Project Report

On

OPTIMIZATION OF FUNCTIONS OF THE LAPLACIAN EIGENVALUES

Submitted by-

P Sharath Chandra (170001033)

3rd Year

Under the Guidance of

Dr. Kapil Ahuja

Department of Computer Science and Engineering

Indian Institute of Technology, Indore

Autumn 2019

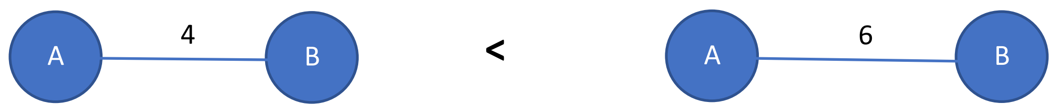
Contents:

PROBLEM STATEMENT:

HOW STRONGLY IS A GRAPH CONNECTED?

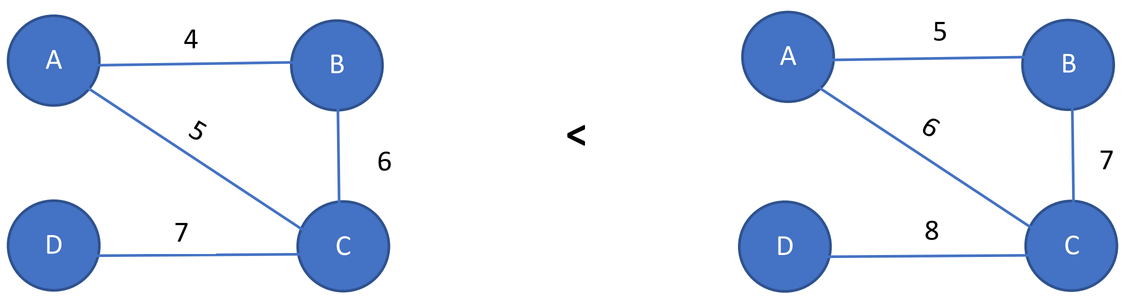
DEFINITION: What does strongly connected mean?

If an edge exists between two nodes of a graph, then the nodes are said to be strongly connected if the weight of the edge is high.

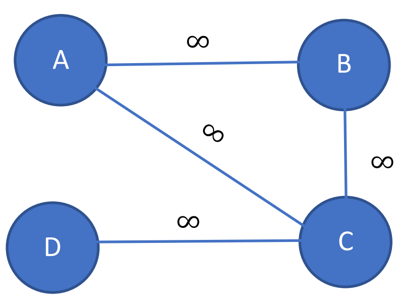


DEFINITION: When is a graph strongly connected?

A graph is said to be strongly connected if all edges are strongly connected.



Weaker Stronger



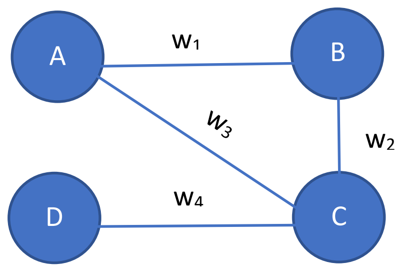
This graph is the most strongly connected graph of the above variant.

So, there is no point in finding the most strongly connected graph, as it is trivial that edges with infinite weight would be the most strongly connected graph.

CONSTRAINTS:

So, we constrain the weights of the graph, in the following way:

Summation of the weights is kept constant.



If the weight of an edge $\boldsymbol{e_i}$ is $\boldsymbol{w_i}$, then:

$$\Sigma\, w_i = W, \text{ for some constant } W.$$

Without loss of generality, $\boldsymbol{W}$ can be assumed to be 1. Then, for any value of $\boldsymbol{W}$, all $\boldsymbol{w_i}$ can be normalized according.

GOAL:

Given a graph *G* and a constant *W*, the goal is to assign optimal weights to the edges, such that the sum of the weights is equal to *W*, and *G* is the most strongly connected.

FORMULATION OF OPTIMIZATION PROBLEM:

FEW DEFINITIONS:

Laplacian Matrix: It is a matrix representation of a graph.

For an undirected and unweighted graph, the Laplacian matrix L is defined as-

$$L = D - A,$$

where D is the degree matrix, which is a diagonal matrix whose diagonal elements corresponds to the degree of the respective node; and A is the adjacency matrix of the graph.

For an undirected and weighted graph, the Laplacian matrix L is defined as-

$$L = I * w * I^T,$$

where I is the incidence matrix of the graph, and w is a diagonal matrix whose diagonal entries correspond to the weight of an edge.

Properties of Laplacian Matrix:

a. L is symmetric and positive-semidefinite.
b. Every row sum of L is equal to zero.
   So, $e = [1, 1, 1, \ldots, 1]^T$ is an eigenvector of L with eigenvalue zero, satisfying-
   $$L * e = 0 = 0 * e$$
c. The algebraic multiplicity of 0 eigenvalue is a measure of number of connected components in the graph.

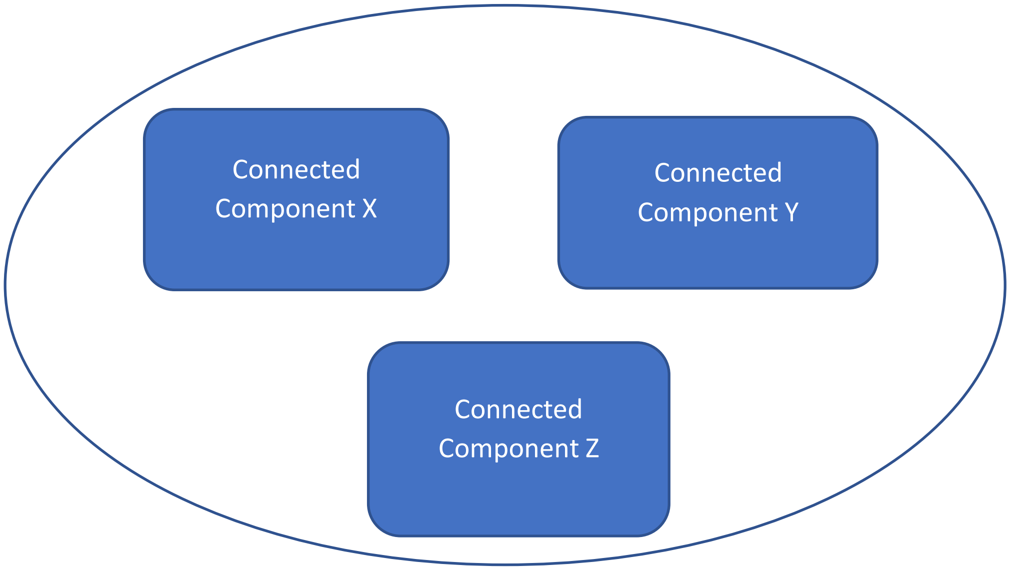
INSIGHTS FROM THE PROPERTIES OF THE LAPLACIAN MATRIX:

From (a), all the eigenvalues of L are non-negative.

From (b), the smallest eigenvalue of L is always zero, $\lambda_1 = 0$.

From (c), the number of connected components in the graph is as the multiplicity of 0 eigenvalue.

The Figure below explains the meaning of (c). Consider a graph G, there are three connected components of G (X, Y, Z), such that though X, Y, Z are connected internally, they are not connected with each other.



Entire Graph ***G***

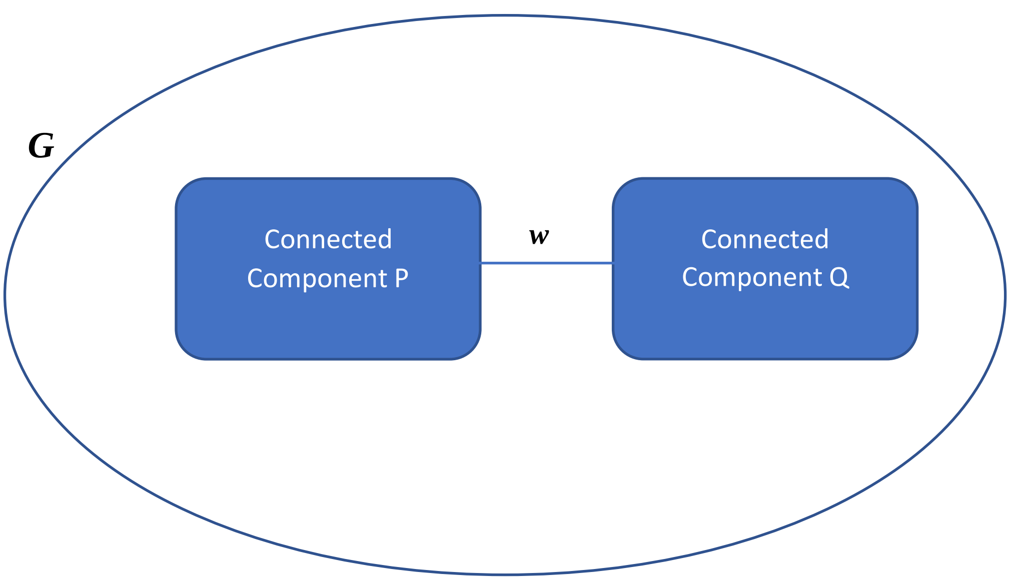
Since, there are three connected components of ***G***, which are not connected with each other; (c) says that there are three eigenvalues of Laplacian matrix of ***G***, that are equal to zero.

Suppose, X, Y and Z are all connected, then there would be only one connected component of ***G***, which implies that there is only one eigenvalue that is equal to zero.

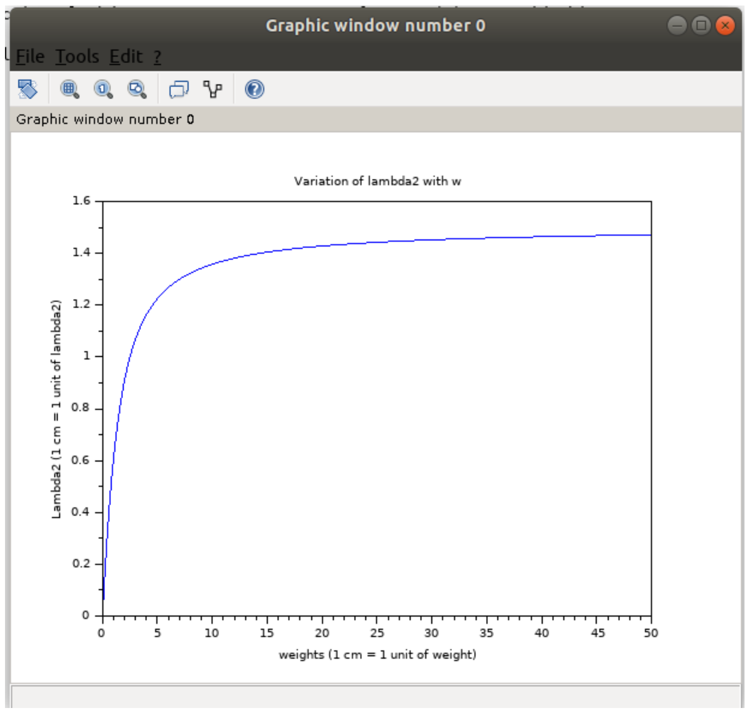
SECOND SMALLEST EIGENVALUE OF THE LAPLACIAN MATRIX:

The second smallest eigenvalue λ_2 is non-zero if the graph is connected.

Consider the given graph ***G***,



there are two connected components P and Q; they are joined by an edge of weight w; if $w = 0$, then P and Q are not connected; else they are connected. Here is a graph which shows the relationship between λ_2 and w:



As it can be seen, λ_2 *increases* with w; this is the key observation.

Now, keeping summation of weights constant, we can vary the weights of the edges of the graph to vary the value of λ_2. The value of w corresponding to maximum value of λ_2 are the weights of the edges of the most strongly connected graph.

OPTIMIZATION PROBLEM:

$$\text{Maximize } \lambda_2$$

$$\text{Subject to: } \boldsymbol{w} \geq 0$$

$$\boldsymbol{e}^T * \boldsymbol{w} = W$$

where $\boldsymbol{e} = [1, 1, \ldots, 1]^T$, $\boldsymbol{w} = [w1, w2, \ldots, wn]^T$, W is a positive constant.

TO SHOW THAT THE OPTIMIZATION PROBLEM IS A CONVEX PROGRAMMING PROBLEM:

DEFINITION: Convex Optimization Problem

The minimization problem whose objective function and equality constraints are convex functions, and the inequality constraints are concave functions.

The obtained optimization problem can be written as-

$$\text{Minimize } -\lambda_2$$
$$\text{Subject to: } \boldsymbol{w} >= 0$$
$$\boldsymbol{e^T * w = W}$$

Here, the objective function, the equality constraint, and the inequality constraint would be the following respectively:

$$f(w) = -\lambda_2$$
$$a(w) = \sum w - W = 0$$
$$c(w) = w >= 0.$$

It was shown that: λ_2 increases with some edge weight **w**, this would imply $\frac{d(\lambda 2)}{dw} > 0$. Same is true with other eigenvalues as well [Ref. (i)]. So, all eigenvalues are concave functions of ***w***.

λ_2 **=** minimum $\{\lambda_2, \ldots., \lambda_n\}$. Since the minimum function over concave functions is also concave, ***-f(w)*** is a concave function; this implies ***f(w)*** is a convex function.

Clearly, ***a(w)*** and ***c(w)*** are linear functions, they are both concave and convex functions.

Since all the above-mentioned conditions for a convex programming problem are satisfied, therefore the obtained optimization problem is a convex programming problem.

INSIGHTS ON SOLVING THE OPTIMIZATION PROBLEM:

[Reference (i)]:

The obtained convex optimization problem can be formulated as a Semi-Definite Programming (SDP) problem:

$$\text{Maximize } \mu$$

$$\text{Subject to: } \mu I <= L + \beta 1 1^T$$

$$w >= 0,\ e^T * w = W$$

where μ, $\beta \in R$, and $w \in R^n$.

[Reference (ii)]:

This quantity $\boldsymbol{\lambda_2}$ is called as the algebraic connectivity of a graph. The following bounds can be shown:

$$\boldsymbol{\lambda_2 <=} (n * d_{min}) /(n-1) <= 2m/(n-1)$$

where n is the number of vertices, m is the number of edges, and d_{min} is the minimum degree of the nodes in a graph.

FUTURE WORK:

The following methods were tried to solve the obtained optimization problem:

a. Contours for the λ_2 as a function 2 and 3 variables were plotted.
b. CVX: Matlab Software for Convex Programming was tried.

Many resources stated that the obtained optimization problem needs to be converted to a semi-definite programming (SDP) problem, whose solution can be found in polynomial time.

So, future work would include to learn about SDP and apply the principles here.

REFERENCES:

1. Convex optimization of Graph Laplacian eigenvalues- Stephen Boyd
2. Algebraic connectivity of graphs- Miroslav Fiedler
3. Weighted Algebraic Connectivity Maximization for Optical Satellite Networks