AfricanJournalofBiological

Sciences

Applicationsof the Domination and Fractional Domination in Computational Biology using LPP Formulation

Mahesh Sarada[1,2], Rekha Jain[1], Ganesh Mundhe[3]

Department of Mathematics, Medi Caps University, Rau, Indore, Madhya Pradesh, India[1]

Pimpri Chinchwad College of Engineering & Research, Ravet, Pune, Maharashtra, India[2]

Army Institute of Technology, Dighi, Pune, Maharashtra, India[3]

Email: mahesh.sarada@gmail.com

rjain5129@gmail.com

ganumundhe@gmail.com

Article History
Volume 6, Issue 5, 2024
Received: 09 May 2024
Accepted: 17 May 2024
doi:*10.33472/AFJBS.6.5.2024. 4340-4358*

Abstract

The main goal of this study is utilizing domination and fractional domination in computational biology. The domination number of a graph is the size of the smallest domination set. The collection of vertices with non-negative weights is known as a fractional domination set such that the sum of the weights of the vertices and their neighbours is at least one. In the given study our proposed approach can be used to apply the domination and fractional domination concepts for connected graph using adjacency matrix and LPP formulation. Here we have used these conceptsin computational biological systems such as gene regulatory networks, protein-protein interaction networks and healthcare network optimization. Computational biology is an interdisciplinary field that applies computational techniques and mathematical models to analyse and interpret biological data.

Keywords: Dominating set, fractional domination number, adjacency matrix, computational biological network, LPP formulation.

I. Introduction

Let $G = (V, E)$is graph with V as vertex set and E as edge set. A subset $D$of $V$is called a dominating set of graphs Gif eachvertex in $V - D$is connected to at least one vertex in D. For every vertex $u \in V - D$, $d(u, D) = 1$ or $N[D] = V$. A domination set $D$is known as minimaldominating set (*MDS*) if no proper subset of $D$is a dominating set of G. The size of smallest dominating set of graphs $G$is called the domination number of graph G and it is denoted by γ (G).The maximum cardinality of minimal dominating set (*MDS*) of graph $G$is called upper dominationnumber of $G$and it is denoted by $\Gamma(G)$.Thedominationnumber noted by$\gamma(G)$andtheupperdominationnumber noted by$\Gamma(G)$ aredefinedas: γ $(G) = min\{|D|: DisMDSofG\}and\Gamma(G) = \max\{|D|: DisMDSofG\}$.

A dominating function $f$to be any$f: V(G) \rightarrow [0, 1]$ is function of G which allocates the values for each vertex $v \in V(G)$ in the unit interval [0,1]. The given function f is called a fractional dominating function if for every vertex $v \in V(G), f(N[v])$ = $\sum_{v \in N[v]} f(v) \geq 1$. It denotes the total value of the vertices in the closed neighbourhood of $v \in V(G)$such that$N[v]$ is at least one, i.e. $(\sum N[v]) \geq 1$. (Since then any vertex $v \in V(G)$is in the closed neighbourhood of at least one vertex in D, where D is subset of vertex set V).

The dominating function $f$is called a minimal fractional dominating function (*MFDF*) if there does not exist a dominating function $g \neq f$for which$g(v) \leq f(v)$for all$v \in V(G)$equivalently $f$is an minimal fractional dominating function (*MFDF*) if for every vertex $v$with$f(v) >$ 0,there exist a vertex$w \in N[v]$such that$\sum_{v \in N[w]} f(v) = 1$.

If there is a vertex v for which given condition is not true meansevery vertex in the closed neighborhood of v obeys $(N[v]) > 1$, then we can decrease $f(v)$ to obtain a smaller fractional dominating function and so f is not a minimal fractional dominating function. The fractional domination number of G denoted byγ_f $(G)$and the upper fractional domination number of G denoted by Γ_f $(G)$are definedas, γ_f (G) $= min$ $\{|f|: f\ is\ an\ MFDF\ G\}$,

Γ_f $(G) = max$ $\{|f|: f\ is\ an\ MFDF\ G\}$, Where $|f|$ = $\sum_{v \in V} f(v)$.

Example Figure 1: The Hajos graph H_3 , (γ_f (G) = 3/2, γ (G) = 2, Γ $(G) = \Gamma_f$ (G) = 3)

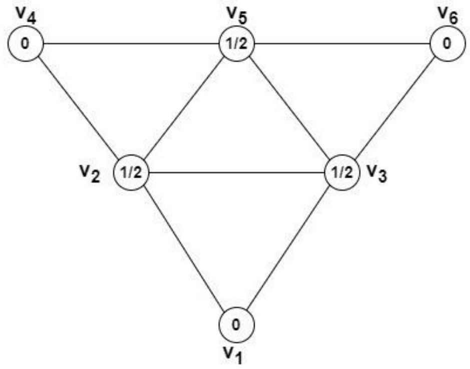


Figure 1

At the Eighteenth Southeastern International conference, graph theory and computing, S.T. Hedetniemi formally defined fractional domination. For the Hajos graph H_3 in Figure 1, let $f(V_2)$ = $f(V_3)$ =$f(V_5)$ = 1/2 and $f(V_1)$ = $f(V_4)$ = $f(V_6)$ = 0 which makesγ_f (G) = 3/2.

Fractional domination number as 1 be the vector of all ones. Let 0 be the vector of all zeros. Let $A(G)$be the adjacency matrix. The fractional domination number γ_f (G) is the value of linear program objective function minimizes $1^T x$, where 1^T is row vector of once and x is column vector of decision variables. Multiplying 1^T by x essentially sums up all the elements of x. subject to conditions$(A(G) + I)x \geq 1$, This represents a set of linear inequality constraints where$A(G)$be the adjacency matrix and I is the identity matrix and$x$ is the decision variable vector. $(A(G) + I)x$it creates a new vector to which each element is at least 1. This constraint essentially ensures that the result of multiplying $(A(G) + I)x$is greater than or equal to 1 for each elementand $x \geq 0$this constraint simply states that each decision variable must be non-negative.

In 1980 Cockayne, Dawes, and Hedetniemi presented the idea and has since been extensively studied in graph theory and related fields.One of the early results in the area is a characterization of graphs with fractional domination number 1, which is equivalent to the domination number being 1. This result is due to Haynes, Hedetniemi, and Slater in 1998, who also proved that computing the fractional domination number is NP-hard.

In 1985, Fink and Jacobson introduced generalization of the concept of domination and independence in graphs. Further in the paper of [2] they provide a survey result on k-domination and k-independence in graphs. They also established a number of bounds and results for this variant of the problem.The authors of the paper [9] has demonstrated a linear time approach for the k-domination issue on networks where every block represents a full bipartite graph, a clique, or a cycle.All graphs include trees, block graphs, cacti and block-cactus graphs. The authors of the paper [11] provided a polynomial-time approach for determining each nontrivial tree's dominator chromatic number.The observations in [5] They offer a technique for determining the maximum n-independent set$S$ and the total n-dominating set Din a connected graph with at least $p \geq 2n + 1$ vertices where p is the order of graph.

The fractional domination number has also been studied in several graph classifications, including trees, planar graph, and hypercubes. In particular, research has demonstrated that the fractional dominance number of a tree is always at most one. In recent years, researchers have also investigated fractional versions of other graph parameters, such as the independent domination number and the total domination number. There has also been interest in the computational complexity of computing these parameters, as well as in algorithms for finding them in various types of graphs. Overall, the study of fractional domination numbers and research on related parameters is also on going in the field of graph theory.

The author of [6] has given relation between fractional dominating number of G and fractional total dominating number of graph complement. They have defined fractional domination number as 1 is the vector of all ones and 0 is the vector of all zeros. Now$A(G)$ is adjacency matrix of graph and I be the identity matrix. The fractional dominating number $\gamma_f\ (G)$is objective function value of linear programming problem:
Objective function minimizes$1^T x$,
subject to constraints$(A(G) + I)x \geq 1$,and $x \geq 0$.(1)
Same like this the fractional total dominating number $\Gamma_f\ (G)$ be the value of linear programming problem of objective function minimize $1^T x$,
subject toconstraints$(A(G)x) \geq 1$, and$x \geq 0$. (2)
Solution to equation (1) is fractional dominating number with non-negative weights on vertices whose sum in any closed neighbourhood is at least one. By forcing x to have integer entries transforms equation (1) into integer program for domination number of graph.Solution to equation (2) is fractional total dominating number with non-negative weights on vertices whose sum in any open neighbourhood is at least one. By forcing x to have integer entries transforms equation (2) into integer program for upper domination number of graph.
Authors of [12] considered the problem of incrementally computing a minimal dominating set of a directed graph after the insertion or deletion of a set of arcs.In this paper, they first show how to incrementally compute a minimal dominating set on arc insertions and then reduce the case of computing a minimal dominating set on arc deletions to the case of insertions.

The second section includesgraph representation with adjacency matrix and LPPformulation. The third section contains applications of domination and fractional domination in computational biological systems. Specifically, gene regulatory networks, protein-protein interaction network, Healthcare Network Optimization.

Somepreliminaryknownresults:

Theorem1.1[7]*Let for any graph G we have* $\left\lceil \frac{n}{1+\Delta(G)} \right\rceil \leq \gamma \ \ (G) \leq n - \Delta(G)$.

Theorem1.2[7]*Let for anygraph G we have* $\frac{n}{1+\Delta(G)} \leq \gamma_f\ (G) \leq \frac{n}{1+\delta(G)}$,

Where $\Delta(G)$ *is maximum degree of graph* and$\delta(G)$ *is minimum degree of graph.*

Theorem1.3[7]*Let for anygraphGwehave*$\gamma_f\ (G) = 1$*if and only if*$\Delta(G) = n - 1$.

Theorem 1.4 [4] *If G is an r-regular graph then* $\gamma_f\ (G) = \frac{n}{r+1}$.

Some results of our work quoted as [13]:

1. If *G is the cycle graph of order n and G′its dual graph, then*$\gamma_f\ (G) + \gamma_f\ (G') = 1 + n/3$ *and* $\gamma_f\ (G) * \gamma_f\ (G') = n/3$.For cycle graph **(**$C_n$**)**where n = 3, 4, 5,..., n.The fractional domination number is$\gamma_f\ (C_n) = n/3$.
2. *If G be the wheel graph of order n ≥ 4 and G′its dual graph, for then*

   $\gamma_f\ (G) + \gamma_f\ (G') = 2$ *and* $\gamma_f\ (G) * \gamma_f\ (G') = 1$.
3. *If G is the completegraph with vertices n < 5 and G′ be its dual graph then*$\gamma_f\ (G) + \gamma_f\ (G') = 2$ *and* $\gamma_f\ (G) * \gamma_f\ (G') = 1$.
4. *If G be the Bi-Star graph of order n and G′its dual graph, then*$\gamma_f\ (G) + \gamma_f\ (G') = 3$ *and* $\gamma_f\ (G) * \gamma_f\ (G') = 2$.
5. *If G be the n-Sunlet graph and G′its dual graph, then*$\gamma_f\ (G) + \gamma_f\ (G') = (n+1)$ *and* $\gamma_f\ (G) * \gamma_f\ (G') = n$.
6. *If G be the graph of Cartesian product* **(**$K_2 \times P_n$**)** *with n >1 then*

   $$\gamma_f(K_2 \times P_n) = \begin{cases} \frac{n+1}{2}, & if n is odd \\ \frac{(n^2+2n)}{2(n+1)}, & if n is even \end{cases}$$
7. *If G be the graph of Cartesian product* **(**$K_2 \times P_n$**)** *for n >1 and G′ be its dual graph then* $\gamma_f\ (G) + \gamma_f\ (G') = \begin{cases} 1 + \frac{n+1}{2}, & if n\ is\ odd \\ 1 + \frac{(n^2+2n)}{2(n+1)}, & if\ n\ is\ even \end{cases}$

8. *If G is the graph of Cartesian product* **(**$K_2 \times C_n$**)** *with 2n vertices n > 2 then,*γ_f $(K_2 \times C_n) = 2n/4$
9. *If G be the graph of Cartesian product* **(**$K_2 \times C_n$**)** *with 2n vertices n > 2 and* (G') *be its dual graph with 2n-m vertices where m = 1, 2,..., mthen*

$$\gamma_f(G) + \gamma_f(G') = \begin{cases} \frac{2n}{4} + \frac{2n-m}{4+m}, & if n = 3\ and m = 1 \\ \frac{2n}{4} + \frac{(2n-2)}{(4+m-1)} & if n = 4,5,6 \ldots\ and m = 2,3,4 \ldots \end{cases}$$

II. Graph representation with adjacency matrix and LPP formulation

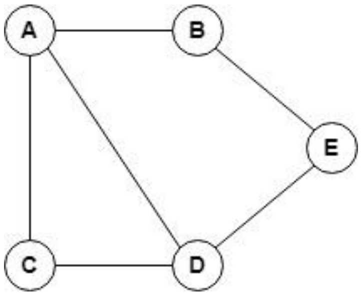
2.1A graph is a non-linear data structure:

A set of nodes, also known as vertices and the edges that join two or more vertices together make up a graph. A graph can also be thought of as a cyclic tree, in which the vertices maintain a complex interaction between one another but do not have a parent-child relationship.

Graph algorithmis a collection of instructions called a graph algorithm connected to each node in a graph. The path between two specified nodes or a single node can be found using some algorithms.The method used to store a graph data structure in memory is called graph representation. Either a linked form or a sequential representation can be used to store the graph. One of three data structures, such as an adjacency matrix, adjacency list, or adjacency set, can be used to represent a graph. The descriptions of these two kinds follow.

2.2Sequential representation:The adjacency matrix is used in graph sequential representation. A matrix with size $n\ x\ n$, where n is the number of vertices in the network, is called an adjacency matrix. The adjacency matrix's rows and columns correspond to a graph's vertices. When there is an edge connecting the vertices, the matrix element is set to 1. The element is set to 0 if the edge is absent.

Given below is an example of graph that shows its adjacency matrix.



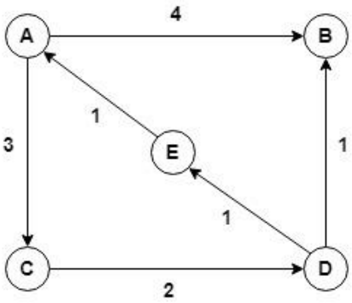
$$A = \begin{array}{c} \begin{array}{ccccc} A & B & C & D & E \end{array} \\ \begin{bmatrix} 0 & 1 & 1 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 \end{bmatrix} \end{array}$$

(Undirected Graph) Adjacency Matrix

Figure 2

Here rows and columns are taken sequentially like (A, B, C, D, E) so that matrix is of order ($\mathbf{5 \times 5}$) matrix.The adjacency matrix for the graph above has been shown to us. Observe that the edge is present in both directions because this graph is undirected. For instance, we can infer that edge BA is present since edge AB is present.

The interactions of the vertices, which are matrix entries set to 1 when the edge is present and to 0 when the edge is absent, are displayed in the adjacency matrix. The weighted graph and associated adjacency matrix are shown below.



(Weighted Graph)

$$\begin{array}{cc} & \begin{array}{ccccc} A & B & C & D & E \end{array} \\ A = & \begin{bmatrix} 0 & 4 & 3 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 2 & 0 \\ 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 \end{bmatrix} \end{array}$$

Adjacency Matrix

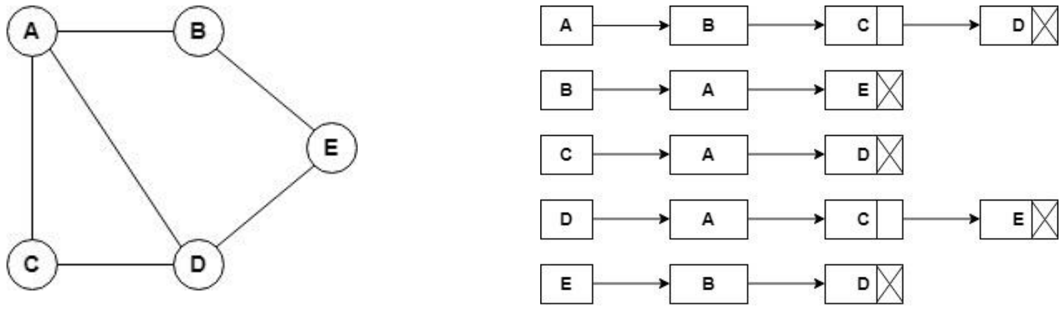
Figure 3

It is evident that a weighted graph's sequential representation differs from that of other graph types. In this case, the weight of the edge itself takes the place of the non-zero values in the adjacency matrix. Since the weight of the edge AB is 4, we set the intersection of A and B to 4 in the adjacency matrix.

In a similar manner, the weights of all the other non-zero values are adjusted. The adjacency list is simpler to use. Traversal means to check if there is an edge from one vertex to another takes O(1) time and removing an edge also takes O(1). A dense graph always requires more space than a sparse graph, which has fewer edges.

2.3Linked representation

For the graph's linked representation, we employ the adjacency list. Every node in the graph is preserved together with a link to the nodes that are next to it thanks to the adjacency list representation. We set the next pointer to null at the end of the list after we have traversed all of the neighbouring nodes. Initially, let us examine an undirected graph along with its adjacency list.

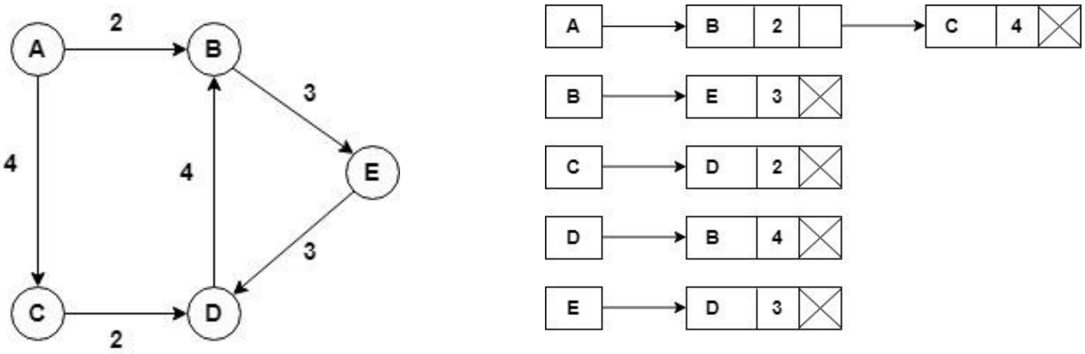


(Undirected Graph)

Adjacency nodes

Figure 4

Every node has a linked list, also known as an adjacency list, as demonstrated above. We have edges to vertices B, C, and D from vertex A. In the associated adjacency list, node A is linked to these nodes as a result. Let's now build the weighted graph's adjacency list.



(Weighted Graph)

Adjacency List

Figure 5

To indicate the weight of an edge in a weighted graph, we include an additional field in the adjacency list node, as demonstrated above. It is simpler to add a vertex to the adjacency list. Because of the linked list implementation, it also conserves space. The operation is inefficient when we need to determine whether there is an edge connecting two vertices. The adjacency list for a weighted directed graph will be shown here.

The edges of the graph and the adjacency list are stored in two different structures. The adjacency list is displayed like as (start-vertex, end-vertex, weight). We will get the output as Graph adjacency listwith (Start-vertex, end-vertex, weight):

(0, 2, 4) (0, 1, 2)(1, 4, 3)(2, 3, 2)(3, 1, 4)(4, 3, 3)

2.4LPP formulation of given graph:

1) Let $G = (V, E)$be an undirected graph with$V$ is set of verticesand E is set of edges.

Decision variable $f(v) = x_v$, binary decision variable indicating whether vertex v is dominated. $f(v) = x_v = 1$ if vertex v is dominated and $f(v) = x_v = 0$ otherwise.

Objective function: Minimize the total number of vertices dominated so

Minimize: $\sum_{v\in V} x_v$

Constraints: i) Every vertex v must be dominated $\sum_{v\in N[v]} f(v) \geq 1$ for all $v \in V(G)$ where $N[v]$ is the closed neighborhood of vertex v it means set of vertices adjacent to v.

ii) Binary decision variables $0 \leq f(v) \leq 1$ for all $v \in V(G)$

with these definitions the LPP representation for the fractional domination number of graph is

Minimize: $\sum_{v\in V} x_v$

Subject to i) $\sum_{v\in N[v]} x_v \geq 1$ for all $v \in V(G)$

ii) $0 \leq x_v \leq 1$ for all $v \in V(G)$

This LPP formulation aims to minimize the total number of vertices that are dominated subject to the constraints that every vertex must be dominated and the decision variables must be binary. Solving this LPP provides the fractional dominating number of graph.

2) Objective function:Minimize $Z = c_1x_1 + c_2x_2 + \ldots + c_nx_n$, where $c_1, c_2, \ldots, c_n$ are the weights assigned to each node in the graph, and $x_1, x_2, \ldots, x_n$ are binary decision variables representing whether a node is included in the solution or not.

Constraints: The constraints ensure that the solution represents a valid graph with no disconnected nodes and no cycles.

Each node can only be included once $x_1 + x_2 + \ldots + x_n = 1$ if there is an edge between two nodes, they must both be included $x_i + x_j \geq 1$ for all (i,j) pairs where $A_{ij} = 1$. There can be no cycle in the solution $x_i + x_j \leq 1$ for all (i,j) pairs where there is cycle in the solution.

The decision variables x_i are binary, taking the value of 1 if node i is included in the solution and 0 otherwise. The objective function Z is minimized by selecting the set of nodes that have the lowest weights while satisfying the constraints. The solution of this LPP will provide a valid graph with theminimum weight.

2.5Example on LPP using a linear programming solver to obtain upper domination number of graph:

Consider the following graph with vertex set $\{V_1, V_2, V_3, V_4, V_5\}$

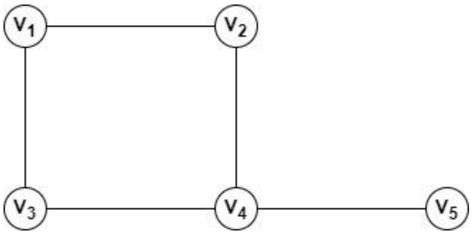


Figure 6

The vertex set$\{V_1, V_4\}$which forms dominating set of the graph with the minimum size of 2. Therefore, the dominating number of given graph $\gamma \quad (G) = 2$.

To find the upper dominating number of this graph$\Gamma(G)$using linear programming, we can follow the steps.

1. Let $x_1,\ x_2,\ x_3,\ x_4,\ and\ x_5$ be binary decision variables representing whether vertex is included or not included in the dominating set.

2. The objective function is to Minimize $z = x_1 + x_2 + x_3 + x_4 + x_5$, which represents the smallest size of the dominating set.

3. The conditions on constraints are

a) If vertex $x_i$where $i = 1,2,3,4,5$is not in the dominating set, then at least one of its neighbours must be in set:

$x_1 + x_3 \geq 1$ (This constraint ensures that either vertex V_1or vertex V_3 (or both) must be connected to some other vertex)

$x_2 + x_4 \geq 1$(This constraint ensures that either vertex V_2 or vertex V_4 (or both) must be connected to some other vertex)

$x_1 + x_2 + x_4 \geq 1$ (This constraint ensures that at least one of the verticesV_1,V_2, or V_4 must be connected to some other vertex)

$x_3 + x_4 \geq 1$ (This constraint ensures that either vertex V_3 or vertex V_4 (or both) must be connected to some other vertex)

$x_4 + x_5 \geq 1$ (This constraint ensures that either vertex V_4 or vertex V_5 (or both) must be connected to some other vertex)

b) If vertex x_i where $i = 1,2,3,4,5$ is in the dominating set, then it does not need to be adjacent to any other vertex in the set:

$x_1 + x_2 \leq 1$ (This constraint means that either vertex V_1 or vertex V_2(or neither) can have an outgoing edge, but not both. It ensures that there is at most one edge leaving vertex V_1 and vertex V_2 combined)

$x_3 + x_4 \leq 1$ (Similarly, this constraint means that either vertexV_3 or vertex V_4 (or neither) can have an outgoing edge, but not both. It ensures that there is at most one edge leaving vertexV_3 andvertex V_4combined)

and $x_5 \leq 1$(This constraint ensures that vertex V_5 can have at most one outgoing edge. Since it has no other adjacent vertex in the given graph, it can have either zero or one outgoing edge)

These constraints effectively enforce that each vertex has a maximum of one outgoing edge, ensuring acyclic behaviour in the graph. They prevent the formation of loops or cycles, making the graph undirected acyclic graph.

c) Each variable x_i must be binary means $x_i \in \{0,1\}$

4. Solve the resulting LPP using a graphical method or linear programming solver to obtain the minimum size of a dominating set.We get the solution like

$x_1 = 0,\ x_2 = 1, x_3 = 0, x_4 = 1, x_5 = 1$. Therefore $z = 3$

This solution corresponds to the set of vertices $\{V_2, V_4, V_5\}$, which form dominating set of the given graph with the minimum size with 3. Therefore, the upper domination number of graph is $\Gamma(G) = 3$.

III. Applicationsof domination and fractional domination in computational biological systems

1) One prominent example of computational biology in action is the field of bioinformatics, which focuses on analysing large-scale biological data, particularly genomic and proteomic data, using computational tools and techniques. The Human Genome Project, completed in 2003, marked a significant milestone in bioinformatics by sequencing the entire human genome, which consists of over three billion base pairs of DNA. Since then, computational methods have been essential in analysing and interpreting genomic data to uncover the genetic basis of diseases, identify potential drug targets, and understand evolutionary relationships between species.

2) Another example is the use of computational modelling and simulation to study biological systems at various levels of complexity. For instance, systems biology employs mathematical models to simulate the behaviour of biological networks, such as gene regulatory networks or metabolic pathways, and predict how perturbations to these networks may affect cellular behaviour. By integrating experimental data with computational models, researchers can gain insights into the underlying mechanisms of diseases like cancer and develop strategies for therapeutic intervention

3) In drug discovery and development, computational biology plays a critical role in rational drug design, virtual screening, and pharmacokinetic modelling. For example, molecular docking algorithms use computational models of protein-ligand interactions to predict the binding affinity of small molecules to target proteins, helping identify potential drug candidates with therapeutic efficacy. Similarly, pharmacokinetic modelling allows researchers to predict the absorption, distribution, metabolism, and excretion of drugs in the

body, aiding in the optimization of drug dosing regimens and minimizing the risk of adverse effects.

4) Furthermore, computational biology is increasingly being applied in personalized medicine to tailor medical treatments to individual patients based on their genetic makeup, lifestyle factors, and disease characteristics. For example, genomic sequencing and analysis can identify genetic variants associated with drug response or disease susceptibility, enabling clinicians to prescribe medications that are most likely to be effective for a particular patient or to recommend preventive measures for individuals at high risk of developing certain diseases.

Computational biology plays a vital role in advancing our understanding of complex biological systems, driving innovation in healthcare, and improving human health outcomes. By leveraging computational techniques and interdisciplinary collaboration, researchers can address fundamental questions in biology and develop novel solutions to pressing medical challenges.

Fractional domination is a concept derived from graph theory, where it is used in computational biology, networks often represent various biological systems such as gene regulatory networks, protein-protein interaction networks, metabolic networks, and ecological networks. Applying fractional domination in computational biology can provide insights into the robustness, stability, and controllability of these biological systems.

Once dominating sets are identified, interpret the biological significance of these sets. For instance, in a gene regulatory network, a dominating set might represent a group of genes that collectively control the expression of other genes in the network. Understanding the roles and interactions of these genes can provide insights into regulatory mechanisms.

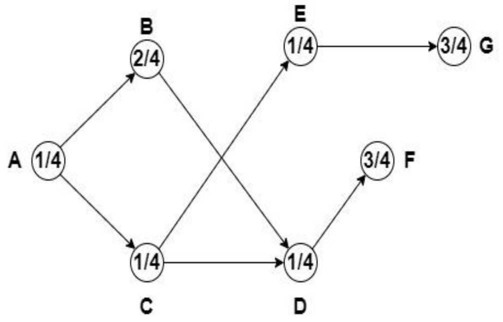
3.1Computational biological system such as gene regulatory networks:

let's consider a simplified example of a gene regulatory network (GRN) represented as a graph. In this network, nodes represent genes, and edges represent regulatory interactions where one gene regulates the expression of another gene. We will apply fractional domination to identify key genes that collectively control a significant portion of the network.

Consider the following gene regulatory network as a graph G:

Genes: A, B, C, D, E, F, G.

Regulatory Interactions:A → B, A → C, B → D, C → D, C → E, D → F, E → G.



Directed graph$(G)$of regulatory interactions

$$\begin{array}{c} \begin{matrix} A & B & C & D & E & F & G \end{matrix} \\ \begin{bmatrix} 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \end{array}$$

Adjacency matrix$A(G)$

Figure 7

Use algorithm to find dominating set within the network. In fractional domination we take a dominating function $f$to be any$f: V(G) \rightarrow [0,1]$ is function of given graph which allocates the values for each vertex $v \in V(G)$ in the unit interval [0,1]. The given function f is called a fractional dominating function if for every vertex $v \in V(G), f(N[v])$ = $\sum_{v \in N[v]} f(v) \geq 1$. It denotes the total value of the vertices in the closed neighbourhood of $v \in V(G)$such that$N[v]$ is at least one it means$(\sum N[v]) \geq 1$. Here in this graph $\gamma_f\ (G) = \frac{12}{4} = 3$.

We can frame it as an optimization problem aimed at minimizing the expression level of a certain gene while satisfying the regulatory constraints. Let's choose gene G for minimization.

Objective function: Minimize xG

Constraints:

1. Gene expression levels are non-negative: $xA, xB, xC, xD, xE, xF, xG\ \geq\ 0$
2. Regulatory interactions: $xB \geq xA, xC \geq xA, xB \geq xD,\ xD \geq xC, xE \geq xC$
   $xF \geq xD,\ xG \geq xE.$

These constraints ensure that the expression levels of genes are regulated according to the specified interactions.With this formulation, the objective is to minimize the expression level of gene G while satisfying the regulatory interactions among the genes.

Illustration: If we take $(A(G)+I)x \geq 1$,will convert it into system of equations like $x_1 + x_2 + x_3 = 1,\ x_2 + x_4 = 1\ , x_3 + x_4 + x_5 = 1, x_4 + x_6 = 1, x_5 + x_7 = 1,\ x_6 = 1$ and $x_7 = 1$. By solving this system of equations will get $x_1 = -1, x_2 = 1, x_3 = 1, x_4 = 0$, $x_5 = 0, x_6 = 1, x_7 = 1$. Hence fractional domination number is objective function minimizes $1^T x = x_1 + x_2 + x_3 + x_4 + x_5 + x_6 + x_7 = 3$.

The goal is to find a set of nodes that collectively cover a certain fraction of the entire network. Let's say we aim to cover 100% of the network. The fractional domination number provides a quantitative measure of the burden of a network, and can be used to design more robust and efficient networks.

After applying fractional domination concept, we find the dominating sets. Let's assume we find the dominating set {A, D, E}.The dominating set represents a group of genes that collectively regulate a significant all portion of the network. In this example, dominating set includes genes A, D and E, which together regulate genes B, C,F and G.

We can assess the robustness of the network by simulating perturbations to the dominating sets. For example, we can remove genes A, D and E from dominating set and observe how it affects the expression of genes B, C,F and G.

Based on the dominating sets, we can devise control strategies to manipulate the network. For instance, by targeting genes A, D and E which are part of dominating set, we may be able to modulate the expression of downstream genes B, C,F and G.

Combine fractional domination analysis with gene expression data or other omics data to enhance the biological relevance and accuracy of the findings. This could involve validating the regulatory interactions predicted by the dominating sets through experimental assays.

This simplified example demonstrates how fractional domination can be applied to analyse gene regulatory networks in computational biology, providing insights into the control and regulation of gene expression within complex biological systems.

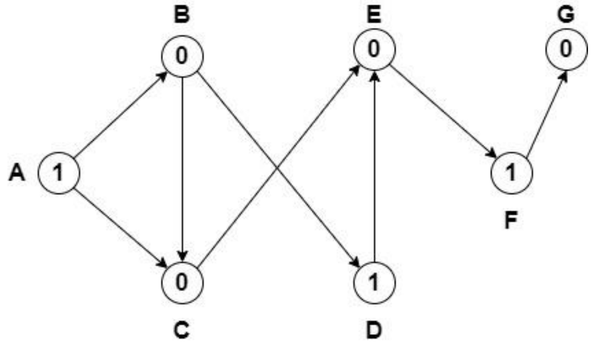
3.2 Computational biological system such as protein-protein interaction network:

We can take an example of a protein-protein interaction network (PPIN) represented as a graph. In this network, nodes represent proteins, and edges represent physical interactions between proteins. We can apply fractional domination to identify key proteins that collectively influence a significant portion of the network.

Consider the following protein-protein interaction network

Proteins: A, B, C, D, E, F, G

Physical Interactions:A → B, A → C, B →C, B→ D, C → E, D →E, E →F,F→G.



$$\begin{array}{c}
\begin{array}{ccccccc} A & B & C & D & E & F & G \end{array} \\
\begin{bmatrix}
0 & 1 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0
\end{bmatrix}
\end{array}$$

Directed graph (G) of protein-protein interaction network

Adjacency matrix $A(G)$

Figure 8

To find dominating set within the network we used fractional domination concept as dominating function $f$to be any$f: V(G) \to [0,1]$ is function of given graph which allocates the values for each vertex $v \in V(G)$ in the unit interval [0,1]. The given function f is called a fractional dominating function if for every vertex $v \in V(G), f(N[v])$ = $\sum_{v \in N[v]} f(v) \geq 1$. It denotes the total value of the vertices in the closed neighbourhood of $v \in V(G)$such that$N[v]$ is at least one it means$(\sum N[v]) \geq 1$. Here in this graph $\gamma_f\ (G) = 3$.If the goal is to find a set of nodes that collectively cover a certain fraction of the entire network. The fractional domination number provides a quantitative measure of the burden of a network, and can be used to design more robust and efficient networks.

Let's define a binary decision variable $x_{ij}$for each pair of proteins i and j where:

1) $x_{ij} = 1$ if there exists a physical interaction from protein i to protein j
2) $x_{ij} = 0$ otherwise.

Objective Function: We aim to minimize the total number of physical interactions, which can be represented as the sum of all decision variables:

Minimize:$\sum_{i,j} x_{ij}$

Constraints: Based on the given interactions, we need to ensure that if a protein interacts with another, it's included in the count:

1. For each pair of interacting proteins, the decision variable must be set to 1
$$xAB + xAC + xBC + xBD + xCE + xDE + xEF + xFG \geq 1$$
2. Each decision variable must be binary: $x_{ij} \in [0,1]$for all *i,j*.

This formulation ensures that we minimize the number of physical interactions while ensuring that if there's an interaction between two proteins, it's counted. Linear programming techniques can be used to solve this optimization problem.

Illustration: If we take $(A(G) + I)x \geq 1$,will convert it into system of equations like $x_1 + x_2 + x_3 = 1,\ x_2 + x_3 + x_4 = 1$, $x_3 + x_5 = 1, x_4 + x_5 = 1, x_5 + x_6 = 1,$ $x_6 + x_7 = 1$and $x_7 = 1$. By solving this system of equations will get $x_1 = 0, x_2 = 1,$ $x_3 = 0, x_4 = 0, x_5 = 1, x_6 = 0, x_7 = 1.$ Hence fractional domination number is objective function minimizes $1^T x = x_1 + x_2 + x_3 + x_4 + x_5 + x_6 + x_7 = 3$.

After applying fractional domination, we find the dominating sets. Let's assume we find the dominating set {A, D, F}. Dominating set represents a group of proteins that collectively interact with a significant portion of the network. In this example, dominating set includes proteins A, Dand F, which together interact with proteins B, C, E and G.

Assess the robustness of the network by simulating perturbations to the dominating sets. For example, we can remove proteins A, D and F from dominating setand observe how it affects the interactions of proteins B, C, E and G.

Devise control strategies based on the dominating sets to manipulate the network. For instance, by targeting proteins A, Dand F, which are part of dominating set, we may be able to influence the interactions of proteins B, C, E and G.

Combine fractional domination analysis with gene expression data, functional annotations, or other omics data to enhance the biological relevance and accuracy of the findings. This could involve validating the protein interactions predicted by the dominating sets through experimental assays.

Given example illustrates how fractional domination can be applied to analyse protein-protein interaction networks in computational biology, providing insights into the key proteins that regulate complex biological processes.

3.3 Healthcare Network Optimization

An example of a health-related application that utilizes the concept of fractional domination number in graph theory.In healthcare systems, it's crucial to optimize the allocation of resources such as medical facilities and personnel to ensure efficient and effective delivery of care to patients. Graph theory can be employed to model the network of healthcare facilities, with vertices representing hospitals, clinics, or other medical centres, and edges representing the connections or accessibility between them.Now, let's define the concept of fractional domination number in this context.

Consider a scenario where a healthcare organization wants to optimize the placement of mobile medical units (MMUs) to provide healthcare access to remote or underserved areas. The organization can model the geographic region as a graph, with vertices representing potential locations for MMUs and edges representing the accessibility between these locations based on factors such as road networks and distance.

Let's say we have a graph representing a region with several villages (vertices) and roads connecting them (edges). Each village requires healthcare access, and the organization aims to deploy MMUs strategically to cover as many villages as possible.

1) Assign weights to vertices based on factors such as population size, healthcare needs, and geographical location. For example, larger villages or those with higher healthcare needs may have higher weights.

2) Deploy MMUs to selected vertices in such a way that the combined coverage provided by these MMUs forms a dominating set with a fractional domination number as close to 1 as possible. This means that the deployed MMUs cover as much of the population as possible while minimizing redundancy and overlap.

3) Use LPP algorithm and LPP optimization techniques from graph theory to determine the optimal placement of MMUs to achieve maximum coverage with minimum resources. This may involve solving fractional domination problems or related optimization problems.

4) By utilizing the concept of fractional domination number in graph theory, healthcare organizations can optimize the deployment of resources such as mobile medical units to provide efficient and equitable healthcare access to populations in need, especially in remote or underserved areas.

Illustration: Let's create a simple graph representing a healthcare network for a region with several villages and roads connecting them. In this example, we will consider four villages (V_1, V_2, V_3, V_4) and the roads (edges) connecting them. Each village represents a potential location for a mobile medical unit (MMU).Here is the graph:

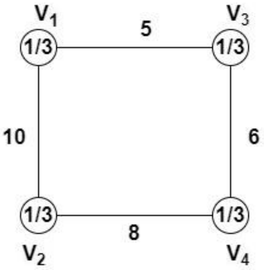


Figure 9

- The vertices (V_1, V_2, V_3, V_4) represent the four villages.
- The edges represent roads connecting the villages.
- The numbers on the edges represent distances or travel times between villages. For example, the edge between $V_1$and V_2 has a weight of 10, indicating that the distance between $V_1$and V_2 is 10 units.

This graph can be used to model the connectivity and distances between villages in a region, allowing us to apply healthcare network optimization techniques such as fractional domination to determine the optimal placement of mobile medical units for maximum coverage and efficiency. Even though vertices $V_1$and V_4 are not connected but with the help of equal weightage we can optimize the allocation of resources such as medical facilities and personnel to ensure efficient and effective delivery of care to patients.Given example illustrates how fractional domination can be applied to analyse healthcare network optimization in computational biology.

IV. Conclusion

Computational biology plays a vital role in advancing our understanding of complex biological systems, driving innovation in healthcare, and improving human health outcomes. By leveraging computational techniques and interdisciplinary collaboration, researchers can address fundamental questions in biology and develop novel solutions to pressing medical challenges.The use of domination and fractional domination is important applications in these fields. The fractional domination number provides a quantitative measure of the burden of a network, and can be used to design more robust and efficient networks.Here we have used these conceptsin computational biological systems such as gene regulatory networks, protein-protein interaction networks and healthcare network optimization.

References

[1] A. V. Aho, J.E. Hopcroft, and J. D. Ullman, “The design and analysis of computer algorithms”, *Addison-Wesley Publishing Company*, (1974).

[2] M. Chellali, O. Favaron, A. Hansberg, and L. Volkmann, “k-Domination and k-Independence in graphs: A Survey”, *Graphs and Combinatorics* 28 (2012), 1–55.

[3] S.Y.Chao, “Graph theory and analysis of biological data in computational biology”, *Advanced Technologies*, 2009, DOI: 10.5772/8205.

[4] G. S. Domke, S.T. Hedetniemi, and R. C. Laskar, “Fractional packings, coverings and irredundance in graphs”, *CongressusNumerantium*. Published (1988), 66, 227-238.

[5] G. H. Fricke, M. A. Henning, O. R. Oellermann, and H. C. Swart, “An algorithm to find two distance domination parameters in a graph”, *Discrete Applied Mathematics* 68 (1996) 85 91.

[6] D. C. Fisher, “Fractional domination and fractional total domination of graph complements”, *Discrete Applied Mathematics,* 122 (2002), 283-291, DOI: /10.1016/S0166-218X (01)00305-5.

[7] T.W. Haynes, S.T. Hedetniemi and P. J. Slater, “Fundamentals of Domination in Graphs*”*, Marcel Dekker., Inc., New York.1998.

[8] Frank Harary, *Graph theory*, by Addison-Wesley publishing company Inc, 1969.

[9] T. W. Haynes, S. T. Hedetniemi, M. A. Henning, “Topics in Domination in Graphs”, *Springer Science and Business Media* LLC, 2020.

[10] J. K. Lana and G. J. Chang, “Algorithmic aspects of the k-domination problem in graphs”, *Discrete Applied Mathematics* 161 (2013) 1513–1520.

[11] H. B. Merouane and M. Chellali, “An algorithm for the dominator chromatic number of a tree”, *Journal of Combinatorial Optimization*, Springer Science Business Media New York 2013.

[12] C. Pang, R. Zhang, Q. Zhang, and J. Wang, “Dominating sets in directed graphs”, *Information Sciences*, 180 (2010) 3647–3652, DOI:10.1016/j.ins.2010.06.009.

[13] M. Sarada, R. Jain, and G. Mundhe, “Bounds for fractional domination number of some graphs and their dual graphs”, *Indian Journal of Natural Sciences*, 14(80), (2023), 61661-61670. ISSN: 0976 – 0997.

[14] M. Sarada and R. Jain, “Relation between the fractional domination number ofgraphs andtheirlinegraph”, communicated in *Jñānābha*, 2024.