involves defining the structural properties of the gene co-expression network (GCN) to ensure compatibility with dynamic community detection algorithms (Calderon et al., 2024; Clemente et al., 2022; Redekar and Varma, 2022):

- Edge-Weighted and Undirected: The network will represent relationships between genes using weighted edges based on their co-expression values. The undirected nature reflects mutual interactions without hierarchical directionality.
- Temporal Representation: The network will be modeled dynamically across the different time points in the datasets, capturing changes in co-expression relationships over time.

Using these properties, we form the GCNs for both the Cho and KEGG datasets through a value-based algorithm for network construction (Calderon et al., 2024; Clemente et al., 2022):

- **Correlation-Based Edge Weights:** Pairwise gene co-expression relationships are quantified using the Pearson Correlation Coefficient (PCC) to measure the linear relationship between gene expression profiles across time points.
- **Thresholding Correlation Values:** A threshold value ($\delta = 0.85$) is applied to PCC scores to determine whether an edge exists between two genes. An edge is created only if the correlation exceeds this threshold, ensuring that only strongly co-expressed genes are connected. According to Calderon et al. (2024), this threshold yielded the best results in previous studies by prioritizing biologically meaningful interactions.
- **Forming GCNs for Cho and KEGG:** Individual GCNs are constructed for the Cho and KEGG datasets. These networks are weighted and undirected, reflecting the mutual interactions between genes. The use of the same construction method across both datasets ensures consistency, facilitating comparisons between the networks and their detected communities. This approach allows for the construction of biologically relevant co-expression networks that serve as the foundation for dynamic community detection and subsequent functional classification.

5.2.2 Applying Dynamic Community Detection. To analyze the dynamic nature of the co-expression network, we identify and implement suitable DCD algorithms that can handle *weighted and undirected networks*. According to Calderon et al. (2024), the following algorithms considered include:

- InfoMap: It is an algorithm that utilizes a map equation to detect community structures in networks and employs the concept of minimum description length in optimally computing partitions. An optimal partition has a good modular structure, which occurs when its description length gets compressed when a random walker stays longer within it.
- TILES: It is an algorithm that is able to detect and track the evolution of overlapping communities in dynamic networks and works similarly to the fall of a domino tile. Its property of compositionality allows computations to be parallelizable, thus resulting to faster computations of communities.

Using the selected DCD algorithms, the community structures within the gene co-expression network will be computed for both the Cho and KEGG datasets.

5.3 Performance Comparison of Hierarchical and Dynamic Community Detection Algorithms

This section outlines the metrics for comparing dynamic and hierarchical community detection algorithms. The goal is to compare the performance of the chosen DCD algorithms to each other and to the hierarchical algorithms used by Calderon et al. (2024) to determine which methods are more suitable in deriving biologically significant results from applying these algorithms to GCNs.

5.3.1 Internal Criteria. To assess the clustering results of the community detection algorithms without referring to any external information such as ground truth data, we identify and employ the following metrics:

- Modularity: It is a metric for measuring the quality of a network's community structure. A high modularity indicates that connections between vertices within the same community are denser than the connections between vertices across different communities.
- Closeness Centrality: It is a metric used to gauge the transmission rate of information to a node from all other nodes in a network. It evaluates the accessibility of a node and its ability to quickly obtain information.

The internal metrics that will be implemented for this study are publicly available from the NetworkX library for Python.

5.3.2 External Criteria. To assess the clustering results of the community detection algorithms compared to external information such as ground truth data, we identify and employ the following metrics:

- **Rand Index (RI):** This metric evaluates the agreement between two partitions by examining how node pairs are classified as belonging to the same or different communities. Scores range from 0 to 1, where 1 indicates perfect agreement.
- **Modified Rand Index (MRI):** A modified version of the Rand Index that accounts for pairs involving nodes missing in one of the partitions. This adjustment improves the metric's robustness when comparing datasets with differing scopes, such as Cho and KEGG.
- **Normalized Mutual Information (NMI):** This metric measures the similarity between two partitions by quantifying their shared information. The score ranges from 0 to 1, with 1 indicating identical partitions and 0 indicating no similarity.

The RI and NMI metrics that will be implemented for this study are publicly available from the Scikit-learn library for Python. The workings and implementation of the Modified Rand Index (MRI) metric is discussed in section 5.1.

6 Schedule of Activities

To keep track of our timeline as we conduct our methodology, we construct the Gantt Chart in Figure 1.

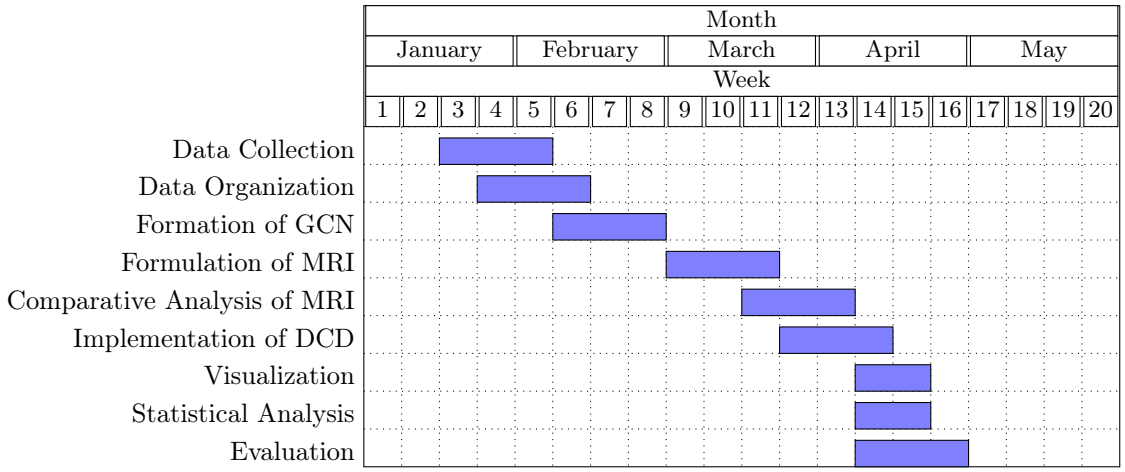

Fig. 1. Timeline for Methodology

References

Bakker, C., Halappanavar, M., & Visweswara Sathanur, A. (2018). Dynamic graphs, community detection, and riemannian geometry. *Applied Network Science*, *3*(1), 3. https://doi.org/10.1007/s41109-018-0059-2

Barabási, A.-L., & Oltvai, Z. N. (2004). Network biology: Understanding the cell's functional organization. *Nature Reviews Genetics*, *5*(2), 101–113. https://doi.org/10.1038/nrg1272

Calderon, J. K., Ventures, P. A., & Clemente, J. (2024). Hierarchical community detection on co-expression networks for functional classification of cell-cycle regulated genes of saccharomyces cerevisiae. *Algorithms and Complexity Lab, University of the Philippines Diliman.*

Chejara, P., & Godfrey, W. W. (2017). Comparative analysis of community detection algorithms. *2017 Conference on Information and Communication Technology (CICT)*, 1–5. https://doi.org/10.1109/INFOCOMTECH.2017.8340627

Cho, R. J., Campbell, M. J., Winzeler, E. A., Steinmetz, L., Conway, A., Wodicka, L., Wolfsberg, T. G., Gabrielian, A. E., Landsman, D., Lockhart, D. J., & Davis, R. W. (1998). A genome-wide transcriptional analysis of the mitotic cell cycle. *Molecular Cell*, *2*(1), 65–73. https://doi.org/10.1016/s1097-2765(00)80114-8

Clemente, J. B., Besas, G., Callado, J., & Evangelista, J. E. (2022). Predicting the biological classification of cell-cycle regulated genes of saccharomyces cerevisiae using community detection algorithms on gene co-expression networks. https://doi.org/10.48550/ARXIV.2208.10119

Dorogovtsev, S. N., & Mendes, J. F. F. (2002). Evolution of networks. *Advances in Physics*, *51*(4), 1079–1187. https://doi.org/10.1080/00018730110112519

Girvan, M., & Newman, M. E. J. (2002). Community structure in social and biological networks. *Proceedings of the National Academy of Sciences*, *99*(12), 7821–7826. https://doi.org/10.1073/pnas.122653799

Held, P., Krause, B., & Kruse, R. (2016). Dynamic clustering in social networks using louvain and infomap method. *2016 Third European Network Intelligence Conference (ENIC)*, 61–68. https://doi.org/10.1109/ENIC.2016.017

Kanehisa, M., Furumichi, M., Sato, Y., Kawashima, M., & Ishiguro-Watanabe, M. (2022). KEGG for taxonomy-based analysis of pathways and genomes. *Nucleic Acids Research*, *51*(D1), D587–D592. https://doi.org/10.1093/nar/gkac963

Kherad, M., Dadras, M., & Mokhtari, M. (2024). Community detection based on influential nodes in dynamic networks. *The Journal of Supercomputing*, *80*(16), 24664–24688. https://doi.org/10.1007/s11227-024-06367-4

Lau, L. Y., Reverter, A., Hudson, N. J., Naval-Sanchez, M., Fortes, M. R. S., & Alexandre, P. A. (2020). Dynamics of gene co-expression networks in time-series data: A case study in drosophila melanogaster embryogenesis. *Frontiers in Genetics*, *11*. https://doi.org/10.3389/fgene.2020.00517

Liu, X., Cheng, H.-M., & Zhang, Z.-Y. (2020). Evaluation of Community Detection Methods. *IEEE Transactions on Knowledge & Data Engineering*, *32*(09), 1736–1746. https://doi.org/10.1109/TKDE.2019.2911943

Ma, X., & Dong, D. (2017). Evolutionary nonnegative matrix factorization algorithms for community detection in dynamic networks. *IEEE Transactions on Knowledge and Data Engineering*, *29*(5), 1045–1058. https://doi.org/10.1109/TKDE.2017.2657752

Márquez, R., & Weber, R. (2023). Dynamic community detection including node attributes. *Expert Systems with Applications*, *223*, 119791. https://doi.org/https://doi.org/10.1016/j.eswa.2023.119791

Ovens, K., Eames, B. F., & McQuillan, I. (2021). Comparative analyses of gene co-expression networks: Implementations and applications in the study of evolution. *Frontiers in Genetics*, *12*. https://doi.org/10.3389/fgene.2021.695399

Parapouli, M., Vasileiadi, A., Afendra, A.-S., & Hatziloukas, E. (2020). Saccharomyces cerevisiae and its industrial applications. *AIMS Microbiology*, *6*(1), 1–32. https://doi.org/10.3934/microbiol.2020001

Priya, A., Sharma, S., Sinha, K., & Yogesh, Y. (2023). Community detection in networks: A comparative study. *2023 International Conference on Device Intelligence, Computing and Communication Technologies, (DICCT)*, 505–510. https://doi.org/10.1109/DICCT56244.2023.10110206

Rahiminejad, S., Maurya, M. R., & Subramaniam, S. (2019). Topological and functional comparison of community detection algorithms in biological networks. *BMC Bioinformatics*, *20*(1), 212. https://doi.org/10.1186/s12859-019-2746-0

Redekar, S. S., & Varma, S. L. (2022). A survey on community detection methods and its application in biological network, 1030–1037. https://doi.org/10.1109/ICAAIC53929.2022.9792913

Rossetti, G., & Cazabet, R. (2018). Community discovery in dynamic networks: A survey. *ACM Comput. Surv.*, *51*(2). https://doi.org/10.1145/3172867

Rossetti, G., Pappalardo, L., Pedreschi, D., & Giannotti, F. (2017). Tiles: An online algorithm for community discovery in dynamic social networks. *Machine Learning*, *106*(8), 1213–1241. https://doi.org/10.1007/s10994-016-5582-8

Sattar, N. S., Buluc, A., Ibrahim, K. Z., & Arifuzzaman, S. (2023). Exploring temporal community evolution: Algorithmic approaches and parallel optimization for dynamic community detection. *Applied Network Science*, *8*(1), 64. https://doi.org/10.1007/s41109-023-00592-1

Seifikar, M., Farzi, S., & Barati, M. (2020). C-blondel: An efficient louvain-based dynamic community detection algorithm. *IEEE Transactions on Computational Social Systems*, *7*(2), 308–318. https://doi.org/10.1109/TCSS.2020.2964197

Singh, D. K., Haraty, R. A., Debnath, N. C., & Choudhury, P. (2020). An analysis of the dynamic community detection algorithms in complex networks. *2020 IEEE International Conference on Industrial Technology (ICIT)*, 989–994. https://doi.org/10.1109/ICIT45562.2020.9067224

Stuart, J. M., Segal, E., Koller, D., & Kim, S. K. (2003). A gene-coexpression network for global discovery of conserved genetic modules. *Science*, *302*(5643), 249–255. https://doi.org/10.1126/science.1087447

Sun, Z., Wang, B., Sheng, J., Yu, Z., Zhou, R., & Shao, J. (2019). Community detection based on information dynamics. *Neurocomputing*, *359*, 341–352. https://doi.org/https://doi.org/10.1016/j.neucom.2019.06.020

Yeung, K. Y., Haynor, D. R., & Ruzzo, W. L. (2001). Validating clustering for gene expression data. *Bioinformatics*, *17*(4), 309–318. https://doi.org/10.1093/bioinformatics/17.4.309

Yu, D., Kim, M., Xiao, G., & Hwang, T. H. (2013). Review of biological network data and its applications. *Genomics Inform*, *11*(4), 200–210. https://doi.org/10.5808/GI.2013.11.4.200

Zhuang, D., Chang, J. M., & Li, M. (2021). Dynamo: Dynamic community detection by incrementally maximizing modularity. *IEEE Transactions on Knowledge and Data Engineering*, *33*(5), 1934–1945. https://doi.org/10.1109/TKDE.2019.2951419