

# ΙΟΝΙΟ ΠΑΝΕΠΙΣΤΗΜΙΟ

ΣΧΟΛΗ ΕΠΙΣΤΗΜΩΝ ΤΗΣ ΠΛΗΡΟΦΟΡΙΑΣ  
ΚΑΙ ΠΛΗΡΟΦΟΡΙΚΗΣ

Τμήμα Πληροφορικής



Πτυχιακή Εργασία

Μελέτη Τοποθέτησης Υπηρεσιών σε Δικτυοκεντρικά  
Περιβάλλοντα με Αλγεβρική Θεωρία Γράφων

Ιωάννης Αγγέλης

Επιβλέπων: Αν. Καθηγητής Κωνσταντίνος Οικονόμου

21 Ιουνίου 2019



## Επιβλέπων Καθηγητής

**Κωνσταντίνος Οικονόμου**, Αναπληρωτής Καθηγητής,  
Ιόνιο Πανεπιστήμιο

## Τριμελής Επιτροπή

**Κωνσταντίνος Οικονόμου**, Αναπληρωτής Καθηγητής,  
Ιόνιο Πανεπιστήμιο

**Δημήτριος Τσουμάκος**, Αναπληρωτής Καθηγητής,  
Ιόνιο Πανεπιστήμιο

**Θεόδωρος Ανδρόνικος**, Αναπληρωτής Καθηγητής,  
Ιόνιο Πανεπιστήμιο



Στην οικογένειά μου, για όλη την υποστήριξη που μου έχουν προσφέρει  
και σε όσους ήταν κοντά μου, όποτε τους χρειάστηκα



## Περίληψη

Στα σημερινά δίκτυα ένα σημαντικό πρόβλημα που συναντάμε είναι η τοποθέτηση υπηρεσιών λόγω του αυξημένου αριθμού των κόμβων που υπάρχουν στα δίκτυα. Στόχος είναι να ελαχιστοποιηθεί το κόστος μέσα στο δίκτυο δηλαδή που θα τοποθετηθεί η υπηρεσία ώστε όλοι οι κόμβοι να έχουν το ελάχιστο κόστος από αυτόν. Το πρόβλημα αυτού του τύπου είναι αντικείμενο μελέτης του Facility Location και πιο συγκεκριμένα το πρόβλημα αυτό ονομάζεται 1–median. Υπάρχουν αρκετές λύσεις και καλές προσεγγίσεις για την τοποθέτηση της υπηρεσίας στο δίκτυο ειδικότερα θα χρησιμοποιήσουμε την αλγεβρική θεωρία γράφων για την τοποθέτηση της υπηρεσίας στο δίκτυο. Η βέλτιστη λύση δίνεται από το Closeness Centrality, το Degree και Eigenvector Centrality δεν δίνουν πάντα την βέλτιστη λύση άλλα μια προσέγγιση για το που θα τοποθετηθεί η υπηρεσία. Το Closeness Centrality για μεγάλα δίκτυα έχει τεράστιο κόστος υπολογισμού, στόχος της πτυχιακής είναι να εξετάσουμε πόσο καλή είναι οι προσεγγίσεις των μέτρων σε σχέση με το Closeness Centrality.

# Abstract

In the 21st century we are in the Information Age. An important role in shaping it has been computer networks. Changing the way we share information and services. In our everyday life we enjoy and use several services. An important problem that we encounter is the Facility Location and more specifically the case of 1-median. The 1-median problem is a special category of the  $p$ -median problem. The problem is NP-hard on general graphs and networks for an arbitrary  $p$  (where  $p$  is a variable). Polynomial time algorithms exist for arbitrary  $p$  when the network is a tree. The aim of the thesis is to study where to place the service on the network and how close we are to the optimal solution using Algebraic Graph Theory. The optimal solution for the 1-median is given by Closeness centrality however there are other centrality measures such as Degree and Eigenvector centrality which provide approaches for optimal solution. For the comparison of the measures, we constructed networks consisting of geometric random with 1000 nodes. For the evaluation of the measures we have defined the concept of total cost for the network and compared the computation time of each measure. The results of Eigenvector centrality with use of Power Method is interesting because it has a very good approach to the placement of the service with a small computation time.

# Ευχαριστίες

Στο σημείο αυτό αισθάνομαι την ανάγκη να εκφράσω τις ειλικρινείς και θερμές ευχαριστίες μου σε όσους συνέβαλλαν στην ολοκλήρωση αυτής της προσπάθειας.

Πρώτα από όλα, στον επιβλέποντα καθηγητή της πτυχιακής μου εργασίας, κύριο Κωνσταντίνος Οικονόμου για τη συνεχή καθοδήγηση, την αμέριστη υποστήριξη, καθώς επίσης και την αδιάκοπη συμπαράσταση και ενθάρρυνση που μου παρείχε σε όλο αυτό το χρονικό διάστημα.

Ακόμη θα επιθυμούσα να ευχαριστήσω και τον κύριο Γιώργο Κουφουδάκη γιατί από την πρώτη στιγμή με βοήθησε και με καθοδήγησε σε αυτήν την προσπάθειά μου.

Θα ήθελα να ευχαριστήσω τους φίλους μου που μου σταθήκανε, με στηρίζανε και με βοηθήσανε ο καθένας με τον δικό του τρόπο .

Τέλος θα ήθελα να ευχαριστήσω την οικογένεια μου που με στήριξε και πίστεψε σε έμενα ειδικότερα σε αυτά τα τέσσερα χρονιά. Τους ευχάριστο για όλα όσα μου έχουνε προσφέρει και είμαι ευγνώμων που με στηρίζουν στην πορεία που έχω ακολουθήσει.

# Περιεχόμενα

|      |                                                 |    |
|------|-------------------------------------------------|----|
| A'   | Εισαγωγή                                        | 1  |
| A'.1 | Βιβλιογραφική Έρευνα . . . . .                  | 2  |
| A'.2 | Αποτελέσματα και Συνεισφορά . . . . .           | 3  |
| A'.3 | Αντιστοιχούμενα Κεφάλαια . . . . .              | 3  |
| B'   | Γραμμική Άλγεβρα                                | 4  |
| B'.1 | Χρήσιμα Στοιχεία Γραμμικής Άλγεβρας . . . . .   | 4  |
| B'.2 | Άθροισμα και Βαθμωτός Πολλαπλασιασμός . . . . . | 6  |
| B'.3 | Πολλαπλασιασμός πινάκων . . . . .               | 9  |
| B'.4 | Ορίζουσες και Έχνος . . . . .                   | 11 |
| B'.5 | Ιδιοτιμές και Ιδιοδιανύσματα . . . . .          | 12 |
| Γ'   | Θεωρία Γραφημάτων                               | 15 |
| Γ'.1 | Βασικές έννοιες της Θεωρία Γραφημάτων . . . . . | 15 |
| Γ'.2 | Ιστορική ανάδρομη . . . . .                     | 16 |
| Γ'.3 | Τύποι Γραφημάτων . . . . .                      | 17 |
| Γ'.4 | Πίνακας Γειτνίασης . . . . .                    | 20 |
| Δ'   | Μοντελοποίηση του 1-median problem              | 25 |
| Δ'.1 | Πως βρίσκουμε 1-median problem . . . . .        | 25 |
| Ε'   | Πειράματα και αποτελέσματα                      | 30 |
| Ε'.1 | Κατασκευή δικτύων . . . . .                     | 30 |
| Ε'.2 | Ορισμός κόστους . . . . .                       | 32 |
| Ε'.3 | Δημιουργία του shortest path tree . . . . .     | 32 |

|                                                          |    |
|----------------------------------------------------------|----|
| Περιεχόμενα                                              | v  |
| Ε'.4 Υπολογισμός των μέτρων . . . . .                    | 33 |
| Ε'.5 Πειράματα . . . . .                                 | 34 |
| Γ' Συμπεράσματα και Μελλοντική Εργασία                   | 47 |
| Γ'.1 Σύνοψη και Συμπεράσματα . . . . .                   | 47 |
| Γ'.2 Μελλοντική Εργασία . . . . .                        | 48 |
| Βιβλιογραφία                                             | 48 |
| Παραρτήματα                                              | 54 |
| I Δημιουργία δικτύων κώδικας                             | 55 |
| II Υπολογισμός συνολικού κόστους                         | 57 |
| III Shortest path tree                                   | 58 |
| IV Degree Centrality                                     | 60 |
| V Closeness Centrality                                   | 61 |
| VI Eigenvector Centrality                                | 62 |
| VII Eigenvector Centrality με την χρήση της Power Method | 63 |
| Γλωσσάρι Ξενικών Όρων                                    | 65 |
| Συμβολισμοί Μεγεθών                                      | 67 |

# Κατάλογος Σχημάτων

|      |                                                                                                                          |    |
|------|--------------------------------------------------------------------------------------------------------------------------|----|
| Γ'.1 | Ο χάρτης των 7 γεφυρών του Königsberg τον 17 <sup>ο</sup> αιώνα. . . . .                                                 | 16 |
| Γ'.2 | Μοντελοποίηση του προβλήματος των 7 γεφυρών του Königsberg με την μορφή γραφήματος. . . . .                              | 17 |
| Γ'.3 | Το Γ'.3α' ένα μη-κατευθυνόμενα γράφημα και το Γ'.3β' ένα κατευθυνόμενα γράφημα. . . . .                                  | 18 |
| Γ'.4 | Το Γ'.4α' είναι ένα συνεκτικό γράφημα και το Γ'.4β' είναι ένα πλήρες γράφημα $K_6$ . . . . .                             | 18 |
| Γ'.5 | Το Γ'.5α' είναι ένα πολυγράφημα και το Γ'.5β' είναι ένα ψευδογράφημα. . . . .                                            | 19 |
| Γ'.6 | Παράδειγμα ενός έμβарου μη-κατευθυνόμενου γραφήματος. . . . .                                                            | 19 |
| Γ'.7 | Δυο διαφορετικά είδη γραφημάτων. . . . .                                                                                 | 22 |
| Δ'.1 | Ένα μη-κατευθυνόμενο γράφημα. . . . .                                                                                    | 26 |
| Ε'.1 | Το Ε'.1α' ένα μη-κατευθυνόμενα γράφημα και το Ε'.1β' το shortest path tree με ρίζα του δέντρου τον κόμβο $u$ . . . . .   | 33 |
| Ε'.2 | Συνολικό κόστος του Closeness, του Degree και του Eigenvector Centrality . . . . .                                       | 35 |
| Ε'.3 | Συνολικό κόστος του Closeness - Eigenvector Centrality με την χρήση της Power Method με 1, 5 και 10 Loop . . . . .       | 36 |
| Ε'.4 | Συνολικό κόστος του Closeness και του Eigenvector Centrality με την χρήση της Power Method με 100 και 200 Loop . . . . . | 36 |
| Ε'.5 | Ταύτιση των αποτελεσμάτων της Power Method 1 Loop με τα αποτελέσματα του Degree Centrality . . . . .                     | 37 |
| Ε'.6 | Συνολικό διάγραμμα κόστους με όλα τα μέτρα . . . . .                                                                     | 38 |
| Ε'.7 | Οι χρόνοι υπολογισμού του Closeness Centrality και του Degree Centrality . . . . .                                       | 39 |

|                                                                                                                                                                                                 |    |
|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----|
| E'.8 Οι χρόνοι υπολογισμού του Closeness Centrality και του Eigenvector Centrality . . . . .                                                                                                    | 40 |
| E'.9 Οι χρόνοι υπολογισμού του Closeness Centrality και του Eigenvector Centrality με την χρήση της Power Method με 1, 5 και 10 Loop . . . .                                                    | 40 |
| E'.10 Οι χρόνοι υπολογισμού του Closeness Centrality και του Eigenvector Centrality με την χρήση της Power Method με 100 και 200 Loop . . .                                                     | 41 |
| E'.11 Αποτελέσματα πολλαπλασιασμού του χρόνου υπολογισμού με το συνολικό κόστος για το Closeness, Degree και το Eigenvector Centrality . . . . .                                                | 42 |
| E'.12 Αποτελέσματα πολλαπλασιασμού του χρόνου υπολογισμού με το συνολικό κόστος για το Closeness και για το Eigenvector Centrality με την χρήση της Power Method για 1, 5 και 10 Loop . . . . . | 43 |
| E'.13 Αποτελέσματα πολλαπλασιασμού του χρόνου υπολογισμού με το συνολικό κόστος για το Closeness και για το Eigenvector Centrality με την χρήση της Power Method για 100 και 200 Loop . . . . . | 43 |
| E'.14 Αποτελέσματα πολλαπλασιασμού του χρόνου υπολογισμού με το συνολικό κόστος για όλα τα μέτρα . . . . .                                                                                      | 44 |

## Κατάλογος Πινάκων

|                                                                                                                                                                    |           |
|--------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------|
| <i>Δ'.1 Τα αποτελέσματα του Closeness centrality για το σχήμα Δ'.1 . . . . .</i>                                                                                   | <i>26</i> |
| <i>Δ'.2 Τα αποτελέσματα του Degree Centrality για το σχήμα Δ'.1. . . . .</i>                                                                                       | <i>27</i> |
| <i>Δ'.3 Τα αποτελέσματα του Eigenvector centrality για το σχήμα Δ'.1 που<br/>αντιστοιχίσουν στην μεγαλύτερη ιδιοτιμή <math>\lambda_1 = 4.0861</math> . . . . .</i> | <i>28</i> |
| <i>Ε'.1 Ποσοστιαία αύξηση κόστους σε σχέση με το Closeness Centrality . . .</i>                                                                                    | <i>38</i> |
| <i>Ε'.2 Αποτελέσματα μετρήσεων του χρόνου των μέτρων σε σχέση με το<br/>Closeness Centrality. . . . .</i>                                                          | <i>41</i> |

# Κεφάλαιο Α΄

## Εισαγωγή

**Σ**τον 21 αιώνα διανύουμε την εποχή της πληροφορίας (The Information Age). Σημαντικό ρόλο στην διαμόρφωση του είχαν και έχουν τα δίκτυα υπολογιστών. Αλλάζοντας έτσι τον τρόπο που διαμοιραζόμαστε τις πληροφορίες και τις υπηρεσίες. Στην καθημερινότητα μας απολαμβάνουμε και χρησιμοποιούμε αρκετές υπηρεσίες. Ένα σημαντικό πρόβλημα που συναντάμε είναι το Facility Location και πιο συγκεκριμένα την περίπτωση του 1–median problem. Το Facility Location για να λυθεί βέλτιστα στην γενική του περίπτωση κατατάσσεται στην NP-Hard κατηγορία προβλημάτων. Αυτό σημαίνει ότι τα προβλήματα αυτής της κατηγορίας δεν μπορούν να λυθούν σε πολυωνυμικό χρόνο. Ωστόσο το 1–median αποτελεί υποκατηγορία του  $p$ –median το οποίο μπορεί να λυθεί σε πολυωνυμικό χρόνο αν το δίκτυο είναι δέντρο. Ο στόχος της πτυχιακής εργασίας είναι να μελετήσουμε που θα τοποθετήσουμε την υπηρεσία στο δίκτυο και πόσο κοντά βρισκόμαστε στην βέλτιστη λύση χρησιμοποιώντας αλγεβρική θεωρία γράφων. Η αλγεβρική θεωρία γράφων είναι ένας κλάδος των μαθηματικών στον οποίον εφαρμόζονται αλγεβρικές μεθόδους σε προβλήματα γραφημάτων. Υπάρχουν τρεις κύριοι κλάδοι της αλγεβρικής θεωρίας γράφων που περιλαμβάνουν την χρήση της Γραμμικής Άλγεβρας (Linear Algebra), την χρήση της Θεωρίας των Ομάδων (Group Theory) και τη μελέτη των Μεταβλητών Γραφημάτων (Graph Invariants). Ο πρώτος κλάδος της θεωρίας των αλγεβρικών γραφημάτων περιλαμβάνει τη μελέτη γραφημάτων σε σχέση με τη γραμμική άλγεβρα. Ειδικότερα, μελετά το φάσμα του πίνακα γειτνίασης ή του πίνακα Laplacian ενός γραφήματος αυτό το τμήμα της θεωρίας των αλγεβρικών γραφημάτων ονομάζεται επίσης φασματική θεωρία γραφημάτων Spectral Graph Theory. Ο δεύτερος κλάδος της

θεωρίας των αλγεβρικών γραφημάτων περιλαμβάνει τη μελέτη γραφημάτων σε σχέση με τη θεωρία των ομάδων, ιδιαίτερα τις ομάδες αυτομορφισμού και τη γεωμετρική θεωρία των ομάδων. Η έμφαση δίνεται σε διάφορες οικογένειες γραφημάτων που βασίζονται στη συμμετρία (όπως συμμετρικά γραφήματα, γραφικά μεταβατικών κορυφών, γραφήματα μεταβατικής ακμής, γραφήματα μεταβατικής απόστασης) και στις σχέσεις συμπερίληψης μεταξύ αυτών των οικογενειών. Τέλος ο τρίτος κλάδος της θεωρίας των αλγεβρικών γραφημάτων αφορά τις αλγεβρικές ιδιότητες των μεταβλητών γραφημάτων και ειδικά το χρωματικό πολυώνυμο. Στην συγκρινόμενη πτυχιακή θα ασχοληθούμε κύριος με το κομμάτι γραμμικής άλγεβρας.

## Α΄.1 Βιβλιογραφική Έρευνα

Ένα δύσκολο κομμάτι στα σημερινά δίκτυα είναι η εύρεση του κόμβου που θα φιλοξενήσει την υπηρεσία ώστε ο κόμβος που θα επιλεγεί να ελαχιστοποιήσει τα κόστη τα οποία μπορούν να εκφράζουν ενεργεία, απόσταση κ.τ.λ. Το πρόβλημα το οποίο μελετάμε είναι το 1—median και για να λυθεί βέλτιστα στην γενική του περίπτωση κατατάσσεται στην NP-Hard κατηγορία προβλημάτων. Η βέλτιστη λύση του δίνεται από το Closeness centrality το οποίο μετρά την κεντρικότητα του κόμβου μέσα στο δίκτυο οριστική από τον Bavelas το 1950. Παρόλο που έχουν προταθεί αρκετοί προσεγγιστικοί αλγόριθμοι για τον υπολογισμό το 1—median. Έχει ιδιαίτερο ενδιαφέρον η μελέτη των δικτύων από την πλευρά της αλγεβρικής θεωρίας γράφων καθώς μας δείχνει συγκρινόμενες ιδιότητες για το δίκτυο οι οποίες μπορούν να χρησιμοποιηθούν προς όφελος μας. Το Degree centrality μετράει την κεντρικότητα του κόμβου ανάλογα με τον αριθμό των γειτόνων που έχει ο κόμβος. Το Eigenvector centrality μετράει την σημαντικότητα του κόμβου ανάλογα με την επιρροή του στους άλλους κόμβους. Ωστόσο το Eigenvector centrality μπορεί να υπολογιστεί και με εναλλακτικό τρόπο δηλαδή με την χρήση της Power method όπου δίνει καλύτερα χρονικά αποτελέσματα σε σχέση με Eigenvector centrality. Το Degree centrality και Eigenvector centrality δεν δίνουν πάντα την βέλτιστη λύση στο πρόβλημα άλλα προσεγγίσεις. Για αυτό τον λόγο θέλουμε να συγκρίνουμε την βέλτιστη λύση με τις άλλες για να δούμε πως αποδίδουν σε δίκτυα με γεωμετρικούς τυχαίους γράφους.

## Α΄.2 Αποτελέσματα και Συνεισφορά

Στα πλαίσια της εργασίας κατασκευάζονται δίκτυα τα οποία αποτελούνται από γεωμετρικούς τυχαίους γράφους. Συγκρίναμε τα μέτρα μεταξύ τους για να δούμε την αποδοτικότητα τους και ορίσαμε ως μετρό συγκρίσεις το συνολικό κόστος στο Κεφάλαιο Ε΄. Ιδιαίτερο ενδιαφέρον έχουν τα αποτελέσματα της Power method επειδή έχει πολύ καλύτερο χρόνο υπολογισμού και δίνει μια ικανοποιητική προσέγγιση σε σχέση με το Closeness centrality.

## Α΄.3 Αντιστοιχούμενα Κεφάλαια

Η συγκεκριμένη πτυχιακή εργασία έχει δομηθεί με τον εξής τρόπο. Στο Κεφάλαιο Β΄ βρίσκονται στοιχεία από την Γραμμική Άλγεβρα μαζί με παραδείγματα τα οποία είναι απαραίτητα για την υλοποίησή και την κατανόηση αυτής της πτυχιακής. Στο Κεφάλαιο Γ΄ δίνεται η έννοια του γραφήματος καθώς και των ιδιοτήτων του. Στο Κεφάλαιο Δ΄ ορίζουμε το πρόβλημα του 1-median, πως βρίσκουμε την λύση του καθώς και την μοντελοποίηση του. Στο Κεφάλαιο Ε΄ περιγράφονται λεπτομερώς πως έγιναν τα πειράματα καθώς και τα αποτελέσματα τους. Τέλος στο Κεφάλαιο Γ΄ βρίσκονται τα συμπεράσματα αυτής της εργασίας άλλα και ιδέες που θα μπορούσαν να υλοποιηθούν στο μέλλον.

## Κεφάλαιο Β΄

### Γραμμική Άλγεβρα

**Σ**ε αυτό το κεφάλαιο θα της πτυχιακής θα κάνουμε μια εισαγωγή στην γραμμική άλγεβρα. Η γραμμική άλγεβρα είναι ένας κλάδος των μαθηματικών ο οποίος βρίσκει πολλές εφαρμογές στην επιστήμη της Πληροφορικής. Σκοπός του κεφαλαίου είναι δούμε και θα ορίσουμε έννοιες όπως οι Πίνακες, οι Ορίζουσες, οι Ιδιοτιμές και τα Ιδιοδιανύσματα. Οι έννοιες αυτές είναι απαραίτητες για να καταλάβει το περιεχόμενο της πτυχιακής εργασίας.

#### Β΄.1 Χρήσιμα Στοιχεία Γραμμικής Άλγεβρας

Ένας **πίνακας** αποτελείται από  $mn$  πραγματικούς αριθμούς διατεταγμένη σε  $m$  γραμμές και  $n$  στήλες. Το στοιχείο που βρίσκεται στην γραμμή  $i$  και στην στήλη  $j$  του πίνακα  $A$  θα συμβολίζεται με  $a_{i,j}$  όπου ορίζεται στο  $\mathbb{F}$ . Ένας πίνακας μεγέθους  $m \times 1$  ονομάζεται **διάνυσμα-γραμμής** μεγέθους  $m$  και για κάθε  $i \in \{1, 2, \dots, m\}$ . Ομοίως ο πίνακας  $1 \times n$  ονομάζεται **διάνυσμα-στήλης** μεγέθους  $n$  και για κάθε  $j \in \{1, 2, \dots, n\}$ . Ο πίνακας  $m \times n$  ονομάζεται **τετραγωνικός** αν  $m = n$  [1].

$$A_{m,n} = \begin{bmatrix} a_{1,1} & a_{1,2} & \cdots & a_{1,n} \\ a_{2,1} & a_{2,2} & \cdots & a_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m,1} & a_{m,2} & \cdots & a_{m,n} \end{bmatrix}$$

Αν ο πίνακας  $A$  είναι τετραγωνικός τότε τα στοιχεία  $a_{i,i}$ ,  $i = 1, 2, \dots, n$  ονομάζονται **διαγώνια στοιχεία** του πίνακα. Ο τετραγωνικός πίνακας  $A = a_{i,j}$

λέγεται διαγώνιος αν  $a_{i,j} = 0$  για κάθε  $i \neq j$ . Στην περίπτωση αυτή γράφουμε  $A = D(a_{1,1}, \dots, a_{n,n})$ . Ένας τετραγωνικός πίνακας  $A = a_{i,j}$  λέγεται **άνω τριγωνικός** αν  $a_{i,j} = 0$  για κάθε  $i > j$  και **κάτω τριγωνικός** αν  $a_{i,j} = 0$  για κάθε  $i < j$ .

Αν ο  $A = a_{i,j}$  είναι  $m \times n$  πίνακας τότε ο  $m \times n$  πίνακας  $A^T = b_{i,j}$  που ορίζεται θέτοντας  $a_{i,j} = b_{j,i}$  για κάθε ζεύγος  $(i, j)$  με  $1 \leq i \leq m$  και  $1 \leq j \leq n$  λέγεται **ανάστροφος** (transpose) του  $A$ . Έτσι, ο ανάστροφος ενός διανύσματος-στήλης είναι ένα διάνυσμα-γραμμή και αντίστροφα. Γενικά, ο ανάστροφος  $A^T$  ενός  $m \times n$  πίνακα  $A$  είναι  $n \times m$  πίνακας που προκύπτει γράφοντας την πρώτη γραμμή του  $A$  ως πρώτη στήλη του  $A^T$ , τη δεύτερη γραμμή του  $A$  ως δεύτερη στήλη του  $A^T$  κ.ο.κ. . Αν λοιπόν  $r_1, r_2, \dots, r_m$  και  $c_1, c_2, \dots, c_n$  είναι οι γραμμές και οι στήλες του  $A$  αντιστοίχια τότε

$$A^T = \begin{bmatrix} r_1^T & r_2^T & \dots & r_m^T \end{bmatrix} = \begin{bmatrix} c_1^T \\ c_2^T \\ \vdots \\ c_n^T \end{bmatrix}$$

Επόμενος οι γραμμές του  $A^T$  είναι οι ανάστροφοι  $c_1^T, c_2^T, \dots, c_n^T$  των στηλών του  $A$  ενώ οι στήλες του  $A^T$  είναι οι ανάστροφοι  $r_1^T, r_2^T, \dots, r_m^T$  των γραμμών του  $A$ .

Ένας τετραγωνικός  $n \times n$  πίνακας  $A$  λέγεται **συμμετρικός** αν  $A = A^T$ .

Σύμφωνα με το θεμελιώδες θεώρημα της γραμμικής άλγεβρας [2], οι διαστάσεις του **Χώρου των στηλών** (column space) του πίνακα ισούται με τις διαστάσεις του **Χώρου των γραμμών** (row space), και η κοινή τιμή τους ονομάζεται **τάξη** (rank) του πίνακα. Η τάξη του πίνακα  $A$  συμβολίζεται με  $\text{rank} A$ . Για κάθε πίνακα  $A$ ,  $\text{rank} A = \text{rank} A^T$ . Εάν ο  $A$  και  $B$  είναι πίνακες της ίδιας τάξης, τότε  $\text{rank}(A + B) \leq \text{rank} A + \text{rank} B$ . Εάν ο  $A$  και  $B$  είναι πίνακες τέτοιοι ώστε να ορίζεται ο  $AB$ , τότε  $\text{rank} AB \leq \min\{\text{rank} A, \text{rank} B\}$ .

Έστω ότι ο  $A$  είναι ένας  $m \times n$  πίνακας. Το σύνολο όλων των διανυσμάτων  $x \in \mathbb{R}^n$  τέτοιο ώστε  $Ax = 0$  είναι εύκολο να δούμε ότι είναι υπό χώρος του  $\mathbb{R}^n$ . Αυτός ο υπό χώρος ονομάζεται **μηδενόχωρος** (ή nullspace) και τον συμβολίζουμε  $\mathcal{N}(A)$ . Οι διαστάσεις του  $\mathcal{N}(A)$  ονομάζεται **Μηδενικότητα** (ή nullity) του  $A$ . Αν ο  $A$  είναι  $m \times n$  πίνακας τότε η μηδενικότητα του  $A$  ισούται με  $n - \text{rank} A$  [3].

Μερικά παραδείγματα

- Ο πίνακας  $A = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 3 & 0 \\ 0 & 0 & -4 \end{bmatrix}$  είναι διαγώνιος και  $A = D(1, 3, -4)$ , ο πίνακας

$$B = \begin{bmatrix} 1 & -2 & 7 \\ 0 & -3 & 5 \\ 0 & 0 & 2 \end{bmatrix} \text{ είναι άνω τριγωνικός, ενώ ο πίνακας } C = \begin{bmatrix} 1 & 0 & 0 \\ 2 & 3 & 0 \\ 4 & 0 & -5 \end{bmatrix} \text{ είναι} \\ \text{κάτω τριγωνικός.}$$

- Ο  $D = \begin{bmatrix} 1 \\ 2 \\ 4 \end{bmatrix}$  είναι ένα διάνυσμα στήλης και ο  $E = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix}$  ένα διάνυσμα γραμμής.

- Ο ανάστροφος του πίνακα  $A = \begin{bmatrix} 1 & 3 \\ 2 & 7 \\ 12 & 5 \end{bmatrix}$  είναι ο πίνακας  $A^T = \begin{bmatrix} 1 & 2 & 12 \\ 3 & 7 & 5 \end{bmatrix}$

- Ο ανάστροφος ενός άνω τριγωνικού πίνακα είναι ο κάτω τριγωνικός δηλαδή ο πίνακας  $B = \begin{bmatrix} 1 & 4 & 2 \\ 0 & -3 & -5 \\ 0 & 0 & 2 \end{bmatrix}$  έχει ανάστροφο τον  $B^T = \begin{bmatrix} 1 & 0 & 0 \\ 4 & -3 & 0 \\ 2 & -5 & 2 \end{bmatrix}$ .

- Ο πίνακας  $A = \begin{bmatrix} 0 & 4 & 2 \\ 4 & -3 & -5 \\ 2 & -5 & 1 \end{bmatrix}$  είναι συμμετρικός ενώ ο  $B = \begin{bmatrix} 1 & 14 \\ 0 & 3 \end{bmatrix}$  δεν είναι συμμετρικός.

## Β'.2 Άθροισμα και Βαθμωτός Πολλαπλασιασμός

Έστω  $n, m$  δυο θετικοί αριθμοί. Αν ο  $A = (a_{i,j})$  και  $B = (b_{i,j})$  είναι δυο  $m \times n$  πίνακες που ορίζονται στο  $\mathbb{F}$ , τότε ο πίνακας  $A + B = (a_{i,j} + b_{i,j}) \in \mathbb{F}^{m \times n}$  λέγεται **άθροισμα** των  $A$  και  $B$ .

Αν ο  $A = (a_{i,j})$  είναι ένας  $m \times n$  πίνακας που ορίζεται στο  $\mathbb{F}$  και  $k \in \mathbb{F}$ , τότε ο πίνακας  $kA = (ka_{i,j}) \in \mathbb{F}^{m \times n}$  λέγεται **βαθμωτό γινόμενο** του  $k$  με τον  $A$  [4]. Τότε, αν

$$A_{m,n} = \begin{bmatrix} a_{1,1} & a_{1,2} & \cdots & a_{1,n} \\ a_{2,1} & a_{2,2} & \cdots & a_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m,1} & a_{m,2} & \cdots & a_{m,n} \end{bmatrix}, B_{m,n} = \begin{bmatrix} b_{1,1} & b_{1,2} & \cdots & b_{1,n} \\ b_{2,1} & b_{2,2} & \cdots & b_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ b_{m,1} & b_{m,2} & \cdots & b_{m,n} \end{bmatrix}$$

Επομένως

$$A_{m,n} + B_{m,n} = \begin{bmatrix} a_{1,1} + b_{1,1} & a_{1,2} + b_{1,2} & \cdots & a_{1,n} + b_{1,n} \\ a_{2,1} + b_{2,1} & a_{2,2} + b_{2,2} & \cdots & a_{2,n} + b_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m,1} + b_{m,1} & a_{m,2} + b_{m,2} & \cdots & a_{m,n} + b_{m,n} \end{bmatrix}$$

και

$$kA = \begin{bmatrix} ka_{1,1} & ka_{1,2} & \cdots & ka_{1,n} \\ ka_{2,1} & ka_{2,2} & \cdots & ka_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ ka_{m,1} & ka_{m,2} & \cdots & ka_{m,n} \end{bmatrix}$$

για κάθε  $k \in \mathbb{F}$ .

Ο  $m \times n$  πίνακας του οποίου όλα τα στοιχεία είναι ίσα με το 0 λέγεται **μηδενικός πίνακας** και τον συμβολίζουμε με  $\mathbf{0}$  [5]. Αν ο  $A = (a_{i,j})$  είναι ένας  $m \times n$  πίνακας τότε ο πίνακας  $-A = (-a_{i,j})$  ονομάζεται **αντίθετος** του  $A$ . Έτσι, αν

$$A_{m,n} = \begin{bmatrix} a_{1,1} & a_{1,2} & \cdots & a_{1,n} \\ a_{2,1} & a_{2,2} & \cdots & a_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m,1} & a_{m,2} & \cdots & a_{m,n} \end{bmatrix}$$

τότε ,

$$-A_{m,n} = \begin{bmatrix} -a_{1,1} & -a_{1,2} & \cdots & -a_{1,n} \\ -a_{2,1} & -a_{2,2} & \cdots & -a_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ -a_{m,1} & -a_{m,2} & \cdots & -a_{m,n} \end{bmatrix}.$$

Ένας τετραγωνικός πίνακας  $A$  λέγεται **αντισυμμετρικός** αν  $A^T = -A$ . Έτσι, ένας τετραγωνικός πίνακας  $A = a_{i,j}$  είναι αντισυμμετρικός αν ισχύει  $a_{i,j} = -a_{j,i}$  για κάθε ζεύγος δεικτών  $(i, j)$ .

Για παράδειγμα ο πίνακας  $A = \begin{bmatrix} 1 & -3 & 2 \\ 3 & 5 & 0 \\ -2 & 0 & -7 \end{bmatrix}$  είναι αντισυμμετρικός ενώ  $B = \begin{bmatrix} 1 & 4 & 0 \\ -4 & 3 & 2 \\ 0 & -3 & 2 \end{bmatrix}$  δεν είναι αντισυμμετρικός.

Παρακάτω οι βασικές ιδιότητες των πράξεων για τους πίνακες [6].

Έστω  $m, n$  δυο θετικοί ακέραιοι αριθμοί. Τότε, για  $A, B, C \in \mathbb{F}^{m \times n}$  και  $\kappa, \lambda \in \mathbb{F}$  ισχύουν τα εξής:

1.  $A + (B + C) = (A + B) + C$  ( Προσεταιριστική ιδιότητα της πρόσθεσης )
2.  $A + \mathbb{O}_{m \times n} = A$  ( Έπαρξη μηδενικού στοιχείου )
3.  $A + (-A) = \mathbb{O}_{m \times n}$  ( Έπαρξη αντίθετου στοιχείου )
4.  $A + B = B + A$  ( Μεταθετική ιδιότητα της πρόσθεσης )
5.  $(A + B)^T = A^T + B^T$
6.  $1A = A$
7.  $(\kappa\lambda)A = \kappa(\lambda A)$
8.  $(\kappa + \lambda)A = \kappa A + \lambda A$  ( Επιμεριστική ιδιότητα )
9.  $\kappa(A + B) = \kappa A + \kappa B$  ( Επιμεριστική ιδιότητα )

Ως συνέπεια των προηγούμενων ιδιοτήτων υποκύπτουν οι ακόλουθες ιδιότητες για το σύνολο των πινάκων.

1. Ο πίνακας  $-A$  είναι ο μοναδικός με την ιδιότητα  $A + (-A) = \mathbb{O} \in \mathbb{F}^{m \times n}$  ( Μοναδικότητα του αντιθέτου.)

$$2. -(A + B) = (-A) + (-B)$$

$$3. 0A = \mathbb{O} \in \mathbb{F}^{m \times n}$$

$$4. \kappa \mathbb{O} = \mathbb{O} \in \mathbb{F}^{m \times n}$$

$$5. (-\kappa)A = -\kappa A = \kappa(-A)$$

Όλες οι παραπάνω ιδιότητες μπορούν να αποδειχθούν σκοπός αυτού του κεφαλαίου είναι να δείξουμε τις βασικές ιδιότητες των πινάκων.

### B'.3 Πολλαπλασιασμός πινάκων

Σε αυτήν την παραγράφω θα δούμε πως γίνεται και ορίζεται ο πολλαπλασιασμός πινάκων. Στην προηγούμενη είδαμε πως γίνεται ο πολλαπλασιασμός ενός στοιχείου με ένα πίνακα  $A^{m \times n}$ .

Αν ο  $A = a_{i,k}$  είναι ένας πίνακας  $m \times n$  και  $B = b_{k,j}$  είναι ένας  $n \times r$ , τότε ορίζουμε ως **γινόμενο του πίνακα A με τον πίνακα B** και το συμβολίζουμε με  $A \bullet B$  ή με  $AB$  του  $m \times r$  πίνακα [7]. Του οποίου κάθε στοιχείο  $c_{i,j}$  είναι το άθροισμα των γινομένων των  $n$  στοιχείων της  $i$ -γραμμής του  $A$  με τα αντίστοιχα  $n$  στοιχεία της  $j$ -στήλης του  $B$ . Δηλαδή, ο  $C = c_{i,j}$  ( $C = AB$ ) που είναι τέτοιος ώστε

$$c_{i,j} = \sum_{k=1}^n a_{i,k} b_{k,j} = a_{i,1} b_{1,j} + a_{i,2} b_{2,j} + \cdots + a_{i,n} b_{n,j}$$

για κάθε ζεύγος δεικτών  $i, j$  με  $1 \leq i \leq m$  και  $1 \leq j \leq n$ .

**Παράδειγμα:** Έστω οι πίνακες  $A = \begin{bmatrix} 3 & -2 \\ 1 & 1 \end{bmatrix}$  και  $B = \begin{bmatrix} 1 & 3 & 2 \\ -3 & 0 & 1 \end{bmatrix}$ . Τότε

το γινόμενο  $AB$  ορίζεται και είναι ένας  $2 \times 3$  πίνακας με αποτελέσματα  $\begin{bmatrix} 9 & 9 & 4 \\ -2 & 3 & 3 \end{bmatrix}$ .

Για παράδειγμα, το στοιχείο  $-2$  στην  $(2,1)$ -θέση του πίνακα  $AB$  προκύπτει ως το άθροισμα  $1 \cdot 1 + 1 \cdot (-3)$ . Εδώ παρατηρούμε ότι ο πίνακας  $BA$  δεν ορίζεται.

Έστω  $n$  ένας θετικός ακέραιος αριθμός. Ο διαγώνιος  $n \times n$  πίνακας  $I_n = D(1, 1, \dots, 1)$  ονομάζεται **ταυτοτικός πίνακας** διάστασης  $n$  [8]. Όταν η διάσταση  $n$  όταν εννοείται γράφουμε  $I$ . Έτσι ο ταυτοτικός πίνακας  $I_n$  είναι ένας  $n \times n$  τετραγωνικός πίνακας που τα στοιχειά στην κυρία διαγώνιο είναι ίσα με το 1 και όλα τα υπόλοιπα ίσα με το 0, δηλαδή

$$I_n = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \end{bmatrix}.$$

Λέγεται ταυτοτικός πίνακας επειδή ο πολλαπλασιασμός αφήνει έναν αμετάβλητο πίνακα. Ο  $I_n$  είναι το **δέλτα του Kronecker** [9]. Αφού ορίσαμε τον πολλαπλασιασμό στους πίνακες ας δούμε τις βασικές ιδιότητες [10].

Αν  $m, n$  είναι θετικοί ακέραιοι και  $A, B, C \in \mathbb{F}^{m \times n}$  και  $\kappa, \lambda$  είναι πραγματικοί αριθμοί τότε:

1.  $A(BC) = (AB)C$  ( Προσεταιριστική ιδιότητα )
2.  $A(B + C) = AB + AC$  και  $(B + C)A = BA + CA$  ( Επιμεριστική ιδιότητα )
3.  $(\lambda A)(\kappa B) = (\lambda\kappa)AB$
4.  $AI_n = I_n A = A$  (Υπαρξη μονάδων )

Έστω τώρα  $A$  ένας τετραγωνικός πίνακας  $n \times n$  και  $k \in \mathbb{N}$  [11] . Τότε ορίζουμε την  $k$ -δύναμη του  $A$  να είναι ο  $n \times n$  πίνακας

$$A^k = \underbrace{A \cdot A \cdot \dots \cdot A}_{k\text{-times}}.$$

Αν το  $k = 0$  τότε  $A^0 = I_n$ .

Ένας τετραγωνικός πίνακας  $A$  διαστάσεων  $n \times n$  λέγεται **αντιστρεψίμος** αν υπάρχει τετραγωνικός πίνακας  $B$  διάστασης  $n \times n$ , τέτοιος ώστε να ισχύει  $A \cdot B = B \cdot A = I_n$  [12]. Σε αυτή την περίπτωση ο πίνακας  $B$  καλείται αντίστροφος του πίνακα  $A$  και γράφουμε  $B = A^{-1}$ . Άρα

$$A \cdot A^{-1} = A^{-1} \cdot A = I_n.$$

## B'.4 Ορίζουσες και Ίχνος

Οι ορίζουσες είναι μαθηματικά αντικείμενα που είναι πολύ χρήσιμα στην ανάλυση και τη λύση συστημάτων γραμμικών εξισώσεων. Οι ορίζουσες ορίζονται μόνο για τετραγωνικούς πίνακες [13]. Αν η ορίζουσα του πίνακα είναι ίση με το 0 τότε ο πίνακας ονομάζεται **Ιδιόμορφος πίνακας** ( ή singular matrix ) ενώ αν η ορίζουσα είναι ίση με το  $\pm 1$  τότε ο πίνακας ονομάζεται **unimodular**. Η ορίζουσα του πίνακα  $A$  συμβολίζεται με  $\det(A)$ . Η ορίζουσα για ένα πίνακα  $A^{2 \times 2}$  ορίζεται ως :

$$\det \begin{bmatrix} a & b \\ c & d \end{bmatrix} \equiv \begin{vmatrix} a & b \\ c & d \end{vmatrix} \equiv ad - bc .$$

Ενώ για έναν πίνακα  $A^{n \times n}$  η ορίζουσα μπορεί να επεκταθεί με την χρήση των **Ελάσσων οριζουσών**

$$\begin{aligned} A_{n,n} &= \begin{bmatrix} a_{1,1} & a_{1,2} & \cdots & a_{1,n} \\ a_{2,1} & a_{2,2} & \cdots & a_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n,1} & a_{n,2} & \cdots & a_{n,n} \end{bmatrix} \equiv \begin{vmatrix} a_{1,1} & a_{1,2} & \cdots & a_{1,n} \\ a_{2,1} & a_{2,2} & \cdots & a_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n,1} & a_{n,2} & \cdots & a_{n,n} \end{vmatrix} = \\ & a_{1,1} \begin{vmatrix} a_{2,2} & a_{2,3} & \cdots & a_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n,2} & a_{n,3} & \cdots & a_{n,n} \end{vmatrix} - a_{1,2} \begin{vmatrix} a_{2,1} & a_{2,3} & \cdots & a_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n,1} & a_{n,3} & \cdots & a_{n,n} \end{vmatrix} + \cdots \pm \\ & a_{1,n} \begin{vmatrix} a_{2,1} & a_{2,2} & \cdots & a_{2,(n-1)} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n,1} & a_{n,2} & \cdots & a_{n,(n-1)} \end{vmatrix} \end{aligned}$$

Παρακάτω ένα παράδειγμα με ορίζουσες

Έστω ο τετραγωνικός πίνακας  $A^{3 \times 3}$  με τα ακόλουθα στοιχεία.

$$A = \begin{bmatrix} 2 & -1 & 5 \\ 0 & 1 & 3 \\ 1 & 2 & 0 \end{bmatrix}$$

$$\text{Τότε } \det A = 2 \begin{vmatrix} 1 & 3 \\ 2 & 0 \end{vmatrix} - (-1) \begin{vmatrix} 0 & 3 \\ 1 & 0 \end{vmatrix} + 5 \begin{vmatrix} 0 & 1 \\ 1 & 2 \end{vmatrix} = 2(1 * 0 - 3 * 2) + 1(0 * 0 - 3 * 1) + 5(0 * 2 - 1 * 1) = 2(-6) + 1(-3) + 5(-1) = -20$$

Έστω ο  $A$  ένας τετραγωνικός πίνακας με διαστάσεις  $n$ . Ως **ίχνος** (ή *trace*) του πίνακα  $A$  ορίζεται το άθροισμα των στοιχείων της κύριας διαγωνίου του.

$$\text{trace} A = a_{1,1} + \cdots + a_{n,n}$$

Ως συνεπεία του ορισμού προκύπτει ότι αν  $A, B$  είναι πίνακες τέτοιο ώστε να ορίζονται οι πίνακες  $AB$  και  $BA$  τότε

$$\text{trace} AB = \text{trace} BA.$$

## Β'.5 Ιδιοτιμές και Ιδιοδιανύσματα

Στο προηγούμενο κεφάλαιο είδαμε πως υπολογίζουμε την τιμή της ορίζουσας ενός πίνακα. Σε αυτό το κεφάλαιο θα δούμε πως υπολογίζουμε τις ιδιοτιμές και τα ιδιοδιανύσματα ενός πίνακα [14].

Έστω ένας τετραγωνικός πίνακας  $A \in \mathbb{F}^{n \times n}$ , αν υπάρχουν  $\lambda \in \mathbb{F}$  και  $X \in \mathbb{F}^{n \times 1}$ , με το  $X \neq 0$  τέτοια ώστε  $AX = \lambda X$ , τότε το  $\lambda$  είναι μια **ιδιοτιμή** του πίνακα  $A$  και το  $X$  ένα **ιδιοδιάνυσμα** του πίνακα  $A$  που αντιστοιχεί στην ιδιοτιμή  $\lambda$ .

Η σχέση  $AX = \lambda X$  μπορεί να γράφει ως εξής:

$$AX = \lambda X \Leftrightarrow AX - \lambda X = 0 \Leftrightarrow (A - \lambda I_n)X = 0$$

Οι ακόλουθες ιδιότητες είναι ισοδύναμες. Έστω  $\lambda \in \mathbb{F}$  και  $A \in \mathbb{F}^{n \times n}$  τότε:

1. Το  $\lambda$  είναι μια ιδιοτιμή του πίνακα  $A$
2. Υπάρχει  $X \in \mathbb{F}^{n \times 1}, X \neq 0$  τέτοιο ώστε  $(A - \lambda I_n)X = 0$
3.  $\det(A - \lambda I_n) = 0$

Για να υπολογίσουμε της ιδιοτιμές και τα ιδιοδιανύσματα πρέπει να βρούμε την ορίζουσα  $\det(A - \lambda I_n)$ . Έστω  $A$  ένας τετραγωνικός  $n \times n$  πίνακας η ορίζουσα  $\det(A - \lambda I_n)$  δίνει ένα πολυώνυμο βαθμού  $n$  ως προς το  $\lambda$  το οποίο ονομάζεται **χαρακτηριστικό**

**πολυώνυμο** του  $A$  και η εξίσωση  $\det(A - \lambda I_n) = 0$  ονομάζεται **χαρακτηριστική εξίσωση** του  $A$  [15]. Με το θεμελιώδες θεώρημα της άλγεβρας η εξίσωση έχει  $n$  σύνθετες ρίζες και αυτές οι ρίζες ονομάζονται **ιδιοτιμές** του  $A$ . Οι ιδιοτιμές μπορεί να μην είναι όλες διακριτές. Ο αριθμός των φορών που μια ιδιοτιμή εμφανίζεται ως ρίζα της χαρακτηριστικής εξίσωσης ονομάζεται **αλγεβρική πολλαπλότητα**. Τότε μπορούμε να φτιάξουμε το χαρακτηριστικό πολυώνυμο ως έξεις:

$$\det(A - \lambda I_n) = (\lambda_1 - \lambda) \cdots (\lambda_n - \lambda).$$

οπού κάθε  $\lambda_i$  είναι ένας μιγαδικός αριθμός.

Η **γεωμετρική πολυπλοκότητα** της ιδιοτιμής  $\lambda$  του  $A$  ορίζεται ως μηδενόχωρος του  $A - \lambda I$ . Η γεωμετρική πολυπλοκότητα τις ιδιοτιμής δεν υπερβαίνει την αλγεβρική πολλαπλότητα του. Εάν  $A$  και  $B$  είναι πινάκες  $m \times n$  και  $n \times m$  αντίστοιχα, όπου  $m \geq n$ , τότε οι ιδιοτιμές του  $AB$  θα είναι ίδιες με τις ιδιοτιμές του  $BA$ , μαζί με το 0 με (πιθανώς περαιτέρω) πολλαπλότητα  $m - n$  [15][16].

Αν  $\lambda_1, \dots, \lambda_n$  είναι οι ιδιοτιμές του  $A$ , τότε  $\det A = \lambda_1 \cdots \lambda_n$ , ενώ το  $\text{trace} A = \lambda_1 + \dots + \lambda_n$ .

Έστω ότι θέλουμε να βρούμε τις ιδιοτιμές και τα ιδιοδιανύσματα του  $A$ . Όπου

$$A = \begin{bmatrix} 2 & 1 & 0 \\ 0 & 1 & -1 \\ 0 & 2 & 4 \end{bmatrix}$$

Έχουμε ότι το χαρακτηριστικό εξίσωση του  $A$  μας δίνει το παρακάτω πολυώνυμο,

$$\begin{aligned} \det(A - \lambda I_3) &= \det \begin{bmatrix} 2 - \lambda & 1 & 0 \\ 0 & 1 - \lambda & -1 \\ 0 & 2 & 4 - \lambda \end{bmatrix} = \\ (2 - \lambda) \begin{vmatrix} 1 - \lambda & -1 \\ 2 & 4 - \lambda \end{vmatrix} - 1 \begin{vmatrix} 0 & -1 \\ 0 & 4 - \lambda \end{vmatrix} + 0 \begin{vmatrix} 0 & 1 - \lambda \\ 0 & 2 \end{vmatrix} &= \\ (2 - \lambda)((1 - \lambda)(4 - \lambda) + 2) &= (2 - \lambda)^2(3 - \lambda) \end{aligned}$$

Οι λύσεις του χαρακτηριστικού πολυωνύμου θα μας δώσουν τις ιδιοτιμές του  $A$  που είναι  $\lambda = 2, \lambda = 3$ . Για να βρούμε τα αντίστοιχα ιδιοδιανύσματα θα πρέπει να

αντικαταστήσουμε την ιδιοτιμή στην σχέση  $(A - \lambda I_n)X = 0$ .

Για  $\lambda = 2$  και  $X = \begin{bmatrix} x \\ y \\ z \end{bmatrix}$  έχουμε:

$$\begin{bmatrix} 0 & 1 & 0 \\ 0 & -1 & -1 \\ 0 & 2 & 2 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \Leftrightarrow$$

$$\begin{aligned} y &= 0 \\ -y - z &= 0 \\ 2y + 2z &= 0 \end{aligned}$$

που ισοδυναμεί με το

$$\begin{aligned} y &= 0 \\ y + z &= 0 \end{aligned}$$

Οι λύσεις του συστήματος είναι  $(x, 0, 0), x \in \mathbb{R}$ . Άρα τα ιδιοδιανύσματα που αντιστοιχούν στην ιδιοτιμή  $\lambda = 2$  είναι τα  $\begin{bmatrix} x \\ 0 \\ 0 \end{bmatrix}, x \in \mathbb{R} - \{0\}$ .

Για  $\lambda = 3$  έχουμε

$$\begin{bmatrix} -1 & 1 & 0 \\ 0 & -2 & -1 \\ 0 & 2 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \Leftrightarrow$$

$$\begin{aligned} -x + y &= 0 \\ -2y - z &= 0 \\ 2y + z &= 0 \end{aligned}$$

ισοδυναμεί με το

$$\begin{aligned} x - y &= 0 \\ 2y + z &= 0 \end{aligned}$$

Οι λύσεις του συστήματος είναι  $(x, x, -2x), x \in \mathbb{R}$ . Άρα τα ιδιοδιανύσματα που αντιστοιχούν στην ιδιοτιμή  $\lambda = 3$  είναι τα  $\begin{bmatrix} x \\ x \\ -2x \end{bmatrix}, x \in \mathbb{R} - \{0\}$ .

## Κεφάλαιο Γ΄

### Θεωρία Γραφημάτων

**Σ**ε αυτό το κεφάλαιο θα μελετήσουμε την Θεωρία Γραφημάτων ή Θεωρία Γράφων μαζί με τα βασικά χαρακτηριστικά της. Είναι γνωστό επίπεδο τον διακριτών μαθηματικών με αρκετές εφαρμογές στον τομέα πληροφορικής. Με τους γράφους μπορούν να μοντελοποιηθούν πολλές διαφορετικές δομές όπως τα δίκτυα υπολογιστών.

#### Γ΄.1 Βασικές έννοιες της Θεωρία Γραφημάτων

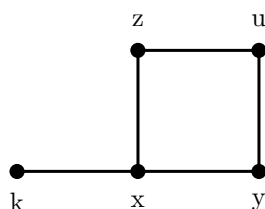
**Ορισμός Γ΄.1** Ένα γράφημα  $G$  είναι ένα ζεύγος συνόλων  $(V, E)$  όπου  $V$  είναι ένα πεπερασμένο μη-κενό σύνολο  $n$  στοιχείων και  $E$  ένα πεπερασμένο σύνολο  $m$  με ζεύγη από στοιχεία του συνόλου  $V$ .

Τα στοιχεία του συνόλου  $V$  ονομάζονται κόμβοι (vertices) ενώ τα στοιχεία του συνόλου  $E$  ονομάζονται ακμές (edges). Τις ακμές ενός γραφήματος  $G = (V, E)$  θα τις συμβολίζουμε  $xy$  όπου  $x, y$  είναι οι κόμβοι του γραφήματος  $G$  [17].

Επομένως, ένα γράφημα  $G = (V, E)$  είναι ένα μαθηματικό αντικείμενο που ορίζεται αυστηρά από το σύνολο των κόμβων  $V(G)$  και το σύνολο των ακμών  $E(G)$ . Για παράδειγμα :

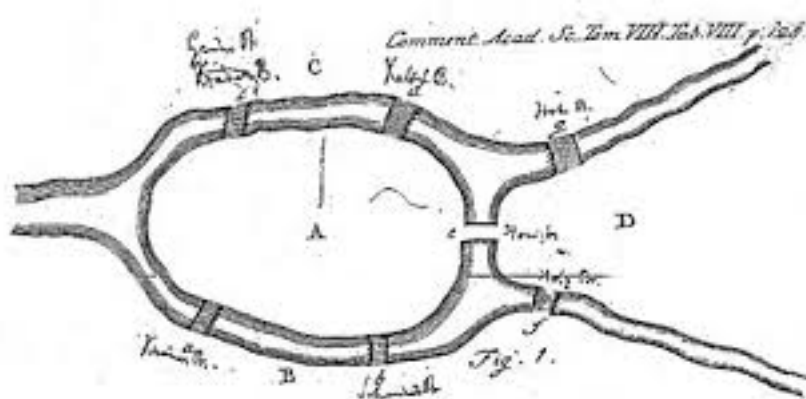
- $V(G) = \{x, y, z, u, k\}$
- $E(G) = \{xy, xz, yu, uz, xk\}$

Ορίζουν το γράφημα  $G$  που αποτελείώστε απο  $n = 5$  κόμβους και 5 ακμές. Παρακάτω η απεικόνιση του.



## Γ'.2 Ιστορική ανάδρομη

**Πότε και πως ξεκίνησαν όλα.** Η θεωρία γραφημάτων θεωρείται ότι ξεκίνησε τον 18<sup>ο</sup> αιώνα, όταν ο Ελβετός μαθηματικός Leonard Euler έλυσε το πρόβλημα των 7 γεφυρών του Königsberg. Ο ποταμός Pregel του διασχίζει την πόλη και τη χώριζε σε τέσσερα τμήματα γης [18].

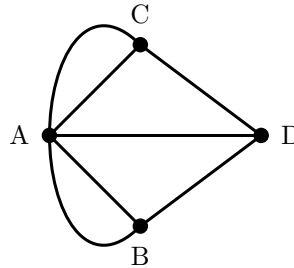


Σχήμα Γ'.1: Ο χάρτης των 7 γεφυρών του Königsberg τον 17<sup>ο</sup> αιώνα.

Το πρόβλημα ήταν να βρεθεί μια διαδρομή όπου ο διαβάτης ξεκινάει από μια χερσαία περιοχή να επισκέπτη τις άλλες τρεις περιοχές και να καταλήγει πάλι στην περιοχή που ξεκίνησε διασχίζοντας και τις 7 γέφυρες με τον περιορισμό ότι κάθε γέφυρα θα διασχιστεί ακριβώς μία φορά.

Το *Πρόβλημα των Γεφυρών του Königsberg* δεν είχε λυθεί μέχρι που ο Euler απέδωσε μία μαθηματική λύση το 1736. Ο Euler ονομάζοντας τις χερσαίες περιοχές

της πόλης  $A, B, C$  και  $D$ , μοντελοποιεί το πρόβλημα στο γράφημα  $G$  όπως βλέπουμε στο παρακάτω σχήμα.



Σχήμα Γ'.2: Μοντελοποίηση του προβλήματος των 7 γεφυρών του Königsberg με την μορφή γραφήματος.

Ο Euler απέδειξε ότι το πρόβλημα των Γεφυρών του Königsberg δεν έχει λύση. Ο Euler απέδειξε γενικότερα τότε γενικά ένα γράφημα  $G$  έχει μια τέτοια διαδρομή, η οποία ονομάζεται διαδρομή Euler, ενώ το γράφημα  $G$  ονομάζεται γράφημα Euler.

### Γ'.3 Τύποι Γραφημάτων

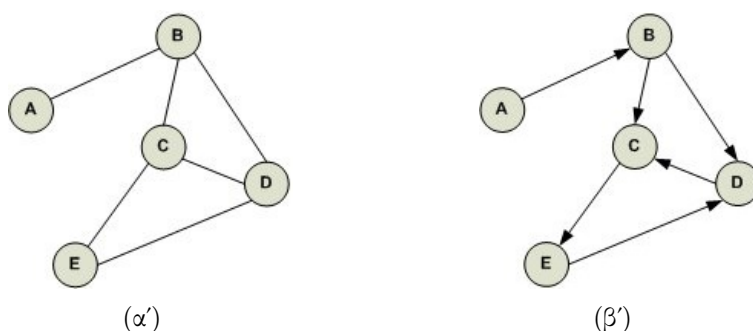
Είδαμε πως ξεκίνησε η θεωρία των γραφημάτων σε αυτή την παράγραφο θα αναφέρουμε μερικούς τύπους των γραφημάτων [19].

**Μη-κατευθυνόμενα γραφήματα:** Η θεώρηση του συνόλου  $E$  του ζεύγους  $(V, E)$  ως ενός συνόλου μη-διατεταγμένων ζευγών κόμβων ορίζει έναν τύπο γραφημάτων που ονομάζουμε μη-κατευθυνόμενα γραφήματα (undirected graphs).

**Κατευθυνόμενα γραφήματα:** Εάν θεωρήσουμε το σύνολο ακμών  $E$  ως ένα σύνολο διατεταγμένων ζευγών κόμβων, τότε το ζεύγος συνόλων  $(V, E)$  ορίζει τον τύπο γραφημάτων που ονομάζουμε κατευθυνόμενα γραφήματα (directed graphs).

Η αναπαράσταση ενός κατευθυνόμενου γραφήματος  $G$ , σε σχέση με αυτή ενός μη-κατευθυνόμενου, διαφοροποιείται μόνον ως προς τις ακμές, όπου τώρα χρησιμοποιούμε τόξα, αντί για γραμμές για να δείξουμε τη διάταξη των κόμβων μίας κατευθυνόμενης ακμής για παρτίδα στο σχήμα Γ'.3β' το τόξο ξεκάνει από τον κόμβο  $A$  και τελειώνει στον κόμβο  $B$ .

**Συνεκτικό γράφημα.** Ένα μη-κατευθυνόμενο γράφημα ονομάζεται συνεκτικό (connected), αν κάθε κόμβος του γραφήματος  $G$  συνδέεται με κάθε άλλο κόμβο μέσω

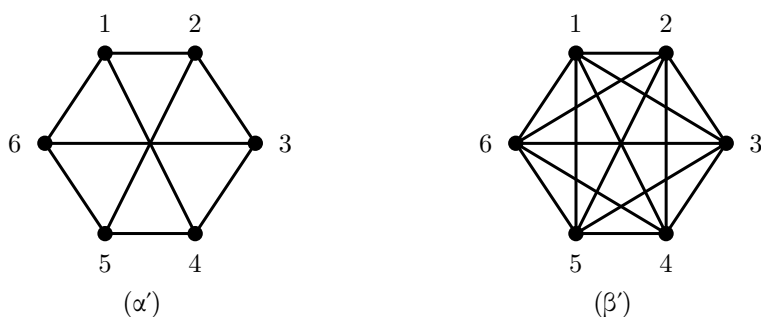


Σχήμα Γ.3: Το Γ'.3α' ένα μη-κατευθυνόμενο γραφήμα και το Γ'.3β' ένα κατευθυνόμενο γραφήμα.

ενός μονοπατιού.

**Πλήρες γράφημα:** Ένα γράφημα  $G$  που αποτελείται από  $n$  κόμβους ονομάζεται πλήρες (complete), αν ο κάθε ζευγάρι κόμβων έχει μια ακμή που να τους συνδέει [20]. Συμβολίζεται με  $K_n$  και ο αριθμός  $L$  των πλευρών ισούται με:

$$L = n(n - 1)/2. \quad (\Gamma'.1)$$

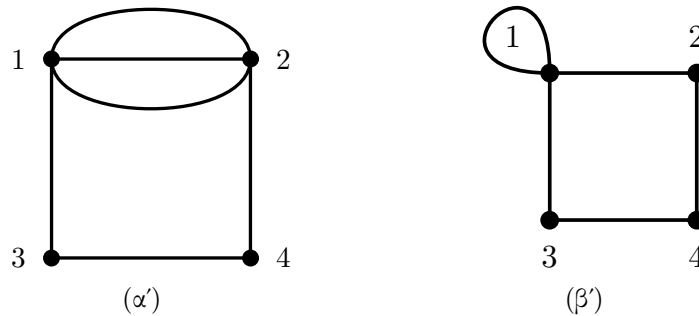


Σχήμα Γ.4: Το Γ'.4α' είναι ένα συνεκτικό γράφημα και το Γ'.4β' είναι ένα πλήρες γράφημα  $K_6$ .

Στην παραπάνω εικόνα βλέπουμε ένα παράδειγμα δυο γραφημάτων. Το γράφημα Γ'.4β' έχει 6 κόμβους και είναι πλήρες όποτε έχει αριθμό πλευρών  $L = 15$  όπου δίνεται από τον τύπο Γ'.1. Ένα πλήρες γράφημα είναι συνεκτικό, το αντίστροφο δεν ισχύ πάντα όπως στην περίπτωση του γραφήματος Γ'.4α'.

**Πολυγράφημα:** Ένα γράφημα  $G$  ονομάζεται πολυγράφημα (multigraph), αν έχει παράλληλες ακμές [21].

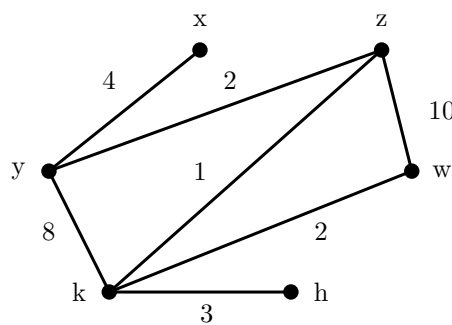
**Ψευδογράφημα:** Ένα γράφημα  $G$  ονομάζεται ψευδογράφημα (pseudograph), αν διαθέτει βρόχος. Δηλαδή αν διαθέτει ακμές της μορφής  $xx \in E(G)$  που ονομάζονται βρόχοι (loops) [22].



Σχήμα Γ'.5: Το Γ'.5α' είναι ένα πολυγράφημα και το Γ'.5β' είναι ένα ψευδογράφημα.

**Απλό γράφημα:** Ένα γράφημα  $G$  ονομάζεται απλό (simple), εάν δεν διαθέτει βρόχους και παράλληλες ακμές [23].

**Έμβαρo γράφημα:** Ένα γράφημα  $G$  ονομάζεται έμβαρo (weighted graph), εάν σε κάθε ακμή  $e \in E(G)$  έχει ανατεθεί ένας πραγματικός αριθμός  $w(e)$ , που ονομάζεται βάρος της ακμής  $e$ . Ένα έμβαρo γράφημα μπορεί να είναι μη-κατευθυνόμενο ή κατευθυνόμενο.



Σχήμα Γ'.6: Παράδειγμα ενός έμβαρo μη-κατευθυνόμενου γραφήματος.

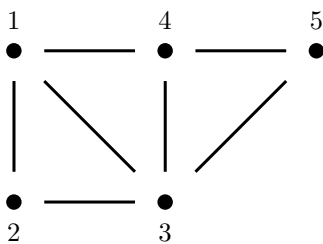
Στα πλαίσια της πτυχιακής μας ενδιαφέρουν κύριος τα άπλα μη-κατευθυνόμενοι γραφήματα.

## Γ'.4 Πίνακας Γειτνίασης

Σε αυτήν την παράγραφο θα ορίσουμε τον πίνακα γειτνίασης, θα δούμε πως συνδέεται με τα γραφήματα καθώς και τις ιδιότητες του.

**Ορισμός Γ'.2** Έστω  $G$  ένα γράφημα με  $V(G) = \{1, \dots, n\}$  και  $E(G) = \{e_1, \dots, e_m\}$ . Ο πίνακας **γειτνίασης** (ή *Adjacency Matrix*) [24] του  $G$  συμβολίζεται με  $\mathcal{A}(G)$ , είναι ένας  $n \times n$  πίνακας και ορίζεται ως. Οι γραμμές και στήλες του  $\mathcal{A}(G)$  είναι ευρετηριασμένες από το  $V(G)$ . Αν  $i \neq j$  τότε η  $(i, j)$ -καταχώρηση του  $\mathcal{A}(G)$  είναι 0 για τις πλευρές  $i$  και  $j$  που είναι μη γειτονικές, και η  $(i, j)$ -καταχώρηση είναι 1 αν οι πλευρές  $i$  και  $j$  είναι μη γειτονικές. Η  $(i, i)$ -καταχώρηση του  $\mathcal{A}(G)$  είναι 0 για  $i = 1, \dots, n$ . Για ευκολία συμβολίζουμε  $\mathcal{A}(G)$  με  $\mathcal{A}$ .

Έστω το γράφημα  $G$ :



Τότε

$$\mathcal{A}(G) = \begin{bmatrix} 0 & 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 \end{bmatrix}.$$

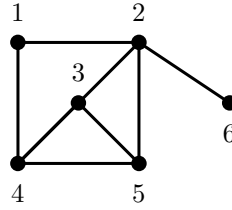
Ο πίνακας  $\mathcal{A}$  είναι συμμετρικός όποτε  $\mathcal{A} = \mathcal{A}^T$  με μηδενικά στοιχεία στην κύρια διαγώνιο. Το στοιχείο που βρίσκεται στην θέση  $(i, j)$  του πίνακα  $\mathcal{A}^k$  είναι ο αριθμός που δείχνει πόσα διαφορετικά μονοπάτια απόστασης  $k$  υπάρχουν από τον  $i$  στον  $j$ .

Έστω ένα συνδεδεμένο γράφημα  $G$  με πλευρές  $\{1, \dots, n\}$ . Η **απόσταση** (ή *distance*)  $d(i, j)$  ανάμεσα στις πλευρές  $i$  και  $j$  ορίζεται ως το ελάχιστο μήκος του μονοπατιού  $(i, j)$ . Ορίζουμε ότι  $d(i, i) = 0$ . Η μεγαλύτερη τιμή που μπορεί να πάρει το  $d(i, j)$  είναι η **διάμετρος** του  $G$  και συμβολίζεται με  $\rho$ .

Έστω ένα συνδεδεμένο γράφημα  $G$  με πλευρές  $\{1, \dots, n\}$  και  $\mathcal{A}$  ο πίνακας γειτνίασης του γραφήματος  $G$ . Αν  $i, j$  είναι πλευρές του  $G$  με  $d(i, j) = m$  τότε οι πίνακες  $I, \mathcal{A}, \dots, \mathcal{A}^m$  είναι γραμμικός ανεξάρτητοι [25].

Έστω ένα συνδεδεμένο γράφημα  $G$  με πλευρές  $\{1, \dots, n\}$  και  $\mathcal{A}$  ο πίνακας γειτνίασης του γραφήματος  $G$ . Ο αριθμός των γειτόνων της πλευράς  $i$  με  $1 \leq i \leq n$ , δίνεται από το άθροισμα της  $i$  γραμμής του πίνακα  $\mathcal{A}$ , συμβολίζεται με  $d_i$  και ονομάζεται ο **βαθμός** (ή degree) της πλευράς  $i$ . Ο  $n \times n$  πίνακας που περιέχει στην κυρία διαγώνια τον βαθμό τις πλευράς  $i$  για κάθε πλευρά του γραφήματος ονομάζεται **degree matrix**.

Έστω ότι έχουμε το παρακάτω γράφημα  $G$  ο πίνακας  $\mathcal{A}$  είναι ο πίνακας γειτνίασης και ο  $\mathcal{D}$  ο degree matrix .



$$\mathcal{A} = \begin{bmatrix} 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix}, \mathcal{D} = \begin{bmatrix} 2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 4 & 0 & 0 & 0 & 0 \\ 0 & 0 & 3 & 0 & 0 & 0 \\ 0 & 0 & 0 & 3 & 0 & 0 \\ 0 & 0 & 0 & 0 & 3 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

**Παρατήρηση** ο πίνακας  $\mathcal{D}$  μπορεί να προκύψει και από τον πίνακα  $\mathcal{A}^2$  κρατώντας την κυρία διαγώνιο του [26] .

**Ορισμός Γ'.3 A shortest path:** Για κάθε σύνδεσμο  $e_l = i \rightarrow j \in \mathcal{L}$  από τον κόμβο  $i \in \mathcal{N}$  στον κόμβο  $j \in \mathcal{N}$  στο δίκτυο αναθέτουμε ένα βάρος ζεύξης  $w(e_l) = w_{i,j}$ , ένα μη αρνητικό πραγματικό αριθμό το οποίο ποσοτικοποιεί μια ιδιότητα του συνδέσμου όπως η καθυστέρηση ή απαιτούμενη ενέργεια. Το σύνολο όλων των βαρών των ζεύξεων ονομάζεται δομή βάρους (link weight structure) του  $G$ . Θεωρούμε μόνο πρόσθετα τα βάρη των ζεύξεων έτσι ώστε το βάρος μιας διαδρομής  $P$  να είναι  $w(P) = \sum_{e_l \in P} w(e_l)$ . Το  $w(P)$  ισούται με το άθροισμα των βαρών των συνδέσμων του (μονοπατιού)  $P$ . Το

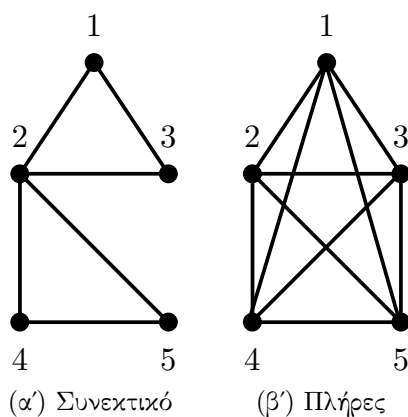
*shortest path* ( $P_{A \rightarrow B}^*$ ) από τον  $A$  στον  $B$  είναι το μονοπάτι με το ελάχιστο βάρος, όπως  $w(P_{A \rightarrow B}^*) \leq w(P_{A \rightarrow B})$  για όλα τα  $P_{A \rightarrow B}$ . Εάν όλα τα βάρη των ζεύξεων είναι ίσα με  $w_{ij} = 1$  το *shortest path* είναι το μικρότερο σε hops μονοπάτι και το  $w(P_{A \rightarrow B}^*)$  ονομάζεται επίσης απόσταση (*distance*) μεταξύ του  $A$  και του  $B$  [27].

Για παράδειγμα στο σχήμα 'Α.6 το *shortest path* από τον κόμβο  $x$  στον  $h$  ισούται με  $w(P_{x \rightarrow h}) = 10$  που αντιστοιχεί στην διαδρομή  $x \rightarrow y \rightarrow z \rightarrow k \rightarrow h$ . Αν όλα τα βάρη των ζεύξεων στο σχήμα 'Α.6 είναι ίσα με  $w(e_i) = 1$  τότε το *shortest path* από τον κόμβο  $x$  στον  $h$  ισούται με τον ελάχιστο αριθμό των hops δηλαδή  $w(P_{x \rightarrow h}) = 3$  και αντιστοιχεί στην διαδρομή  $x \rightarrow y \rightarrow k \rightarrow h$ .

### Ιδιοτιμές και ιδιότητες του Πίνακας Γειτνίασης

**Θεώρημα Γ'.1** Το θεώρημα του Gerschgorin [28] όταν εφαρμοσθή στον πίνακα γειτνίασης δηλώνει ότι οποιαδήποτε ιδιοτιμή του  $\mathcal{A}$  βρίσκεται στο διάστημα  $[-d_{max}, d_{max}]$ . Ως εκ τούτου,  $\lambda_1 \leq N - 1$  και αυτό το μέγιστο επιτυγχάνεται στο πλήρες γράφημα. Όπου  $d_{max}$  η μεγαλύτερη τιμή του *degree matrix*,  $\lambda_1$  η μεγαλύτερη ιδιοτιμή και  $N$  ο αριθμός των κορυφών.

Θα εφαρμόσουμε το παραπάνω θεώρημα στο παρακάτω γραφήματα.



Σχήμα Γ'.7: Δυο διαφορετικά είδη γραφημάτων.

Για το γράφημα Γ'.7α' έχουμε:

$$\mathcal{A} = \begin{bmatrix} 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 \end{bmatrix}, \mathcal{D} = \begin{bmatrix} 2 & 0 & 0 & 0 & 0 \\ 0 & 4 & 0 & 0 & 0 \\ 0 & 0 & 2 & 0 & 0 \\ 0 & 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 0 & 2 \end{bmatrix}$$

Για να βρούμε τις ιδιοτιμές του πίνακα  $\mathcal{A}$  πρέπει να υπολογίσουμε την χαρακτηριστική εξίσωση  $\det(\mathcal{A} - \lambda I_n) = 0$ . Το χαρακτηριστικό πολυώνυμο του  $\mathcal{A}$  ισούται με  $\lambda^5 - 6\lambda^3 - 4\lambda^2 + 5\lambda + 4$  και οι ρίζες του είναι  $-1.5616, -1, -1, 1, 2.5616$ . Όλες οι ρίζες του βρίσκονται μέσα στο διάστημα  $[-4, 4]$  όπου  $d_{max} = 4$ .

Ενώ το γράφημα Γ'.7β' έχουμε:

$$\mathcal{A} = \begin{bmatrix} 0 & 1 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 & 1 \\ 1 & 1 & 0 & 1 & 1 \\ 1 & 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 1 & 0 \end{bmatrix}, \mathcal{D} = \begin{bmatrix} 4 & 0 & 0 & 0 & 0 \\ 0 & 4 & 0 & 0 & 0 \\ 0 & 0 & 4 & 0 & 0 \\ 0 & 0 & 0 & 4 & 0 \\ 0 & 0 & 0 & 0 & 4 \end{bmatrix}$$

Ακολουθώντας την προηγούμενη διαδικασία έχουμε για το γράφημα Γ'.7β' ότι το χαρακτηριστικό πολυώνυμο του  $\mathcal{A}$  ισούται με  $\lambda^5 - 10\lambda^3 - 20\lambda^2 - 15\lambda - 4$ . Για την μεγαλύτερη ιδιοτιμή του  $\mathcal{A}$  ισχύει ότι  $\lambda_1 \leq N - 1$  όπου  $N$  ο αριθμός των άκμων. Έχοντας έτσι ρίζες  $-1, -1, -1, 4$  που βρίσκονται στο διάστημα  $[-4, 4]$ .

**Θεώρημα Γ'.2** Ο αριθμός των διακριτών ιδιοτιμών του πίνακα γειτνίασης  $\mathcal{A}$  είναι τουλάχιστον ίσος με το  $\rho + 1$ , όπου  $\rho$  η διάμετρος του γραφήματος.

Αν εφαρμόσουμε το θεώρημα Γ'.2 [29] στο γράφημα Γ'.7α' χωρίς να έχουμε υπολογίσει τις ιδιοτιμές του ξέρουμε ότι ο αριθμός  $n$  των διακριτών ιδιοτιμών θα είναι μεγαλύτερος ή ίσος με το  $\rho + 1$ . Δηλαδή  $n \geq \rho + 1$  και  $\rho = 2$  επόμενος  $n \geq 3$ . Ενώ για το γράφημα Γ'.7β' έχουμε  $\rho = 1$  και  $n \geq 2$ . Αν υπολογίσουμε τις ιδιοτιμές βλέπουμε ότι στο γράφημα Γ'.7α' ισχύει η ανισότητα ενώ για το γράφημα Γ'.7β' ισχύει η ισότητα.

Η μεγαλύτερη ιδιοτιμή  $\lambda_1$  [30] του πίνακα γειτνίασης  $\mathcal{A}$  εμφανίζεται σε πολλές εφαρμογές. Πιο συγκεκριμένα οι δυναμικές διεργασίες του γραφήματος, ο ανάστροφος της μεγαλύτερης ιδιοτιμής  $\lambda_1$  χαρακτηρίζει ένα κατώφλι της μετάβασης της φάσης τόσο της εξάπλωσης του ιού (VanMieghem2009) όσο και τον συγχρονισμό των συζευγμένων ταλαντωτών (Restrepo2005) στα δίκτυα.

Η διάφορα ανάμεσα στην μεγαλύτερη ιδιοτιμή  $\lambda_1$  και την δεύτερη μεγαλύτερη  $\lambda_2$ , ονομάζεται Spectral gap [31] και δεν είναι πότε μεγαλύτερη από το  $N$ :

$$\lambda_1 - \lambda_2 \leq N \quad (\Gamma'.2)$$

Η ισότητα στην εξίσωση  $\Gamma'.2$  εμφανίζεται σε περίπτωση που έχουμε ένα πλήρες γράφημα. Εάν αφαιρέσουμε μια πλευρά από το πλήρες γράφημα, το Spectral gap μειώνεται τουλάχιστον κατά 1. Το Spectral gap παίζει σημαντικό ρόλο στις δυναμικές διεργασίες του γραφήματος και χαρακτηρίζει το **robustness** του γραφήματος λόγω της σχέσης του με την αλγεβρική συνδεσιμότητα.

## Κεφάλαιο Δ΄

# Μοντελοποίηση του 1-median problem

$\Sigma$  τα προηγούμενα κεφαλαία είδαμε τα βασικά στοιχεία από την γραμμική άλγεβρα και την θεωρία γραφημάτων. Στόχος αυτού του κεφαλαίου είναι να κατανοήσουμε το 1-median problem και να το μοντελοποιήσουμε με την χρήση αλγεβρικής θεωρίας γράφων.

### Δ΄.1 Πως βρίσκουμε 1-median problem

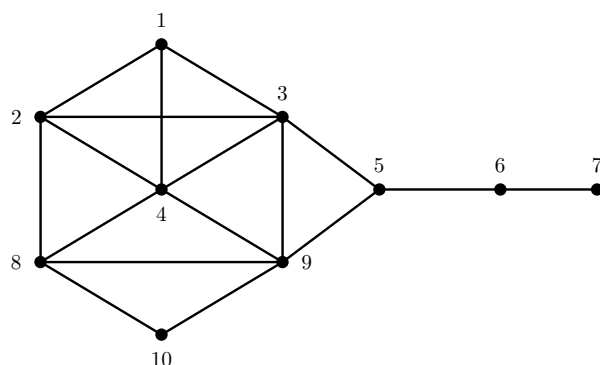
Το 1-median problem είναι μια ιδική υποκατηγορία του προβλήματος k-median problem το οποίο για να λυθεί βέλτιστα σε ένα γενικό γράφημα κατατάσσεται στην NP-Hard [32] κατηγορία προβλημάτων. Το 1-median λύνεται σε πολυωνυμικό χρόνο αν το γράφημα είναι δέντρο. Το 1-median problem έχει σκοπό να εντοπίσει τη θέση του κόμβου στο δίκτυο, έτσι ώστε το συνολικό κόστος να είναι ελάχιστο. Η βέλτιστη λύση του δίνεται από το Closeness centrality [33] που μετρά την κεντρικότητα του κόμβου. Υπάρχουν και άλλα μέτρα όπως το Degree centrality που μετρά την σημαντικότητα του κόμβου ανάλογα με αριθμό των γειτόνων. Το Eigenvector centrality το οποίο μετρά την επιρροή που έχει ο κόμβος στο δίκτυο. Το Degree centrality και Eigenvector centrality δεν δίνουν πάντα την βέλτιστη στο πρόβλημα του 1-median.

Το **Closeness centrality** ενός κόμβου είναι η μέτρηση της κεντρικότητας του κόμβου στο δίκτυο. Υπολογίζεται από το άθροισμα των shortest paths μεταξύ του κόμβου και όλων των άλλων κόμβων στο δίκτυο. Όποτε όσο πιο κεντρικά βρίσκετε ο

κόμβος τόσο πιο κοντά βρίσκεται στους υπολοίπους κόμβους. Το Closeness centrality οριστικό από τον Bavelas το 1950 ως:

$$C(x) = \sum_y \frac{1}{d(y, x)} \quad (\Delta'.1)$$

Όπου  $d_{y,x}$  το shortest path μεταξύ του κόμβου  $x$  και  $y$ . Ας υπολογίσουμε το Closeness centrality στο παρακάτω γράφημα έχουμε τα ακόλουθα αποτελέσματα.



Σχήμα Δ'.1: Ένα μη-κατευθυνόμενο γράφημα.

| Closeness centrality |        |
|----------------------|--------|
| Κόμβος               | Score  |
| 1                    | 0.0526 |
| 2                    | 0.0588 |
| 3                    | 0.0714 |
| 4                    | 0.0625 |
| 5                    | 0.0667 |
| 6                    | 0.0476 |
| 7                    | 0.0345 |
| 8                    | 0.0588 |
| 9                    | 0.0714 |
| 10                   | 0.0500 |

Πίνακας Δ'.1: Τα αποτελέσματα του Closeness centrality για το σχήμα Δ'.1 .

Όπως βλέπουμε ο κόμβος 3 και 9 μέσα από τον πίνακα Δ'.1 για το σχήμα Δ'.1 είναι οι πιο κεντρικοί έχοντας την μεγαλύτερη τιμή για την επιλογή του median επιλέγουμε

έναν από του δυο. Στην περίπτωση που το γράφημα έχει βάρος στις ζεύξεις του το shortest path από έναν κόμβο προς έναν άλλων ενδεχόμενος θα αλλάξει. Το Closeness centrality για τον υπολογισμό έχει χρονική πολυπλοκότητα  $O(n^3)$  όπου  $n$  ο αριθμός των κόμβων στο δίκτυο.

Το **Degree Centrality** [34] είναι το ευκολότερο μετρό για την μέτρηση της κεντρικότητας ενός κόμβου. Μετρά την σημαντικότητα του κόμβου ανάλογα με αριθμό των γειτόνων. Αν  $\mathcal{A}$  ο πίνακας γειτνίασης ενός δικτύου  $\mathcal{N}$  με  $n$  κόμβους τότε το degree centrality της πλευράς  $i \in \mathcal{N}$  είναι ο βαθμός  $d_i$  που δίνεται από :

$$d_i = \sum_{j=1}^n \mathcal{A}_{ij}. \quad (\Delta'.2)$$

| Degree Centrality |       |
|-------------------|-------|
| Κόμβος            | Score |
| 1                 | 3     |
| 2                 | 4     |
| 3                 | 5     |
| 4                 | 5     |
| 5                 | 3     |
| 6                 | 2     |
| 7                 | 1     |
| 8                 | 4     |
| 9                 | 5     |
| 10                | 2     |

Πίνακας Δ'.2: Τα αποτελέσματα του Degree Centrality για το σχήμα Δ'.1.

Όπως βλέπουμε στον πίνακα Δ'.2 οι σημαντικότεροι κομβοί χρησιμοποιώντας το Degree Centrality για το δίκτυο είναι ο 3, 4, και ο 9. Για την επιλογή του median επιλέγω ένα από τους τρεις. Το Degree Centrality για τον υπολογισμό του έχει χρονική πολυπλοκότητα  $O(n)$ .

Το **Eigenvector centrality** [35] μετρά την επιρροή του κόμβου στο δίκτυο. Με  $x(i)$  θα συμβολίζουμε την τιμή του eigenvector centrality για τον κόμβο  $n_i$ , τότε

$$x(i) = \frac{1}{\lambda} \sum_{j=1}^n \mathcal{A}_{ij} x(j) \quad (\Delta'.3)$$

Η μεταβλητή  $\lambda$  είναι σταθερή και η εξίσωση  $\Delta'.3$  μπορεί να ξαναγραφεί με την μορφή διανύσματος, όπου  $X = [x(1), x(2), \dots, x(N)]'$  είναι το διάνυσμα των αποτελεσμάτων του Eigenvector centrality όλων των κόμβων.

$$X = \frac{1}{\lambda} \mathcal{A}X$$

$$\lambda X = \mathcal{A}X \quad (\Delta'.4)$$

Το Eigenvector centrality παίρνει το όνομα του από την ιδιοτιμή  $\lambda$  και το  $X$  είναι το αντίστοιχο ιδιοδιάνυσμα του πίνακα  $\mathcal{A}$ . Το eigenvector centrality ενός κόμβου ορίζεται βάση της μεγαλύτερης ιδιοτιμής  $\lambda_1$  του  $\mathcal{A}$ , που ονομάζεται κυρία ιδιοτιμή (principal eigenvalue). Για οποιαδήποτε ιδιοτιμή  $\lambda$  του πίνακα  $\mathcal{A}$  ισχύει  $\lambda_1 > |\lambda|$ . Το ιδιοδιάνυσμα  $X = [x(1), x(2), \dots, x(N)]'$  που αντιστοιχεί στην μεγαλύτερη ιδιοτιμή ονομάζεται κύριο ιδιοδιάνυσμα (principal eigenvector). Τότε το eigenvector centrality που αντιστοιχεί στον κόμβο  $n_i$  είναι το αντίστοιχο στοιχείο  $x_i$  του κύριου ιδιοδιανύσματος  $X$ .

| Eigenvector centrality |        |
|------------------------|--------|
| Κόμβος                 | Score  |
| 1                      | 0.3101 |
| 2                      | 0.3784 |
| 3                      | 0.4317 |
| 4                      | 0.4572 |
| 5                      | 0.2175 |
| 6                      | 0.0566 |
| 7                      | 0.0139 |
| 8                      | 0.3473 |
| 9                      | 0.4005 |
| 10                     | 0.1830 |

Πίνακας  $\Delta'.3$ : Τα αποτελέσματα του Eigenvector centrality για το σχήμα  $\Delta'.1$  που αντιστοιχίσουν στην μεγαλύτερη ιδιοτιμή  $\lambda_1 = 4.0861$ .

Στον πίνακα Δ'.3 βλέπουμε ότι τα αποτελέσματα eigenvector centrality είναι το ιδιοδιάνυσμα που αντιστοιχεί στην ιδιοτιμή  $\lambda_1$  και ο κόμβος 4 έχει την μεγαλύτερη τιμή. Το eigenvector centrality για τον υπολογισμό του έχει χρονική πολυπλοκότητα  $O(n^3)$ .

Υπάρχει και άλλος τρόπος για τον υπολογισμό eigenvector centrality με την χρήση της **Power method** [36] για την εύρεση της μεγαλύτερης ιδιοτιμής. Η Power method προσεγγίζει τις ιδιοτιμές του πίνακα  $\mathcal{A}$  με επαναληπτικό τρόπο. Πολλαπλασιάζοντας τον πίνακα  $\mathcal{A}$  με ένα μη μηδενικό διάνυσμα  $x_0$  επαναληπτικά η μεγαλύτερη ιδιοτιμή  $|\lambda_1|$  θα αυξηθεί εκθετικά.

$$\begin{aligned}
 x_1 &= \mathcal{A}x_0 \\
 x_2 &= \mathcal{A}x_1 = \mathcal{A}(\mathcal{A}x_0) = \mathcal{A}^2x_0 \\
 x_3 &= \mathcal{A}x_2 = \mathcal{A}(\mathcal{A}^2x_0) = \mathcal{A}^3x_0 \\
 &\vdots \\
 x_k &= \mathcal{A}x_{k-1} = \mathcal{A}(\mathcal{A}^{k-1}x_0) = \mathcal{A}^kx_0
 \end{aligned}
 \tag{Δ'.5}$$

Όσο περισσότερες επαναλήψεις πραγματοποιηθούν τόσο καλύτερη προσέγγιση έχουμε για την μεγαλύτερη ιδιοτιμή. Το στοιχείο  $i$  του διανύσματος  $x_k$  αντιπροσωπεύει τον κόμβο  $n_i$ . Η Power method για τον υπολογισμό της έχει χρονική πολυπλοκότητα  $O(n^2)$ .

Το Eigenvector centrality παρόλο που δεν δίνει πάντα την βέλτιστη λύση στο 1-median problem έχει εφαρμογή σε αλγόριθμους Probabilistic Flooding. Οι αλγόριθμοι τύπου Probabilistic Flooding έχουν εφαρμογές σε δίκτυα μεγάλης κλίμακας γιατί μπορούν να μεταδώσουν πληροφορία σε όλο το δίκτυο ελαχιστοποιώντας τις άσκοπες μεταδόσεις σε σχέση με το κλασικό Flooding [37]. Στο Probabilistic Flooding έχειδειχτεί ότι αν ο αλγόριθμος ξεσηκώνει από έναν κόμβο οποίος έχει υψηλό Eigenvector centrality τότε ο αλγόριθμος οδηγεί σε υψηλότερη κάλυψη δικτύου. Οι κόμβοι με υψηλό Eigenvector centrality είναι πιο πιθανό να καλυφθούν [38], [39], [40].

## Κεφάλαιο Ε΄

### Πειράματα και αποτελέσματα

**Σ**τόχος των πειραμάτων είναι να βρούμε σε ποιον κόμβο θα τοποθετήσουμε την υπηρεσία μέσα στο δίκτυο. Για την επιλογή του κόμβου θα χρησιμοποιούμε τα ακόλουθα μετρά Clooseness, το Degree, το Eigenvector centrality και το Eigenvector centrality με την χρήση της Power Method. Το περιβάλλον το οποίο χρησιμοποιήθηκε για την διεξαγωγή των πειραμάτων είναι το MATLAB.

#### Ε΄.1 Κατασκευή δικτύων

Για την διεξαγωγή των πειραμάτων χρειάζεται να κατασκευάσουμε τοπολογίες δικτύων πιο συγκεκριμένα τυχαίους γεωμετρικούς γράφους (random geometric graph). Οι τοπολογίες των δικτύων θα είναι τύχες, όποτε η θέση του κόμβου πρέπει να είναι τυχαία. Για τον λόγο αυτό η μεταβλητή  $x$  αντιπροσωπεύει την θέση του κόμβου πάνω στο άξονα τον  $X$ , η μεταβλητή  $y$  αντιπροσωπεύει την θέση του κόμβου στον άξονα τον  $Y$ . Τα διαστήματα τα οποία έχουμε επιλέξει για τον άξονα τον  $X$  και  $Y$  είναι  $x, y \in [0, 1]$ . Έχουμε κατασκευάσει ένα νοητό επίπεδο όπου οι κόμβοι βρίσκονται μέσα σε αυτό. Για τα πειράματά μας έχουμε επιλέξει να κατασκευάσουμε δίκτυα με 1000 κόμβους. Κάθε κόμβος έχει μια ακτίνα μετάδοσης  $R_c$  (radius circle) η οποία είναι ίδια για όλους τους κόμβους. Αν για παράδειγμα ο κόμβος  $a$  βρίσκεται μέσα στην ακτίνα του κόμβου  $b$  τότε δημιουργούμε μια ζεύξη στο δίκτυο ανάμεσα στον  $a$  και τον  $b$ . Για να δούμε αν ο κόμβος  $a$  βρίσκεται στην ακτίνα του κόμβου  $b$  και το αντίστροφο θα πρέπει να μετρήσουμε την απόστασή τους. Αφού οι κόμβοι του δικτύου αναπαριστώνται σε

ένα επίπεδο η μεταξύ τους απόσταση δίνεται από την **ευκλείδεια μετρική**.

Η ευκλείδεια μετρική είναι μία συνάρτηση  $d : \mathbb{R}^n \times \mathbb{R}^n \rightarrow \mathbb{R}$  όπου αντιστοιχεί σε δυο διανύσματα  $X, Y$  του  $n$ -διάστατο χώρο  $\mathbb{R}^n$  με  $X = [x_1, \dots, x_n]$  και  $Y = [y_1, \dots, y_n]$  με τον αριθμό

$$d(X, Y) = \sqrt{(y_1 - x_1)^2 + (y_2 - x_2)^2 + \dots + (y_n - x_n)^2} = \sqrt{\sum_{i=1}^n (y_i - x_i)^2} \quad (E'.1)$$

Στην περίπτωση του επιπέδου όπου ορίζεται από δυο διαστάσεις η εξίσωση E'.1 γράφεται ως

$$d(X, Y) = \sqrt{(y_1 - y_2)^2 + (x_1 - x_2)^2} \quad (E'.2)$$

Την πληροφορία της τοπολογίας του δικτύου μας την δίνει ο πίνακας γειτνίασης. Για να τον κατασκευάσουμε προγραμματιστικά χρησιμοποιούμε τις συντεταγμένες από τα διανύσματα  $X = [x_1, \dots, x_N]$  και  $Y = [y_1, \dots, y_N]$  με  $N$  ο αριθμός των κόμβων. Ο κόμβος  $n_i$  αναπαριστάτε από της συντεταγμένες  $(x_i, y_i)$  με  $i, j \in N$ . Αν η ευκλείδεια απόσταση του κόμβου  $n_i$  με τον κόμβο  $n_j$  είναι μικρότερη ή ίση με το  $Rc$  τότε το στοιχείο του πίνακα γειτνίασης που βρίσκεται στην γραμμή  $i$  και στην στήλη  $j$  ορίζετε η τιμή του ως 1 αλλιώς αν είναι μεγαλύτερη ορίζεται με 0.

Επειδή θέλουμε να δούμε την συμπεριφορά των μέτρων στο δίκτυο ανάλογα με τον αριθμό των ζεύξεων, δεν αρκεί να έχουμε ένα δίκτυο με ένα συγκεκριμένο  $Rc$ . Για αυτό το λόγο δημιουργούμε νέα δίκτυα με διαφορετικό  $Rc$ . Επειδή τα δίκτυα τα οποία δημιουργούμε είναι τυχαία δεν θα μπορούμε να κατανοήσουμε την την αλλαγή που έχει γίνει γιατί όλοι οι κόμβοι έχουν μετακινηθεί σε νέες θέσεις. Η λύση σε αυτό το πρόβλημα είναι να κρατήσουμε τις θέσεις των κόμβων και κάθε φορά που θέλουμε να δημιουργούμε ένα νέο δίκτυο αλαλάζουμε μόνο το  $Rc$ . Τα δίκτυα με  $rc < 0,1$  καταλήγουν να είναι μη-συνεκτικά δίκτυα ενώ τα δίκτυα με  $rc > 0,3$  γίνονται πολύ πυκνά διότι υπάρχει τεράστιος αριθμός ζεύξεων.

Για κάθε κόμβο του δικτύου έχει ανατεθεί ένας τυχαίος αριθμός  $L_i$  με  $0 \leq L_i \leq 1$  ο οποίος ποσοτικοποιήσει το φορτίο (Load) τις κινήσεις του εκάστοτε κόμβου. Με βάση τα προηγούμενα καταλήξαμε να κατασκευάζουμε δίκτυα των 1000 κόμβων σε ένα επίπεδο με διαστάσεις  $1 \times 1$  και  $Rc$  από 0.1 έως 0.3 με βήμα αύξησης για το νέο δίκτυο 0.05 και τυχαία βάρη στους κόμβους τα οποία παραμένουν σταθερά για όλα τα δίκτυα.

Ο αντίστοιχος κώδικας για την δημιουργία του δικτύου βρίσκεται στο **Παράρτημα I**.

## Ε'.2 Ορισμός κόστους

Τα δίκτυα τα οποία έχουμε κατασκευάσει είναι μη-κατευθυνόμενα και δεν είναι έμβαρα άλλα έχουν φορτίο στους κορύμβους τους. Για να μεταδοθεί μια πληροφορία από έναν κόμβο σε κάποιον κόμβο μέσα στο δίκτυο υπάρχει ένα κόστος. Έστω ότι έχουμε τον κόμβο  $s$  που θέλει να στείλει μια πληροφορία στον κόμβο  $u$ . Αφού το δίκτυο μας δεν είναι έμβαρo η απόσταση από τον  $s$  στον  $u$  ισούται με το shortest path από τον  $d(u, s)$ . Το κόστος ισούται με το άθροισμα την απόστασης  $d(u, s)$  πολλαπλασιαζόμενο κάθε φορά από το φορτίο του κόμβου που περνάει η πληροφορία. Τότε

$$d(u, s)L(u) \quad (\text{Ε'.3})$$

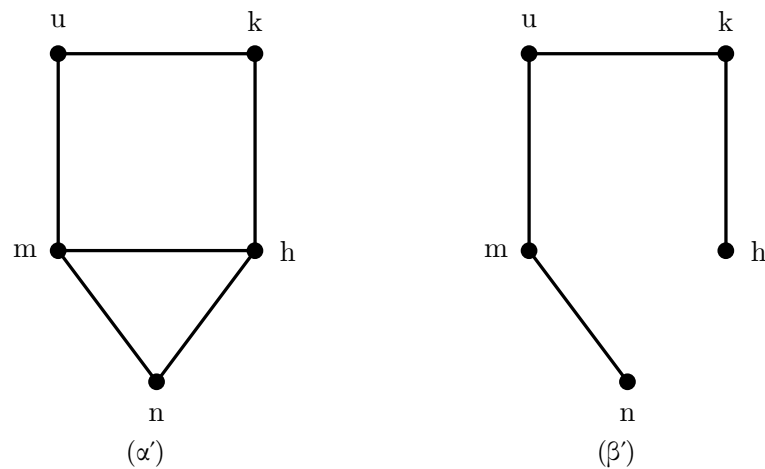
Με  $L(u)$  συμβολίζουμε το κόστος κάθε φορά του κόμβου  $u$ . Για να αξιολογήσουμε έναν κόμβο χρειάζεται να ξέρουμε το συνολικό κόστος που έχει. Το συνολικό κόστος ενός κόμβου  $s$  ορίζεται και υπολογίζεται ως:

$$C(s) = \sum_{u \in V} d(u, s)L(u) \quad (\text{Ε'.4})$$

Όπου  $V$  το σύνολο όλων των κόμβων του δικτύου. Για να βρούμε το shortest path ανάμεσα σε δύο κόμβους χρησιμοποιούμε τον αλγόριθμο *dijkstra*. Η προηγούμενη προσέγγιση είναι παρόμοια με Closeness centrality υπολογίζοντας και κόστος του φορτίου του κάθε κόμβου. Ο κόμβος στο δίκτυο ο οποίος έχει το μικρότερο κόστος είναι η βέλτιστη λύση στο 1-median problem. Ο αντίστοιχος κώδικας βρίσκεται στο **Παράρτημα II**.

## Ε'.3 Δημιουργία του shortest path tree

Δεδομένου ενός συνεκτικού, μη-κατευθυνόμενου γραφήματος  $G$ , το shortest path tree [41] με ρίζα του δέντρου (tree) τον κόμβο  $v$  είναι ένα spanning tree  $T$  του  $G$ , τέτοιο ώστε η απόσταση από την ρίζα  $v$  προς οποιοδήποτε άλλων κόμβο  $u$  του  $T$  είναι το shortest path από τον  $v$  στον  $u$  στο γράφημα  $G$ .



Σχήμα Ε'.1: Το Ε'.1α' ένα μη-κατευθυνόμενο γραφήμα και το Ε'.1β' το shortest path tree με ρίζα του δέντρου τον κόμβο  $u$ .

Για να κατασκευάσουμε το shortest path tree θα χρησιμοποιήσουμε τον πίνακα γειτνίασης και θα εφαρμόσουμε αλγόριθμο *dijkstra* τον κόμβο  $u$  προς όλους για να βρούμε όλα τα shortest paths του κόμβου  $u$ . Η υλοποίηση του κώδικα βρίσκεται στο Παράρτημα ΙΙΙ.

## Ε'.4 Υπολογισμός των μέτρων

Έχουμε χρησιμοποιήσει την βιβλιοθήκη Matlab Network Analysis η οποία προσφέρει αρκετές υπό-ρουτίνες έτοιμες. Για τον υπολογισμό του Degree Centrality χρησιμοποιήσαμε την συνάρτηση `degrees` με είσοδο τον πίνακα γειτνίασης. Η συνάρτηση αυτή επιστέφει ένα μονοδιάστατο πίνακα όπου η  $i$ -θέση του πίνακα αντιπροσωπεύει το κόμβο  $i$  και το στοιχείο του πίνακα τον αριθμό των γειτόνων. Για να βρούμε ποίος κόμβος έχει τους περισσότερους γείτονες αρκεί να βρούμε το μεγαλύτερο στοιχείο του πίνακα και η θέση του θα μας υποδείξει τα στοιχεία του κόμβου. Υπάρχει περίπτωση περισσότερη από έναν κόμβο να έχουν την μέγιστη τιμή του Degree Centrality. Η υλοποίηση του αντιστοίχου κώδικα βρίσκεται στο Παράρτημα ΙV.

Αντίστοιχα για τον υπολογισμό του Closeness Centrality για κάθε κόμβο που έχουμε στο δίκτυο χρησιμοποιούμε την συνάρτηση `closeness`. Δίνουμε ως είσοδο τον πίνακα γειτνίασης του δικτύου και μας επιστέφει έναν μονοδιάστατο πίνακα με τα scores όλων των κόμβων, η  $i$ -θέση του πίνακα με το υψηλότερο score είναι η βέλτιστη

λύση του 1-median problem. Στην περίπτωση που έχουμε στο δίκτυο φορτίο στους κόμβους θα χρησιμοποιήσουμε την συνάρτηση `closeness_with_weight_of_vertex` στη θέση της `closeness` και θα ψάξουμε να βρούμε το μικρότερο κόστος. Η υλοποίηση του αντιστοίχου κώδικα βρίσχετε στο **Παράρτημα V**.

Για τον υπολογισμό του Eigenvector Centrality αρκεί να βρούμε την μεγαλύτερη ιδιοτιμή  $\lambda_1$  του πίνακα γειτνίασης μαζί με το αντιστοιχώ ιδιοδιάνυσμα. Η  $i$ -θέση του διανύσματος αναπαριστά τον κόμβο  $i$  και το περιεχόμενο του την επιρροή που έχει στο δίκτυο. Για να βρούμε ποίος κόμβος έχει την μεγαλύτερη επιρροή στο δίκτυο αρκεί να ψάξουμε σε πια θέση βρίσκεται η μεγαλύτερη τιμή του. Η αντίστοιχος κώδικας βρίσκεται στο **Παράρτημα VI**.

Το Eigenvector Centrality με την χρήση της **Power Method** υπολογίζεται πολλαπλασιάζοντας ένα μοναδιαίο μονοδιάστατο πίνακα μήκους  $n$  με τον πίνακα  $A^k$ . Το αποτέλεσμα του πολλαπλασιασμού δίνει ένα διάνυσμα όπου η  $i$ -θέση αντιπροσωπεύει τον κόμβο  $i$  και το περιεχόμενο του την επιρροή που έχει ο κόμβος. Υπολογιστικά προκύπτει το πρόβλημα ότι το διάνυσμα  $x_k$  μεγαλώνει πολύ γρήγορα το πρόβλημα αυτό ονομάζεται Overflow και για να το αντιμετωπίσουμε κάνουμε εξομάλιση το διάνυσμα (**vector normalization**) μετά από κάθε πολλαπλασιασμό με τον πίνακα  $A$ . Η συγκεκριμένη υλοποίηση του κωδικά βρίσκεται στο **Παράρτημα VII**.

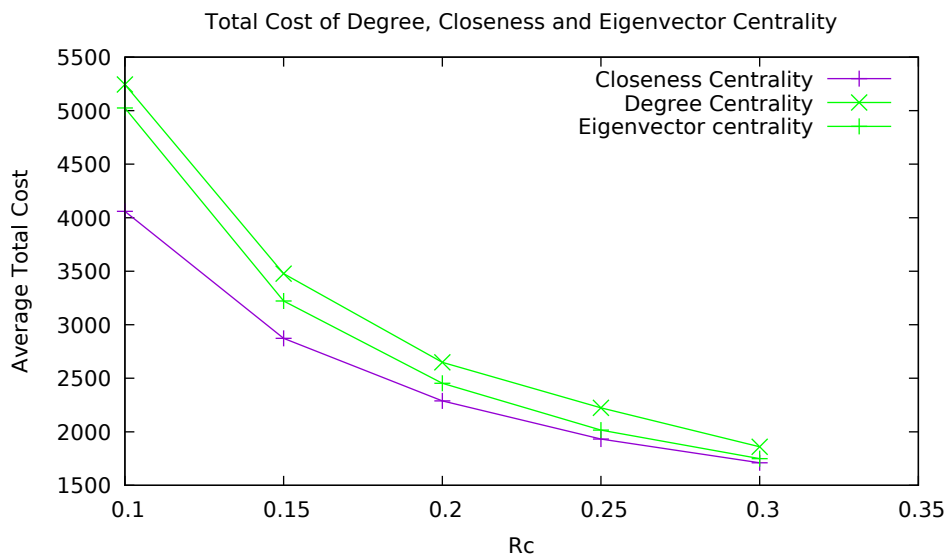
## Ε'.5 Πειράματα

Η φιλοσοφία των πειραμάτων είναι να χρησιμοποιήσουμε κάθε φορά μια συγκεκριμένη τοπολογία και να μεταβάλλεται μόνο το `rc`. Σε κάθε πείραμα θα μετράμε τα μετρά `Closeness Centrality`, `Degree Centrality`, `Eigenvector Centrality` και `Eigenvector centrality` με την χρήση τις `Power Method` για το εκάστοτε δίκτυο. Όλα τα προηγούμενα μέτρα προτείνουν ένα κόμβο στο δίκτυο με την υψηλότερη ή χαμηλότερη τιμή. Οι κόμβοι οι οποίοι προτείνονται από τα μέτρα πρέπει να αξιολογηθούν. Για την αξιολόγηση θα κατασκευάσουμε τα `shortest path tree` με ριζά του δέντρου τον εκάστοτε κόμβο. Για κάθε `shortest path tree` υπολογίσουμε το συνολικό του κόστος (Παράρτημα III). Επίσης ένα άλλο σημαντικό μέτρο για την αξιολόγηση είναι ο χρόνος υπολογισμού του κάθε μέτρου.

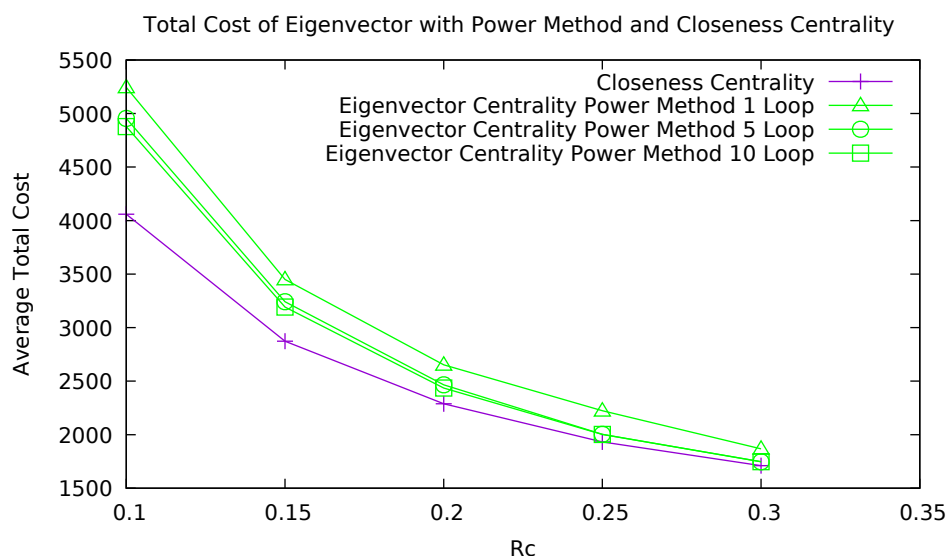
Έχουμε κατασκευάσει 20 διαφορετικά δίκτυα με 1000 κόμβους. Για κάθε δίκτυο υπάρχουν τοπολογίες όπου οι κόμβοι παραμένουν σταθερή και αλλάζει το `rc`. Τα παρακάτω διαγράμματα έχουν προκύψει από τον υπολογισμό του μέσου όρου των α-

ποτελεσμάτων των 20 διαφορετικών δικτύων. Όλα τα διαγράμματα θα συγκρίνουν το εκάστοτε μέτρο με το Closeness Centrality γιατί αποτελεί την βέλτιστη λύση στο 1-median problem.

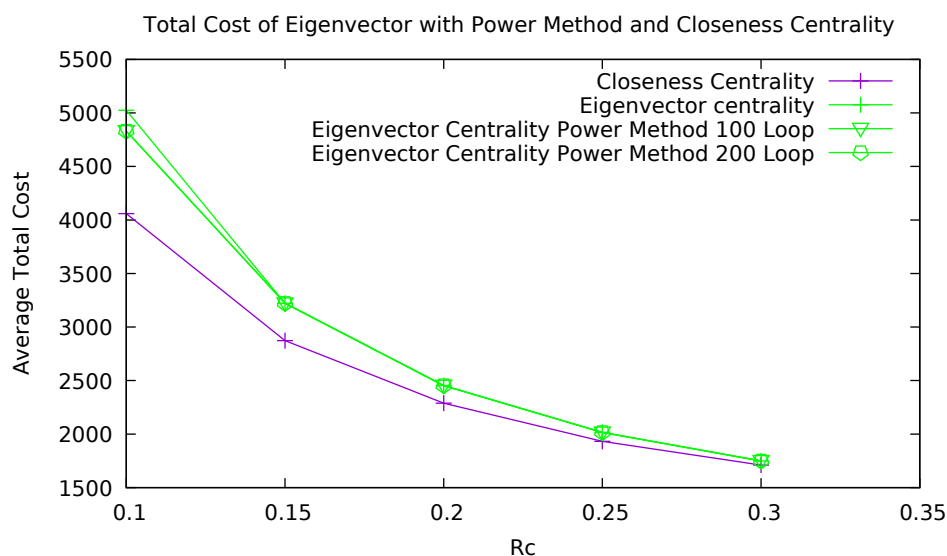
Όπως βλέπουμε στα σχήματα Ε'.2, Ε'.3, Ε'.4, Ε'.5 παρουσιάζονται τα διαγράμματα κόστους. Το χαμηλότερο κόστος το έχει το Closeness Centrality το οποίο επιβεβαιώνετε και από την θεωρία, ενώ το υψηλότερο κόστος το έχει το Degree Centrality. Όσο αυξάνεται το  $rc$  στα δίκτυα το κόστος του κάθε κόμβου μειώνεται διότι με την αύξηση του δημιουργούνται νέες ζεύξεις και νέα μονοπάτια μέσα στο δίκτυο. Είναι προφανές ότι ο κόμβος ο οποίος έχει επιλεγεί από ένα μετρώ σε ένα δίκτυο με συγκεκριμένο  $rc$ . Δεν είναι απαραίτητο να επιλεγεί ξανά επειδή αλλαρίζοντας το  $rc$  αλλάζει και η τοπολογία του δικτύου. Στο διάγραμμα Ε'.2 βλέπουμε ότι το κόστος του Eigenvector Centrality βρίσκεται ανάμεσα από το Closeness και το Degree Centrality. Καθώς το  $rc$  αυξάνεται παρατήσουμε ότι το κόστος του Eigenvector και του Degree πλησιάζει το κόστος του Closeness.



Σχήμα Ε'.2: Συνολικό κόστος του Closeness, του Degree και του Eigenvector Centrality



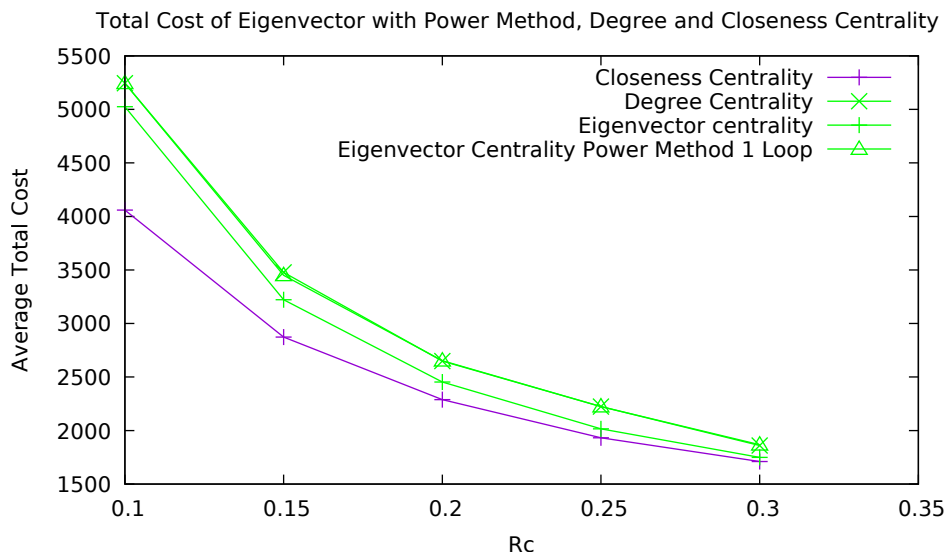
Σχήμα Ε'.3: Συνολικό κόστος του Closeness - Eigenvector Centrality με την χρήση της Power Method με 1, 5 και 10 Loop



Σχήμα Ε'.4: Συνολικό κόστος του Closeness και του Eigenvector Centrality με την χρήση της Power Method με 100 και 200 Loop

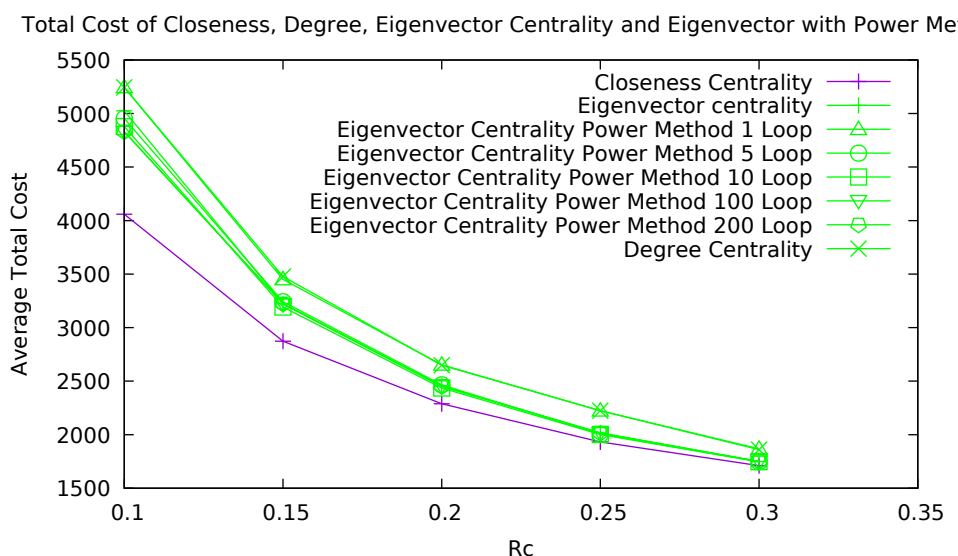
Στα διαγράμματα Ε'.3 και Ε'.4 βλέπουμε τα αποτελέσματα του Egenvector Central-ity με την χρήση της Power Method. Στο σχήμα Ε'.5 παρατηρούμε ότι Egenvector

Centrality Power Method 1 Loop είναι το Degree Centrality το οποίο έχει υπολογιστή πολλαπλασιάζοντας τον πίνακα γειτνιάσης με ένα μη μηδενικό διάνυσμα που περιέχει μόνο τον αριθμό 1 (Παράρτημα VII). Μετά από έναν συγκεκριμένο αριθμό επαναλήψεων τα αποτελέσματα Power Method θα ταυτιστούν με τα αποτελέσματα του Eigenvector Centrality διάγραμμα Ε'.3 και Ε'.4 .



Σχήμα Ε'.5: Ταύτιση των αποτελεσμάτων της Power Method 1 Loop με τα αποτελέσματα του Degree Centrality

Από της μετρήσεις των πειραμάτων προκύπτει ο πίνακας Ε'.1 που δείχνει την ποσοστιαία διάφορα της αυξήσεις του κόστους για κάθε μετρό σε σχέση με το Closeness Centrality. Τα αποτελέσματα της Power Method 1 Loop παραλείπονται γιατί ταυτίζονται με τα αποτελέσματα του Degree Centrality.



Σχήμα Ε'.6: Συνολικό διάγραμμα κόστους με όλα τα μέτρα

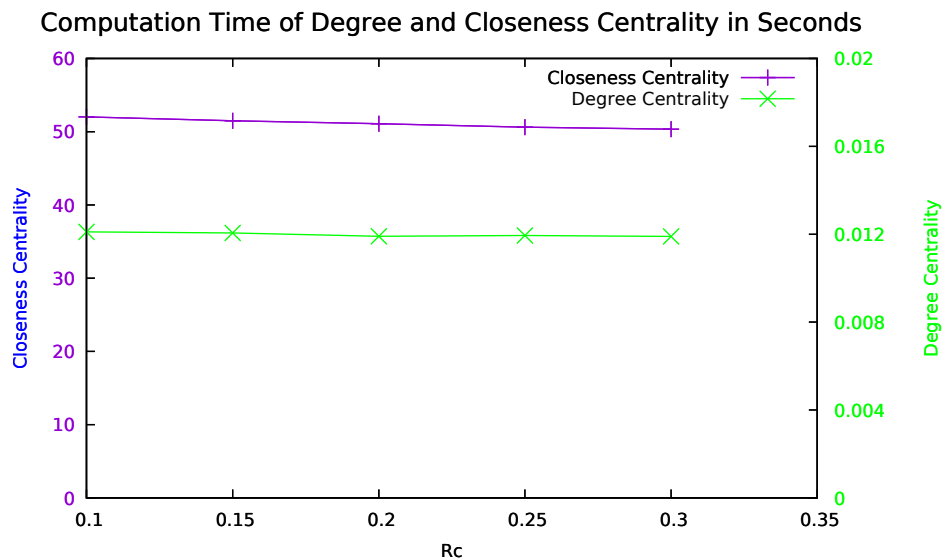
| Ποσοστιαία διάφορα αύξησης κόστους |        |             |        |         |          |          |
|------------------------------------|--------|-------------|--------|---------|----------|----------|
| Rc                                 | Degree | Eigenvector | Loop 5 | Loop 10 | Loop 100 | Loop 200 |
| 0.1                                | 29.2%  | 23.8%       | 22%    | 20.1%   | 19%      | 19%      |
| 0.15                               | 21%    | 12.1%       | 12.8%  | 11.1%   | 12.1%    | 12.1%    |
| 0.2                                | 15.7%  | 7.1%        | 7.3%   | 6.4%    | 7.1%     | 7.1%     |
| 0.25                               | 15.1%  | 4.3%        | 3.6%   | 3.6%    | 4.3%     | 4.3%     |
| 0.3                                | 9.7%   | 2.2%        | 2.1%   | 2.2%    | 2.2%     | 2.2%     |

Πίνακας Ε'.1: Ποσοστιαία αύξηση κόστους σε σχέση με το Closeness Centrality .

Ο πίνακας Ε'.1 δείχνει με ποσοστά πόσο της εκατό είναι καλύτερο το Closeness Centrality σε σχέση με τα υπολοιπα μέτρα. Αξίζει να σημειώσουμε ότι συνολικά μετά το Closeness ακολουθεί το Eigenvector μετά η Power Method και τέλος το Degree Centrality. Παρατηρούμε επίσης στο πίνακα Ε'.1 ότι η Power Method με 10 επαναλήψεις έχει πολύ καλύτερα αποτέλεσμα από άλλα μέτρα. Πιστεύουμε ότι το αποτέλεσμα αυτό οφείλετε στο γεγονός ότι δεν έχουν γίνει αρκετές επαναλήψεις για ταυτιστεί το ιδιοδιάνυσμα της Power Method με το Eigenvector Centrality. Όποτε ο κόμβος οποίος προτείνεται είναι τυχαίος βέβαια πρέπει να γίνει μελλοντική έρευνα για την συμπεριφορά αυτή ωστόσο δεν μπορούμε να αγνοήσουμε ότι δίνει τα καλύτερα

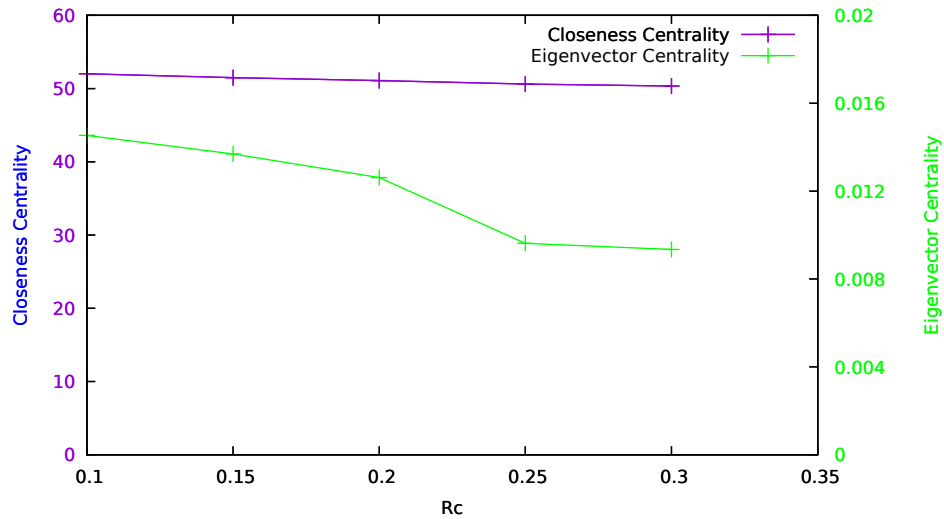
αποτελέσματα μετά το Closeness Centrality.

Οι χρόνοι υπολογισμού του κάθε μέτρου είναι αναλογεί της χρονικής πολυπλοκότητας του. Στα σχήματα Ε'.7, Ε'.8, Ε'.9 και Ε'.10 δείχνουν του χρόνους υπολογισμού το κάθε μέτρου.

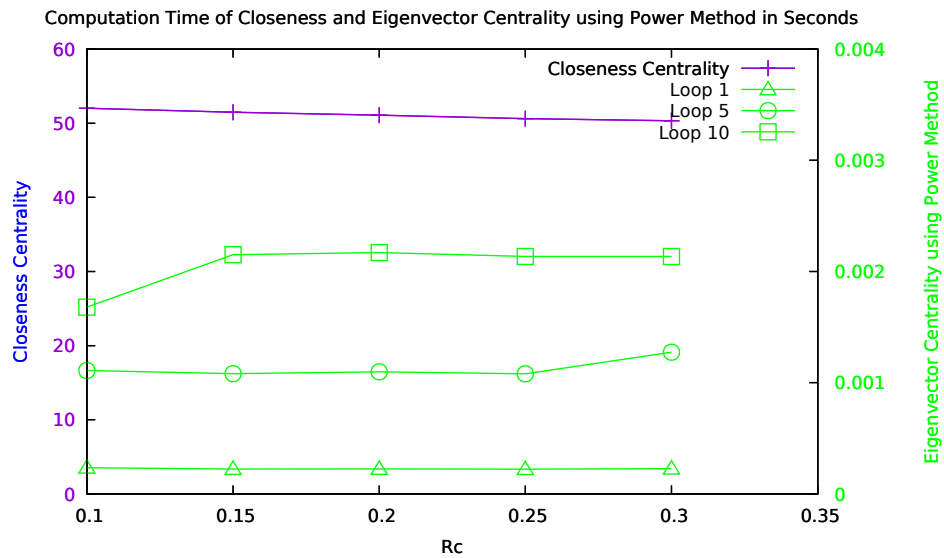


Σχήμα Ε'.7: Οι χρόνοι υπολογισμού του Closeness Centrality και του Degree Centrality

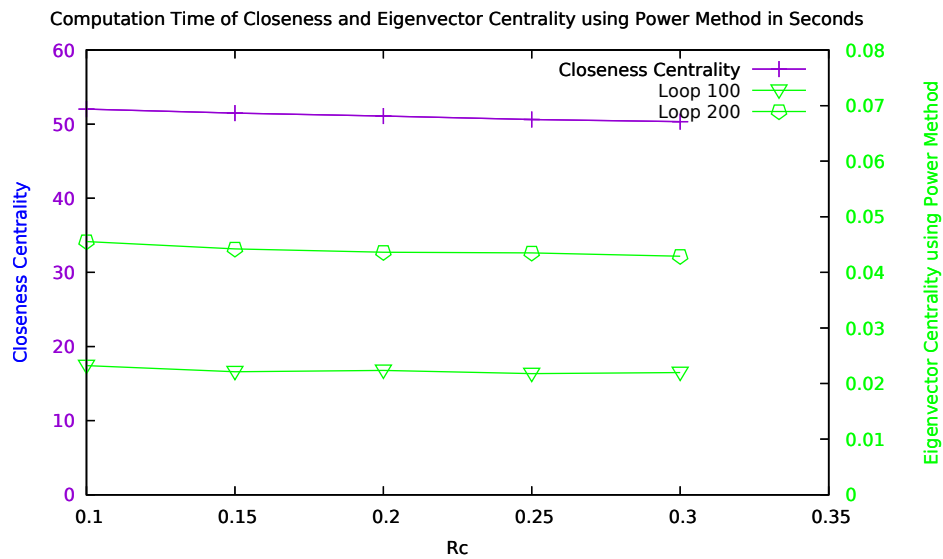
Computation Time of Eigenvector and Closeness Centrality in Seconds



Σχήμα Ε'.8: Οι χρόνοι υπολογισμού του Closeness Centrality και του Eigenvector Centrality



Σχήμα Ε'.9: Οι χρόνοι υπολογισμού του Closeness Centrality και του Eigenvector Centrality με την χρήση της Power Method με 1, 5 και 10 Loop



Σχήμα Ε'.10: Οι χρόνοι υπολογισμού του Closeness Centrality και του Eigenvector Centrality με την χρήση της Power Method με 100 και 200 Loop

Όπως παρατηρούμε από τα σχήματα Ε'.7, Ε'.8, Ε'.9 και Ε'.10 υπάρχει τεράστια διάφορα στον χρόνο υπολογισμού του Closeness Centrality σε σχέση με τα υπόλοιπα μέτρα. Ο επόμενος πίνακας δείχνει πόσες φορές είναι πιο εύκολο στον υπολογισμό το κάθε μέτρο σε σχέση με το Closeness Centrality.

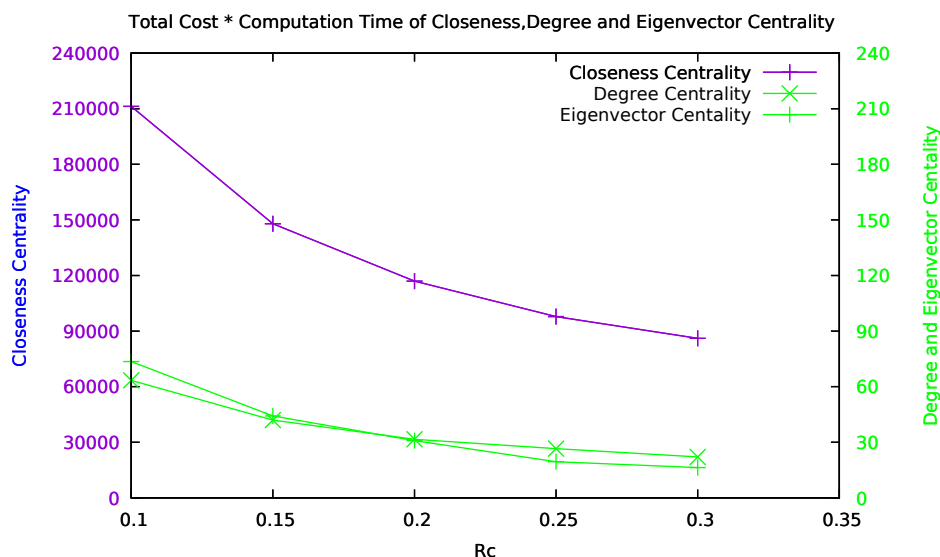
| Αποτελέσματα Υπολογίσιμων |        |             |        |        |         |          |          |
|---------------------------|--------|-------------|--------|--------|---------|----------|----------|
| Rc                        | Degree | Eigenvector | Loop 1 | Loop 5 | Loop 10 | Loop 100 | Loop 200 |
| 0.1                       | 4296   | 3580        | 220731 | 46890  | 30978   | 2238     | 1141     |
| 0.15                      | 4229   | 3760        | 230409 | 47613  | 23930   | 2326     | 1163     |
| 0.2                       | 4289   | 4049        | 228307 | 46568  | 23544   | 2283     | 1170     |
| 0.25                      | 4238   | 5258        | 227485 | 46899  | 23706   | 2322     | 1162     |
| 0.3                       | 4229   | 5386        | 222109 | 39390  | 23572   | 2289     | 1172     |

Πίνακας Ε'.2: Αποτελέσματα μετρήσεων του χρόνου των μέτρων σε σχέση με το Closeness Centrality.

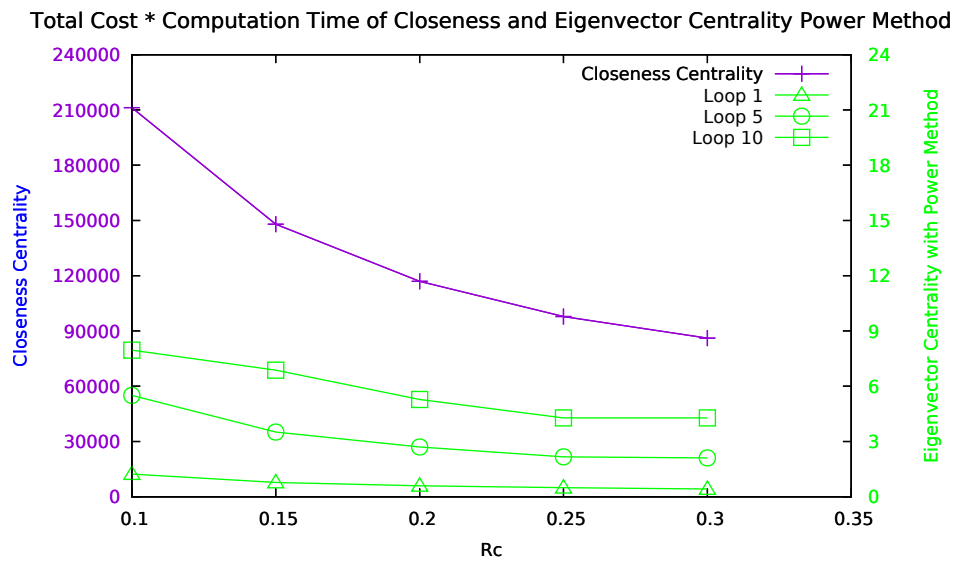
Ο πίνακας Ε'.2 δείχνει ότι για τον χρόνο διάστημα που χρειάζεται να υπολογιστεί το Closeness Centrality στο διάστημα αυτό μπορεί να υπολογιστεί το αντιστοιχώ μέτρο αντίστοιχες φορές. Όσο πιο μεγάλος είναι ο αριθμός αυτός τόσο πιο λίγο χρόνο

απαιτείται για να υπολογισμός του μέτρου. Παρατηρούμε από το σχήμα Ε'.5 ότι το Degree και το Eigenvector Centrality με την χρήση της Power Method 1 Loop έχουν το ίδιο συνολικό κόστος άλλα έχουνε μια διάφορα μεγέθους. Αυτή η διάφορα οφείλετε στο διαφορετικό τρόπο υπολογισμού. Δηλαδή ο πολλαπλασιασμός ενός πίνακα με ένα διάνυσμα είναι πιο φτηνός από ότι το άθροισμα της κάθε γραμμής του πίνακα.

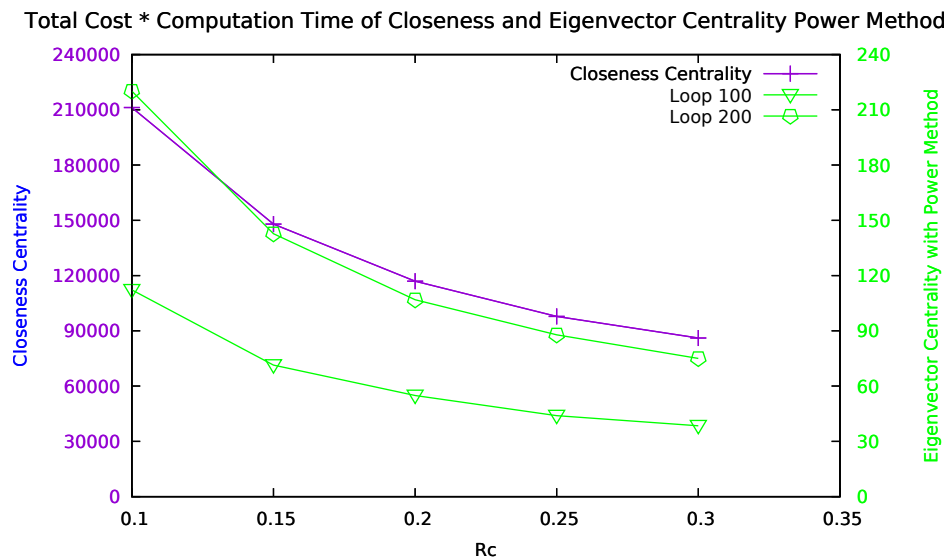
Είδαμε στα σχήματα Ε'.2, Ε'.3, Ε'.4, Ε'.5 το συνολικό κόστος του κάθε μέτρου και στα σχήματα Ε'.7, Ε'.8, Ε'.9, Ε'.10 τους χρόνους υπολογισμού το κάθε μέτρου. Προφανώς δεν αρκεί μόνο κοιτάζουμε πιο μέτρο έχει το χαμηλότερο συνολικό ή το χαμηλότερο χρόνο υπολογισμού. Για παράδειγμα αν ένα μετρό έχει χαμηλό συνολικό κόστος και τεράστιο χρόνο υπολογισμού ή το αντίθετο δεν θα ήταν η αποδοτικότερη επιλογή για να την χρησιμοποιήσουμε στην επιλογή κόμβου για την τοποθέτηση της υπηρεσίας στο δίκτυο. Για την επιλογή αυτή θα πρέπει να επιλέξουμε το μέτρο που έχει χαμηλό συνολικό κόστος και χαμηλό χρόνο υπολογισμού τα σχήματα Ε'.11, Ε'.12, Ε'.13, Ε'.14 δείχνουν τον πολλαπλασιασμό ανάμεσα στο συνολικό κόστος και στον χρόνο υπολογισμού.



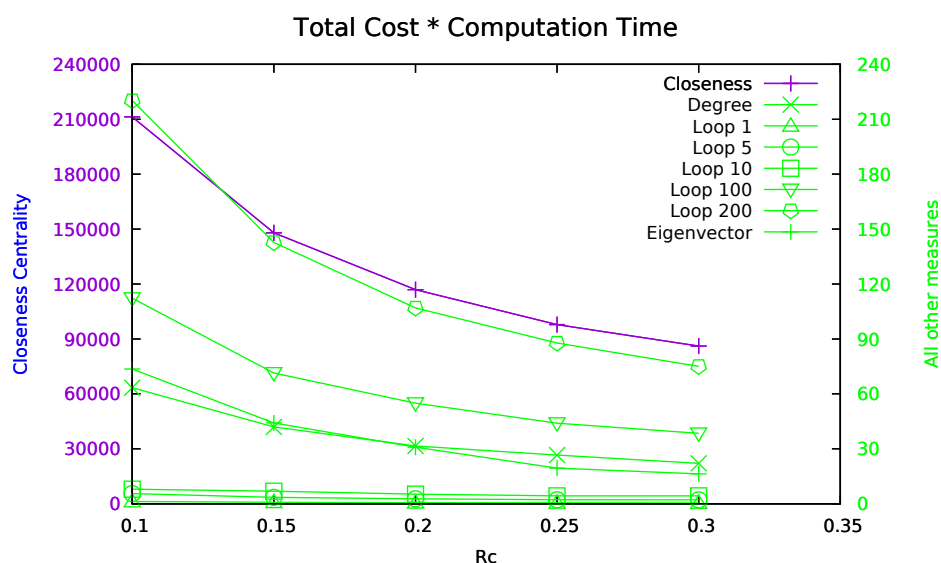
Σχήμα Ε'.11: Αποτελέσματα πολλαπλασιασμού του χρόνου υπολογισμού με το συνολικό κόστος για το Closeness, Degree και το Eigenvector Centrality



Σχήμα Ε'.12: Αποτελέσματα πολλαπλασιασμού του χρόνου υπολογισμού με το συνολικό κόστος για το Closeness και για το Eigenvector Centrality με την χρήση της Power Method για 1, 5 και 10 Loop



Σχήμα Ε'.13: Αποτελέσματα πολλαπλασιασμού του χρόνου υπολογισμού με το συνολικό κόστος για το Closeness και για το Eigenvector Centrality με την χρήση της Power Method για 100 και 200 Loop



Σχήμα Ε'.14: Αποτελέσματα πολλαπλασιασμού του χρόνου υπολογισμού με το συνολικό κόστος για όλα τα μέτρα

Από τα σχήματα Ε'.11, Ε'.12, Ε'.13, Ε'.14 παρατηρούμε ότι το Closeness Centrality είναι πολύ δαπανηρό και υπάρχει διάφορα τάξης σε σχέση με τα υπόλοιπα μέτρα.

Στο σχήμα Ε'.14 παρατηρούμε ότι η το Eigenvector Centrality με την χρήση Power Method με 100 – 200 Loop έχουν υψηλότερο κόστος από το Eigenvector και το Degree Centrality. Επίσης στο σχήμα Ε'.11 βλέπουμε ότι το Eigenvector Centrality έχει ελαφρώς καλύτερα αποτελέσματα από το Degree Centrality.

Ενδιαφέρον έχουν και τα αποτελέσματα από το σχήμα Ε'.12. Όπως έχουμε ξανά πει το Eigenvector Centrality με την χρήση Power Method με 1 Loop είναι το Degree Centrality υπολογισμένο με άλλο τρόπο. Τα αποτελέσματα με της Power Method με 5 και 10 Loop δίνουν πολύ καλύτερα αποτελέσματα από το Eigenvector Centrality γιατί έχουν αρκετά κοντινές τιμές στο συνολικό κόστος (σχήμα Ε'.3) και πολύ καλύτερο χρόνο υπολογισμού (σχήματα Ε'.8, Ε'.9) .

Από τα πειράματα που έγιναν το μόνο μετρο που επηρεάζεται από το βάρος των κόμβων στο δίκτυο είναι το Closeness Centrality. Δεν έχει σημασία στον χρόνο υπολογισμού αν οι κόμβοι έχουν ίδιο το βάρος η διαφορετικό. Το μόνο που αλλάζει είναι ότι το δίκτυο το οποίο έχει διαφορετικό βάρος στους κόμβους έχει υψηλότερο συνολικό κόστος σε σχέση με το ίδιο δίκτυο ίδιο βάρος σε όλους στους κόμβους του. Για αυτό σε στα πειράματα τα οποία έγιναν το κόστος για όλους τους κόμβους είναι

έναν σταθερό αριθμό π.χ 1 .



## Κεφάλαιο Γ΄

# Συμπεράσματα και Μελλοντική Εργασία

Το κύριο αντικείμενο αυτής της πτυχιακής εργασίας ήταν η μελέτη της τοποθέτησης υπηρεσιών σε δικτυακά περιβάλλοντα με την χρήση αλγεβρικής θεωρίας γράφων. Στην Γραμμική Άλγεβρα ορίσαμε την έννοια του πίνακα, πως γίνεται η πρόσθεση στου πίνακες, τον βαθμωτό πολλαπλασιασμό, τον πολλαπλασιασμό στους πίνακες, τις ορίζουσες, τις ιδιοτιμές και τα ιδιοδιανύσματα καθώς και τις ιδιότητες τους. Στην Θεωρία Γραφημάτων είδαμε τις βασικές έννοιες καθώς και τους τύπους των γραφημάτων. Ορίσαμε τον πίνακα γειτνίασης μαζί με τις ιδιότητες του και είδαμε πως συνδέεται με γραφήματα. Στο κεφάλαιο Δ΄ ορίσαμε το πρόβλημα του 1—median, πως βρίσκουμε την λύση του χρησιμοποιώντας το Closeness, το Degree και το Eigenvector Centrality. Τέλος στο κεφάλαιο Ε΄ παρατηρήσαμε την συμπεριφορά τους καθώς και πως υλοποιούμε τα πειράματα.

### Γ΄.1 Σύνοψη και Συμπεράσματα

Καταλήξαμε στο συμπέρασμα ότι όταν έχουμε ένα δίκτυο το οποίο αντιπροσωπεύει έναν γεωμετρικό τυχαίο γράφο και θέλουμε να τοποθετήσουμε μια νέα υπηρεσία η βέλτιστη λύση δίνεται από το Closeness Centrality. Διότι βρίσκει τον κόμβο που ελαχιστοποιεί τις αποστάσεις που έχει ως αποτέλεσμα ο κόμβος αυτός να έχει το μικρότερο συνολικό κόστος μέσα στο δίκτυο. Η λύση αυτή είναι η βέλτιστη αναφορικά με το συνολικό κόστος του κόμβου. Άλλα έχει το μειονέκτημα ότι όσο αυξάνεται ο αριθμός των

κόμβων αυξάνεται ραγδαία ο χρόνος υπολογισμού καθιστώντας τη μη αποτελεσματική λύση για μεγάλα δίκτυα.

Το Degree Centrality είναι το ευκολότερο μέτρο στον υπολογισμό του και είναι αρκετά γρήγορο άλλα έχει την χειρότερη προσέγγιση για την επιλογή του κόμβου από τα μέτρα τα οποία συγκρίνουμε. Που έχει ως αποτέλεσμα ο κόμβος αυτός να έχει το μεγαλύτερο συνολικό κόστος.

Στον ίδιο χρόνο που μπορούμε υπολογίζουμε το Degree έχουμε την δυνατότητα να υπολογίσουμε το Eigenvector Centrality. Το οποίο δίνει πολύ καλύτερη προσέγγιση για την επιλογή του κόμβου, με συνέπεια το συνολικό κόστος να είναι χαμηλότερο σε σχέση με το Degree.

Ενδιαφέρων έχουν τα αποτελέσματα του Eigenvector Centrality με την χρήση της Power Method. Τα αποτελέσματα της με 1-Loop ταυτίζονται με το συνολικό κόστος του Degree Centrality άλλα έχει αρκετά χαμηλότερο χρόνο υπολογισμού. Επίσης με 5 Loop τα αποτελέσματα προσεγγίζουν το συνολικό κόστος του Eigenvector Centrality με μικρότερο τον χρόνο υπολογισμού.

Ωστόσο τα αποτελέσματα συνολικού κόστους των 100 και 200 Loop ταυτίζονται με τα αποτελέσματα του Eigenvector Centrality άλλα έχουνε μεγαλύτερο χρόνο υπολογισμού από αυτό.

Τέλος τα αποτελέσματα των 10 Loop έχουν την καλύτερη προσέγγιση για την επιλογή της τοποθέτησης της υπηρεσίας στο δίκτυο με το χαμηλότερο συνολικό κόστος μετά το Closeness Centrality. Έχοντας έναν πολύ χαμηλό χρόνο υπολογισμού καθιστώντας το μια πολύ καλή επιλογή για μεγάλα δικτυακά περιβάλλοντα τα οποία μεταβάλλονται πολύ γρήγορα. Να σημειωθεί ότι όσο πιο πυκνό είναι δίκτυο (Δηλ. αυξάνεται ο αριθμός των ζεύξεων) όλα τα μέτρα τείνουν να πλησιάσουν το συνολικό κόστος το Closeness Centrality.

## Γ'.2 Μελλοντική Εργασία

Οι μελλοντικές κατευθύνσεις εργασίας σχετίζονται με την μελέτη τις συμπεριφοράς των μέτρων σε διαφορετικά γραφήματα όπως η μοντελοποίηση τους με πραγματικά δεδομένα. Καθώς και την μελέτη των ιδιοτήτων της Power Method και της Spectral Graph Theory.

## Βιβλιογραφία

- [1] R. Bapat, “Basic definitions,” in *Graphs and Matrices*, p. 1, Springer, 2011.
- [2] R. Bapat, “Vector spaces associated with a matrix,” in *Graphs and Matrices*, p. 2, Springer, 2011.
- [3] R. Bapat, “Vector spaces associated with a matrix,” in *Graphs and Matrices*, p. 2, Springer, 2011.
- [4] H. Anton and C. Rorres, “Operations on matrices,” in *Elementary linear algebra: applications version*, pp. 53–54, John Wiley & Sons, 2010.
- [5] V. V. G. M. A. AKIVIS, “The matrix of a linear transformation and its determinant,” in *An Introduction to LINEAR ALGEBRA AND TENSORS*, p. 71, DOVER PUBLICATIONS, 1972.
- [6] L. V. Stephen Boyd, “Matrix addition and scalar-matrix multiplication,” in *Introduction to Applied Linear Algebra*, p. 117, Cambridge University, 2018.
- [7] L. V. Stephen Boyd, “Matrix-matrix multiplication,” in *Introduction to Applied Linear Algebra*, p. 177, Cambridge University, 2018.
- [8] L. V. Stephen Boyd, “Zero and identity matrices,” in *Introduction to Applied Linear Algebra*, p. 113, Cambridge University, 2018.
- [9] T. W. Judson, “Kronecker delta,” in *Abstract Algebra Theory and Applications*, p. 139, Orthogonal Publishing L3C, 2018.
- [10] L. V. Stephen Boyd, “Properties of matrix multiplication,” in *Introduction to Applied Linear Algebra*, p. 179, Cambridge University, 2018.

- [11] L. V. Stephen Boyd, “Matrix power,” in *Introduction to Applied Linear Algebra*, p. 186, Cambridge University, 2018.
- [12] L. V. Stephen Boyd, “Matrix power,” in *Introduction to Applied Linear Algebra*, p. 202, Cambridge University, 2018.
- [13] R. Bapat, “Trace and determinant,” in *Graphs and Matrices*, p. 2, Springer, 2011.
- [14] E. R. T.S. Blyth, “Eigenvalues and eigenvectors,” in *Basic Linear Algebra*, p. 154, Springer, 2005.
- [15] R. Bapat, “Characteristic polynomial,” in *Graphs and Matrices*, p. 5, Springer, 2011.
- [16] E. R. T.S. Blyth, “Geometric multiplicity,” in *Basic Linear Algebra*, p. 156, Springer, 2005.
- [17] U. M. J.A. Bondy, “Graphs and their representation,” in *Graph Tgeory*, p. 2, Springer, 2008.
- [18] U. M. J.A. Bondy, “Euler tours,” in *Graph Tgeory*, p. 86, Springer, 2008.
- [19] U. M. J.A. Bondy, “Directed graphs,” in *Graph Tgeory*, p. 31, Springer, 2008.
- [20] U. M. J.A. Bondy, “Special families of graphs,” in *Graph Tgeory*, p. 4, Springer, 2008.
- [21] R. J. W. LOWELL W. BEINEKE, “Graphs and matrices,” in *Topics in Algebraic Graph Theory*, p. 57, Cambridge University Press, 2005.
- [22] R. J. W. LOWELL W. BEINEKE, “Graphs,” in *Graph Theory*, p. 10, West-view Press, 1969.
- [23] R. Bapat, “Graphs,” in *Graphs and Matrices*, pp. 9–10, Springer, 2011.
- [24] R. Bapat, “Adjacency matrix,” in *Graphs and Matrices*, p. 25, Springer, 2011.
- [25] R. Bapat, “Lemma 3.2. linearly independent,” in *Graphs and Matrices*, p. 26, Springer, 2011.

- [26] P. V. MIEGHEM, “1.1 interpretation and contemplation,” in *GRAPH SPECTRA FOR COMPLEX NETWORKS*, p. 3, Cambridge University Press, 2011.
- [27] P. V. MIEGHEM, “A shortest path,” in *GRAPH SPECTRA FOR COMPLEX NETWORKS*, p. 27, Cambridge University Press, 2011.
- [28] P. V. MIEGHEM, “Gerschgorin’s theorem 36 applied to the adjacency matrix,” in *GRAPH SPECTRA FOR COMPLEX NETWORKS*, p. 29, Cambridge University Press, 2011.
- [29] P. V. MIEGHEM, “Diameter of a graph,” in *GRAPH SPECTRA FOR COMPLEX NETWORKS*, p. 41, Cambridge University Press, 2011.
- [30] P. V. MIEGHEM, “Bounds for the largest, positive eigenvalue  $\lambda_1$ ,” in *GRAPH SPECTRA FOR COMPLEX NETWORKS*, p. 46, Cambridge University Press, 2011.
- [31] P. V. MIEGHEM, “Spectral gap,” in *GRAPH SPECTRA FOR COMPLEX NETWORKS*, p. 55, Cambridge University Press, 2011.
- [32] K. L. M. Mark S. Daskin, “2.1 introduction,” in *Location Science*, pp. 21–22, Springer, 2015.
- [33] T. P. R. F. W. EDITH COHEN, DANIEL DELLING, “Introduction,” in *Computing Classic Closeness Centrality, at Scale*, p. 1, 2014.
- [34] J. S. Tore Opsahl, Filip Agneessens, “Degree,” in *Node Centrality in Weighted Networks: Generalizing Degree and Shortest Paths*, p. 6, 2010.
- [35] H. R. Muhammad U. Ilyas, “Eigenvector centrality,” in *Identifying Influential Nodes in Online Social Networks Using Principal Component Centrality*, p. 2.
- [36] D. C. F. RON LARSON, “Power method for approximating eigenvalues,” in *Elementary Linear Algebra*, p. 551, Houghton Mifflin Harcourt, 2009.
- [37] S. A. Konstantinos Oikonomou, George Koufoudakis, “Probabilistic flooding,” in *Probabilistic Flooding Coverage Analysis in Large Scale Wireless Networks*, pp. 1–6, IEEE, 2014.

- [38] S. A. I. S. Konstantinos Oikonomou, George Koufoudakis, “Analysis of spectral properties for efficient coverage under probabilistic flooding,” in *2018 IEEE 19th International Symposium on “A World of Wireless, Mobile and Multimedia Networks” (WoWMoM)*, pp. 1–9, IEEE, 2018.
- [39] G. Koufoudakis, “Summary and conclusions,” in *Analysis of Probabilistic Information Dissemination in Wireless Networks Employing Algebraic Graph Theory Elements*, pp. 109–110, IONIAN UNIVERSITY, 2019.
- [40] K. G. S. A. Konstantinos Oikonomou, George Koufoudakis, “Probabilistic flooding coverage analysis for efficient information dissemination in wireless networks,” in *Computer Networks, Vol 140*, pp. 51–61, ELSEVIER, 2018.
- [41] K.-M. C. Bang Ye Wu, “Shortest-paths trees,” in *Spanning Trees and Optimization Problems*, pp. 34–36, CHAPMAN And HALL/CRC, 2003.

## Παραρτήματα



## Παράρτημα Ι

### Δημιουργία δικτύων κώδικας

```
1 clear all
2 num_nodes = input('Give the number of nodes > ');
3 grid_x = input('Give the size of x grid > ');
4 grid_y = input('Give the size of y grid > ');
5 networkID = input('Give the ID for network > ');
6
7 %Create and Initialaze the tables of x and y
8 x_num_nodes = zeros(1,num_nodes);
9 y_num_nodes = zeros(1,num_nodes);
10
11 %weight_of_vertex = zeros(1,num_nodes);
12 %Generate the x and y for each node
13 for i = 1 : num_nodes
14     x_num_nodes(i) = mod(rand * grid_x, grid_x);
15     y_num_nodes(i) = mod(rand * grid_y, grid_y);
16 end
17
18 %random weight for the nodes
19 weight_of_vertex = rand(num_nodes,1);
20 %RC
21 rc_list = [0.1;0.15;0.2;0.25;0.3];
22
23 %Create a folder for the networks
24 folder_name = strcat('Network_',num2str(networkID),'_of_10_with_',
    num2str(num_nodes));
25 folder_c = char(folder_name);
26 mkdir(folder_c);
```

```

27
28 % Euclidean distance two nodes and give value in the adjstice matrix
29 for r = 1 : length(rc_list)
30     %Create and Initialaze the adjstice matrix of graf
31     adj = zeros(num_nodes,num_nodes);
32     for i = 1:num_nodes
33         x_point_1 = x_num_nodes(i);
34         y_point_1 = x_num_nodes(i);
35         for j = 1:num_nodes
36             x_point_2 = x_num_nodes(j);
37             y_point_2 = x_num_nodes(j);
38             if i == j
39                 adj(i,j) = 0;
40             else
41                 euclidean_dis = sqrt(sum (sum(x_point_2 - x_point_1) .^2 + sum(
42                     y_point_2 - y_point_1).^2 ));
43                 if euclidean_dis <= rc_list(r)
44                     adj(i,j) = 1;
45                 else
46                     adj(i,j) = 0;
47                 end
48             end
49         end
50     is_connected = isconnected(adj);
51     if is_connected == 0
52         fprintf('The network is not Connected\n');
53     else
54         fprintf('The network is Connected\n');
55         file_name = strcat("Simulation_graph-",num2str(networkID), '_with_rc_',
56             num2str(rc_list(r)), '_with_', num2str(num_nodes), '_nodes.mat');
57         file_c = char(file_name);
58         rc = rc_list(r);
59         save(file_name, 'adj', 'num_nodes', 'weight_of_vertex', 'rc');
60
61         %Move the files in the folder
62         movefile(file_c, folder_c);
63     end
end

```

Listing I.1: Δημιουργία δικτύων

## Παράρτημα II

### Υπολογισμός συνολικού κόστους

```
1 function C = closeness_with_weight_of_vertex(adj, weight_of_vertex)
2     num_nodes = length(adj);
3     C = zeros(num_nodes,1); % initialize closeness vector
4     for s = 1 : num_nodes
5         for u = 1 : num_nodes
6             [distance, ~] = dijkstra(adj, u, s);
7             C(s) = C(s) + (distance * weight_of_vertex(u));
8         end
9     end
10 end
```

Listing II.1: Συνοληκό κόστος

## Παράρτημα III

### Shortest path tree

```
1 function calculate_the_total_cost = calculate_the_total_cost_of_node_m(  
    position_of_median_from_mesuers , adj , num_nodes , weight_of_vertex )  
2     %%% Initialize variables for the calculate the  
3     %%% cost from the position of node  
4     u = position_of_median_from_measure ;  
5     path_of_nodes = zeros ( num_nodes , num_nodes ) ;  
6     treeAdj_from_adj = zeros ( num_nodes , num_nodes ) ;  
7     %%% Total cost of tree with facilites the  
        position_of_median_from_measure  
8     Cost_of_median_tree = zeros ( 1 , 1 ) ;  
9  
10    %%% Find the Shortest path from adj with root the vertex u  
11    for v = 1 : num_nodes  
12        [ ~ , path ] = dijkstra ( adj , u , v ) ;  
13        for j = 1 : max ( size ( path ) )  
14            path_of_nodes ( v , j ) = path ( j ) ;  
15        end  
16    end  
17  
18    %%% Crete the Shortest path tree from adjacency matrix  
19    for i_for_tree = 1 : num_nodes  
20        for j = 1 : num_nodes  
21            if ~ ( ( j + 1 ) > num_nodes ) % Chech the bounds of matrix  
22                if path_of_nodes ( i_for_tree , j ) ~= 0 && path_of_nodes (   
                    i_for_tree , j + 1 ) ~= 0  
23                    treeAdj_from_adj ( path_of_nodes ( i_for_tree , j ) , path_of_nodes (   
                        i_for_tree , j + 1 ) ) = 1 ;
```

```

24         treeAdj_from_adj(path_of_nodes( i_for_tree ,j+1),path_of_nodes
25         ( i_for_tree ,j)) = 1;
26     end
27 end
28 end
29
30 for n = 1 : num_nodes
31     [distanse ,~] = dijkstra(treeAdj_from_adj ,n,u);
32     Cost_of_median_tree = Cost_of_median_tree + (distanse *
33     weight_of_vertex(u));
34 end
35 calculate_the_total_cost = Cost_of_median_tree;
36 end
37

```

Listing III.1: Shortest path tree

## Παράρτημα IV

### Degree Centrality

```
1 %Degree centrality
2 f_degree = @() degrees(adj);
3 stop_measure_Degree Centrality = timeit(f_degree);
4 [degree,~,~] = degrees(adj);
5 max_degree = max(degree);
6 position_of_median_with_degree Centrality = find(degree == max_degree);
7 num_of_position_of_median_with_degree Centrality = numel(
    position_of_median_with_degree Centrality);
8
9 for i = 1 : num_of_position_of_median_with_degree Centrality
10     the_list_of_positions_of_the_measures(end+1) =
        position_of_median_with_degree Centrality(i);
11 end
```

Listing IV.1: Degree Centrality

## Παράρτημα V

### Closeness Centrality

```
1 % f_clonsenes = @() closeness_with_weight_of_vertex(adj,  
    weight_of_vertex);  
2  
3 %Clonsenes with out weigth(- load) in vertices  
4 f_clonsenes = @() closeness(adj);  
5 stop_measure_Closeness_centrality = timeit(f_clonsenes);  
6 Cost_of_median = closeness(adj);  
7 max_cost = max(Cost_of_median);  
8 position_test = find (Cost_of_median == max_cost);  
9  
10 %Find the position of the node  
11 position_of_median = find(Cost_of_median == max_cost);  
12 num_of_position_of_median_with_Closeness_centrality = numel(  
    position_of_median);  
13  
14 for i =1: num_of_position_of_median_with_Closeness_centrality  
15 the_list_of_positions_of_the_measures(end+1) = position_of_median(i);  
16 end
```

Listing V.1: Degree Centrality

## Παράρτημα VI

### Eigenvector Centrality

```
1 %Eigenvector centrality
2
3 f_eig_vector = @() eig(adj,1);
4 stop_measure_Eigenvector_centrality = timeit(f_eig_vector);
5 %Find the largest egenvalue and the corresponding eigenvector
6 [eig_vector , eig_value] = eig(adj,1);
7 eig_vector = -eig_vector;
8 the_max_value_of_eig_vector = max(eig_vector);
9 position_of_median_with_eig_veceor_centrality = find(eig_vector ==
    the_max_value_of_eig_vector);
10 %Transpose the Vector
11 position_of_median_with_eig_veceor_centrality =
    position_of_median_with_eig_veceor_centrality';
12 num_position_of_median_with_eig_veceor_centrality = numel(
    position_of_median_with_eig_veceor_centrality);
13
14 for i = 1 : num_position_of_median_with_eig_veceor_centrality
15     the_list_of_positions_of_the_measures(end+1) =
        position_of_median_with_eig_veceor_centrality(i);
16 end
```

Listing VI.1: Eigenvector Centrality

## Παράρτημα VII

### Eigenvector Centrality με την χρήση της Power Method

```
1 %Initialize the vector for Power Method = [1,1, ... 1]
2 x0 = ones(1,num_nodes);
3 %A set of numbers where each number represents the number of loops for
  the
4 % Power Method.
5 num_of_terms = [5;10;100;200;];
6 % The number of the elements in num_of_terms
7 num_loop = numel(num_of_terms);
8 % The vectors is the variable how store all the vectors of the function
  cul_eigvector_sentrality_Power_Method
9 vectors = zeros(num_nodes,num_loop);
10 position_of_median_with_eigvector centrality_Power_Method = zeros(1,
  num_loop);
11 stop_measure_Eigenvector centrality_Power_method = zeros(1,num_loop);
12
13 for i = 1 : num_loop
14     f = @( ) cul_eigvector_sentrality_Power_Method(adj,x0,num_of_terms(i))
        ;
15     stop_measure_Eigenvector centrality_Power_method(i) = timeit(f);
16     vecrtor = cul_eigvector_sentrality_Power_Method(adj,x0,num_of_terms(
        i));
17     Max_vector = max(vecrtor);
18 % It is possibility to exist more than one node in the network with
  the
```

## 64 Παράρτημα VII. Eigenvector Centrality με την χρήση της Power Method

```
19 % largest value of eigenvector centrality
20 Temp_position_of_median_with_eigenvector_centrality_Power_Method =
    find(Max_vector == vecrtor,1,'last');
21 position_of_median_with_eigenvector_centrality_Power_Method(i) =
    Temp_position_of_median_with_eigenvector_centrality_Power_Method;
22 for j = 1 : num_nodes
23     vectors(j,i) = vecrtor(j);
24 end
25 end
26
27 for i = 1 : num_loop
28     the_list_of_positions_of_the_measures(end+1) =
        position_of_median_with_eigenvector_centrality_Power_Method(i);
29 end
30
```

Listing VII.1: Eigenvector Centrality

```
1 function [eigvector_centrality] =
    cul_eigvector_sentrality_Power_Method(adj,x0,num_of_terms)
2 Xn = x0 * adj;
3 Xn = Xn/norm(Xn);
4 for i = 1 : num_of_terms -1
5     Xn = Xn * adj;
6     Xn = Xn/norm(Xn);
7 end
8 eigvector_centrality = Xn;
9 end
```

Listing VII.2: Eigenvector Centrality Function

## Γλωσσάρι Ξενικών Όρων

|                       |                                                               |
|-----------------------|---------------------------------------------------------------|
| Adjacency Matrix      | Πίνακας γειτνίασης                                            |
| Shortest Path         | Ελάχιστο μονόπατη                                             |
| Link Weight Structure | Δομή Βάρους                                                   |
| Rank                  | Τάξη του πίνακα                                               |
| Robustness of a graph | Η αντίσταση του γραφήματος στη ζημία ευρωστία ενός γραφήματος |
| Singular matrix       | Ιδιόμορφος πίνακας                                            |
| Trace                 | Τίχνος                                                        |
| Undirected graphs     | Μη-κατευθυνομένα γραφήματα                                    |
| Directed graphs       | Κατευθυνομένα γραφήματα                                       |
| Conected              | Συνεκτικό γράφημα                                             |
| Complete              | Πλήρες                                                        |
| Column space          | Χώρος των στηλών                                              |
| Row space             | Χώρος των γραμμών                                             |
| Multigraph            | Πολυγράφημα                                                   |
| Nullspace             | Μηδενόχωρος                                                   |
| τλ Нуллиψ             | Μηδενηκότητα                                                  |
| Pseudograph           | Ψευδογράφημα                                                  |
| Simple                | Απλό γράφημα                                                  |

66 Παράρτημα VII. *Eigenvector Centrality με την χρήση της Power Method*

Weighted Έμβανο γράφημα  
graph

## Συμβολισμοί Μεγεθών

|               |                                                   |
|---------------|---------------------------------------------------|
| $\mathbb{F}$  | Το σύνολο των πραγματικών ή των μιγαδικών αριθμών |
| $\mathbb{O}$  | Ο μηδενικός πίνακας                               |
| $D$           | Η διαγώνιος του πίνακα                            |
| $I_n$         | Ο ταυτοτικός πίνακας                              |
| $\det(A)$     | Η ορίζουσα του πίνακα $A$                         |
| $\lambda_1$   | Η μεγαλύτερη ιδιοτιμή του πίνακα                  |
| $\mathcal{A}$ | Ο πίνακας γειτνίασης                              |
| $\rho$        | Η διάμετρος του δικτύου                           |
| $A^T$         | Ο ανάστροφος πίνακας του $A$                      |
| $V$           | Το σύνολο των κόμβων                              |
| $E$           | Το σύνολο των ζεύξεων                             |