

University Of Thessaly



Complex Networks

ECE434

---

## Problem Set 3

---

**Authors:**

Lefkopoulou Eleni-Maria - 2557

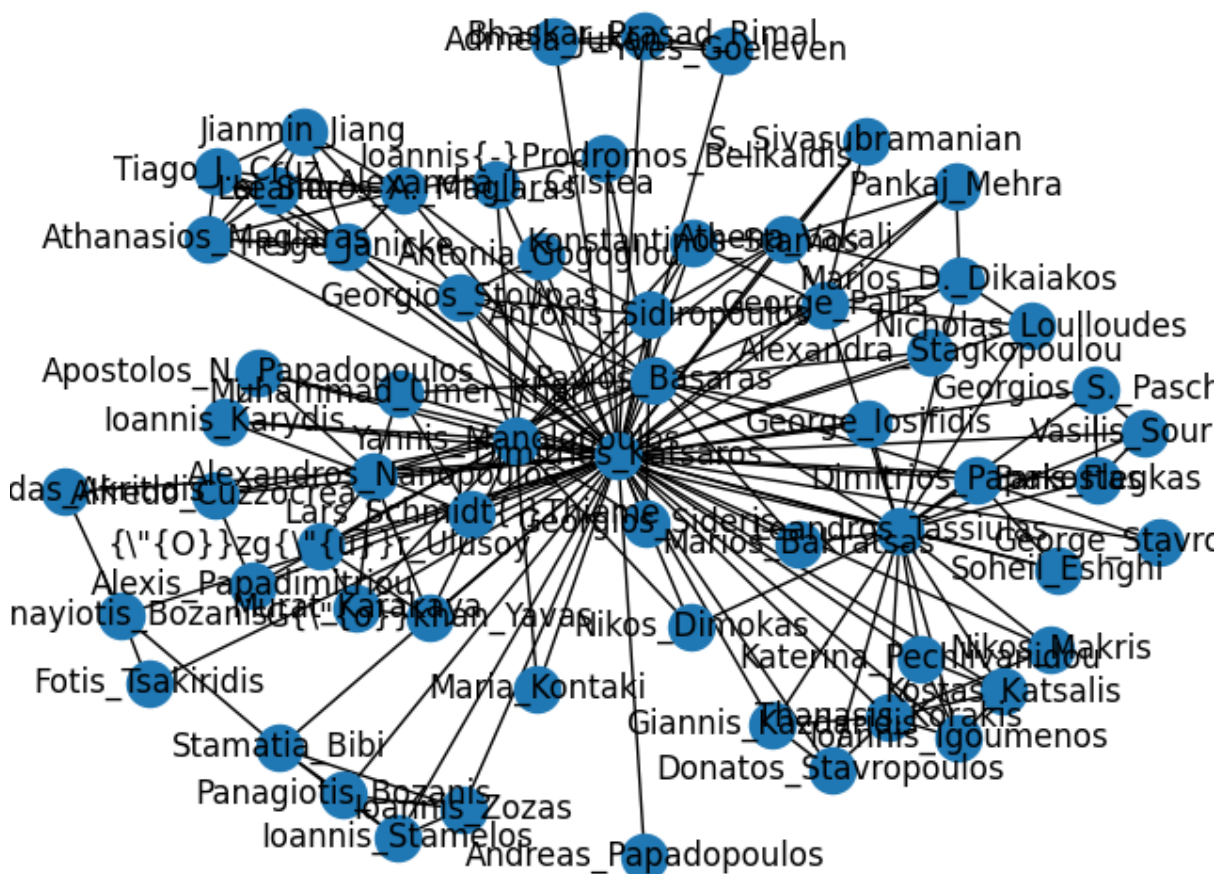
Karanika Athanasia - 2530

9 Μαΐου 2021

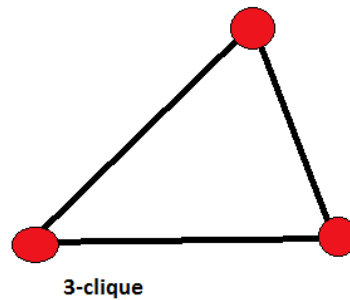
## Πρόβλημα-01

Για δίκτυο των co-authors για το Δημήτρη Κατσαρό μας ζητείται να υπολογίσουμε όλες τις (επικαλυπτόμενες και μη) κοινότητες με τον αλγόριθμο Clique Percolation Method (CPM).

Όσον αφορά την ευρεση κοινοτήτων αντιμετωπίζεται ως διαδικασία "percolation" από τον αλγόριθμο. Ξεκινώντας από μια  $k$ -clique, κόμβοι προοδευτικά προσκολλώνται με κριτήριο του να είναι προσβάσιμοι διαμέσου μιας προσκείμενης κλίκας (2 κλίκες είναι προσκείμενες εάν μοιράζονται  $k-1$  κόμβους). Ο αλγόριθμος υποστηρίζει επικαλυπτόμενες κοινότητες. Όσο μεγαλύτερες  $k$ -cliques υπάρχουν σε ένα δίκτυο τόσο λιγότερες κοινότητες θα βρεθούν.



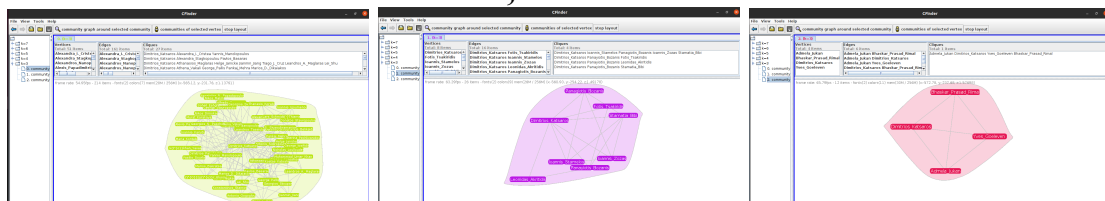
Με τη χρήση του λογισμικού CFinder υπολογίστηκαν οι παρακάτω communities για διάφορα  $k$ .



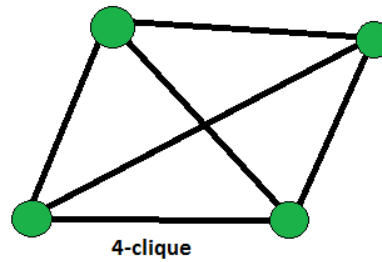
- Όπου 3-cliques →

- **K = 3 communities**

{**0**: Dimitrios Katsaros Alexandra I. Cristea Yannis Manolopoulos Alexandra Stagkopoulou Pavlos Basaras Alexandros Nanopoulos Apostolos N. Papadopoulos Gökhan Yavas Murat Karakaya Özgür Ulusoy Alexis Papadimitriou Alfredo Cuzzocrea Antonia Gogoglou Antonis Sidiropoulos Athanasios Maglaras Helge Janicke Jianmin Jiang Tiago J. Cruz Athena Vakali George Pallis George Stavropoulos Leandros Tassiulas Nicholas Loulloudes Pankaj Mehra S. Sivasubramanian Thanasis Korakis Dimitrios Papakostas Soheil Eshghi Donatos Stavropoulos Giannis Kazdaridis George Iosifidis Georgios S. Paschos Georgios Sideris Georgios Stoupas Ioannis Igoumenos Ioannis Karydis Ioannis-Prodrornos Belikaidis Leandros A. Maglaras Katerina Pechlivanidou Kostas Katsalis Konstantinos Stamos Lars Schmidt-Thieme Lei Shu Maria Kontaki Marios Bakratsas Marios D. Dikaiakos Muhammad Umer Khan Nikos Dimokas Nikos Makris Paris Flegkas Vasilis Surlas }, {**1**: Dimitrios Katsaros Ioannis Stamelos Panagiotis Bozanis Panayiotis Bozanis Fotis Tsakiridis Ioannis Zozas Stamatia Bibi Leonidas Akritidis }, {**2**: Admela Jukan Dimitrios Katsaros Yves Goeleven Bhaskar Prasad Rimal}

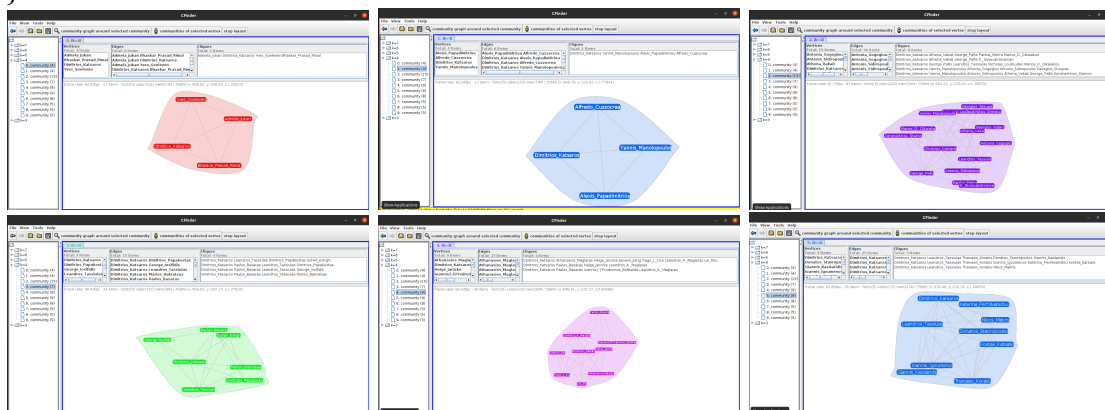


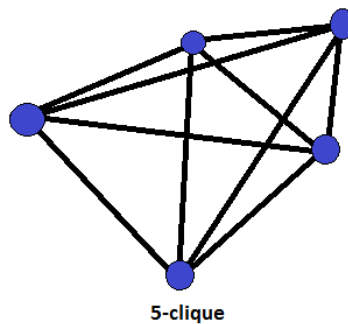
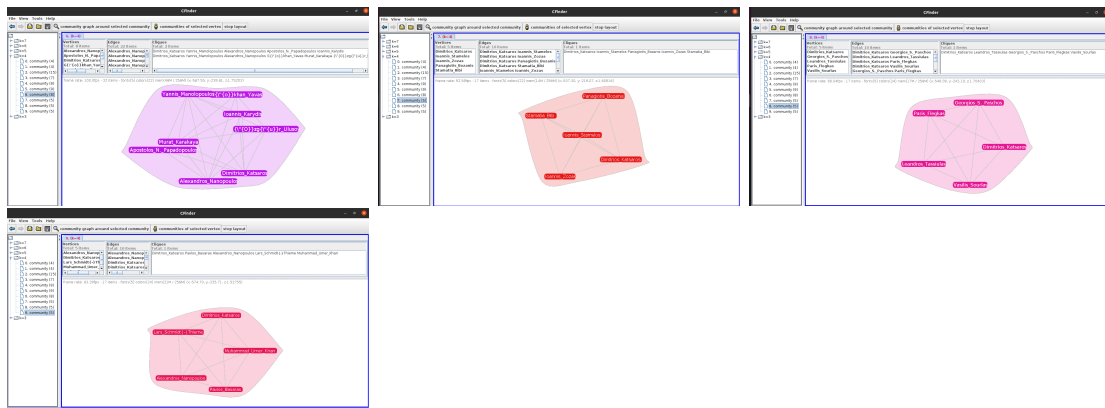
• Όπου 4-cliques —>



• **K = 4 communities**

{0: Admela Jukan Dimitrios Katsaros Yves Goeleven Bhaskar Prasad Rimal } {1: Dimitrios Katsaros Yannis Manolopoulos Alexis Papadimitriou Alfredo Cuzzocrea } {2: Dimitrios Katsaros Yannis Manolopoulos Antonia Gogoglou Antonis Sidiropoulos Athena Vakali George Pallis Leandros Tassiulas Nicholas Loulloudes Pankaj Mehra S. Sivasubramanian Georgios Sideris Georgios Stoupas Konstantinos Stamos Marios D. Dikaiakos Nikos Dimokas } {3: Dimitrios Katsaros Pavlos Basaras Leandros Tassiulas Dimitrios Papakostas Soheil Eshghi George Iosifidis Marios Bakratsas } {4: Dimitrios Katsaros Pavlos Basaras Athanasios Maglaras Helge Janicke Jianmin Jiang Tiago J. Cruz Ioannis-Prodrornos Belikaidis Leandros A. Maglaras Lei Shu } {5: Dimitrios Katsaros Leandros Tassiulas Thanasis Korakis Donatos Stavropoulos Giannis Kazdaridis Ioannis Igoumenos Katerina Pechlivanidou Kostas Katsalis Nikos Makris } {6: Dimitrios Katsaros Yannis Manolopoulos Alexandros Nanopoulos Apostolos N. Papadopoulos Gökhan Yavas Murat Karakaya Özgür Ulusoy Ioannis Karydis } {7: Dimitrios Katsaros Ioannis Stamelos Panagiotis Bozanis Ioannis Zozas Stamatia Bibi } {8: Dimitrios Katsaros Leandros Tassiulas Georgios S. Paschos Paris Flegkas Vasilis Sourlas } {9: Dimitrios Katsaros Pavlos Basaras Alexandros Nanopoulos Lars Schmidt-Thieme Muhammad Umer Khan }



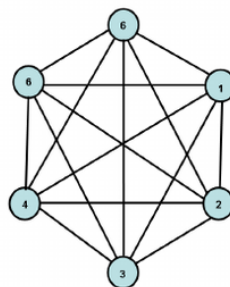


• Όπου 5-cliques →

#### • k=5 communities

{**0**: Dimitrios Katsaros Yannis Manolopoulos Alexandros Nanopoulos Apostolos N. Papadopoulos Ioannis Karydis } {**1**: Dimitrios Katsaros Yannis Manolopoulos Antonia Gogoglou Antonis Sidiropoulos Georgios Stoupas } {**2**: Dimitrios Katsaros Ioannis Stamelos Panagiotis Bozanis Ioannis Zozas Stamatia Bibi } {**3**: Dimitrios Katsaros George Pallis Leandros Tassioulas Nicholas Loulloudes Marios D. Dikaiakos } {**4**: Dimitrios Katsaros Athena Vakali George Pallis Pankaj Mehra Marios D. Dikaiakos } {**5**: Dimitrios Katsaros Leandros Tassioulas Thanasis Korakis Donatos Stavropoulos Giannis Kazdaridis} {**6**: Dimitrios Katsaros Leandros Tassioulas Georgios S. Paschos Paris Flegkas Vasilis Sourlas } {**7**: Alexandros Nanopoulos Lars Schmidt-Thieme Muhammad Umer Khan } {**8**: Dimitrios Katsaros Yannis Manolopoulos Alexandros Nanopoulos Gökhan Yavas Murat Karakaya Özgür Ulusoy } {**9**: Dimitrios Katsaros Leandros Tassioulas Thanasis Korakis Ioannis Igoumenos Katerina Pechlivanidou Kostas Katsalis } {**10**: Dimitrios Katsaros Yannis Manolopoulos Antonis Sidiropoulos Athena Vakali George Pallis Konstantinos Stamos } {**11**: Dimitrios

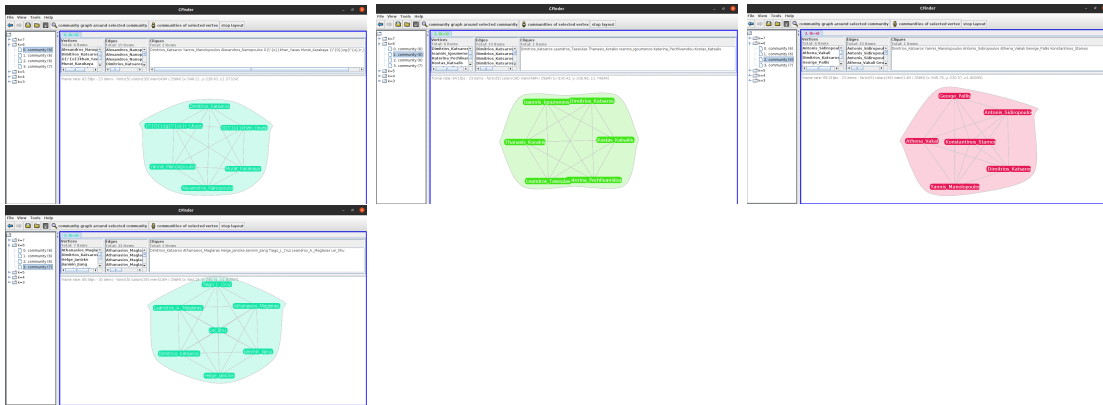
Katsaros Athanasios Maglaras Helge Janicke Jianmin Jiang Tiago J. Cruz  
Leandros A. Maglaras Lei Shu }



• Όπου 6-cliques  $\rightarrow$

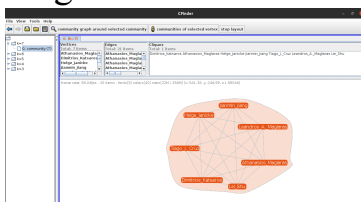
•  $k = 6$  communities

{0: Dimitrios Katsaros Yannis Manolopoulos Alexandros Nanopoulos Gökhan Yavas Murat Karakaya Özgür Ulusoy } {1: Dimitrios Katsaros Leandros Tassioulas Thanasis Korakis Ioannis Igoumenos Katerina Pechlivanidou Kostas Katsalis } {2: Dimitrios Katsaros Yannis Manolopoulos Antonis Sidiropoulos Athena Vakali George Pallis Konstantinos Stamos } {3: Dimitrios Katsaros Athanasios Maglaras Helge Janicke Jianmin Jiang Tiago J. Cruz Leandros A. Maglaras Lei Shu }



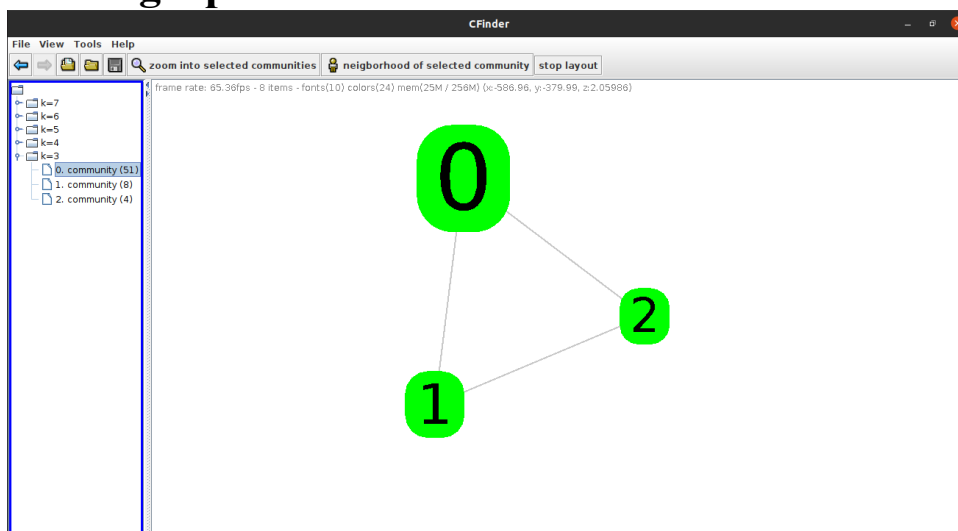
- **k = 7 communities**

{0: Dimitrios Katsaros Athanasios Maglaras Helge Janicke Jianmin Jiang  
Tiago J. Cruz Leandro A. Maglaras Lei Shu }

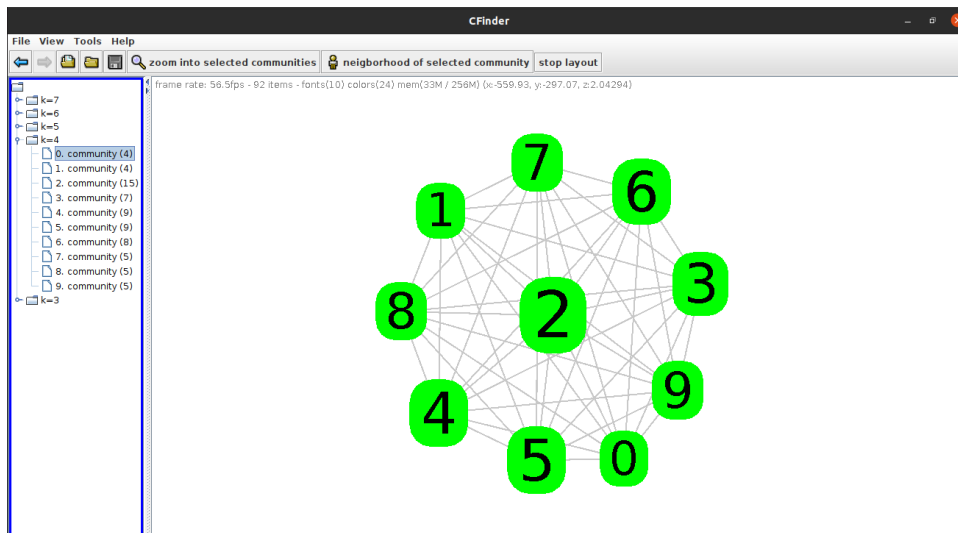


Επίσης μέσω του λογισμικού μπορούμε να δούμε οπτικοποιημένα το graph of communities όπου ποσοτικά ανάλογα με τα μέλη της κοινότητας αλλάζει και το σχήμα των communities.

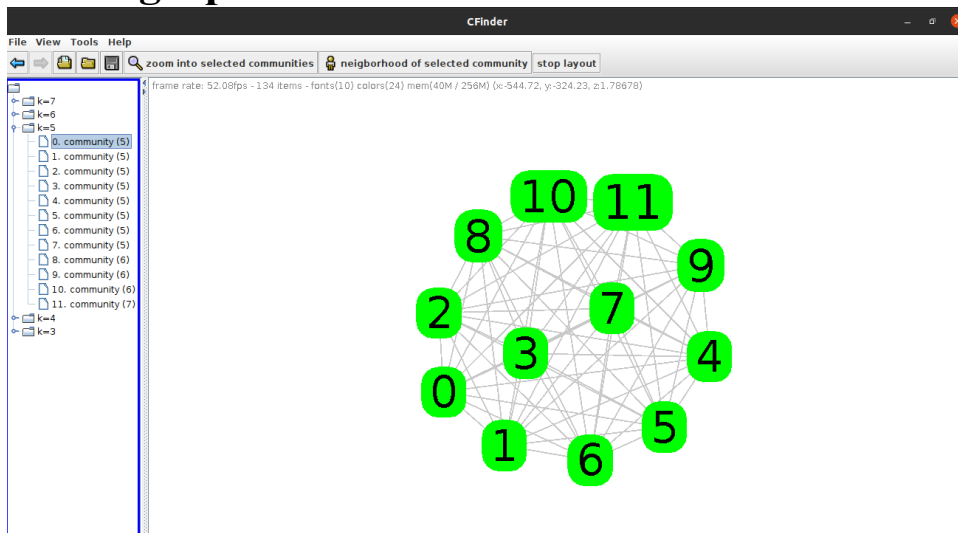
- **K = 3 graph of communities**



- **K = 4 graph of communities**

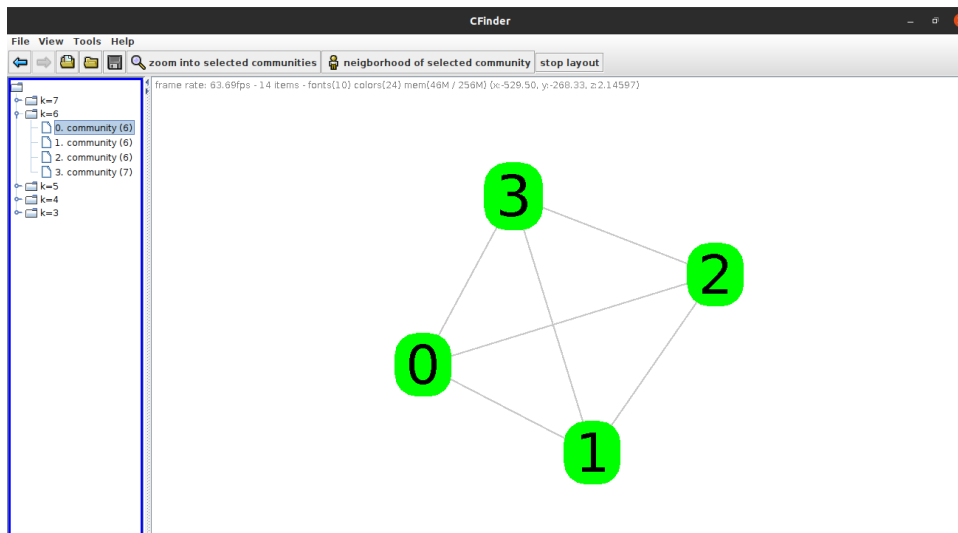


- **K = 5 graph of communities**



- **K = 6 graph of communities**

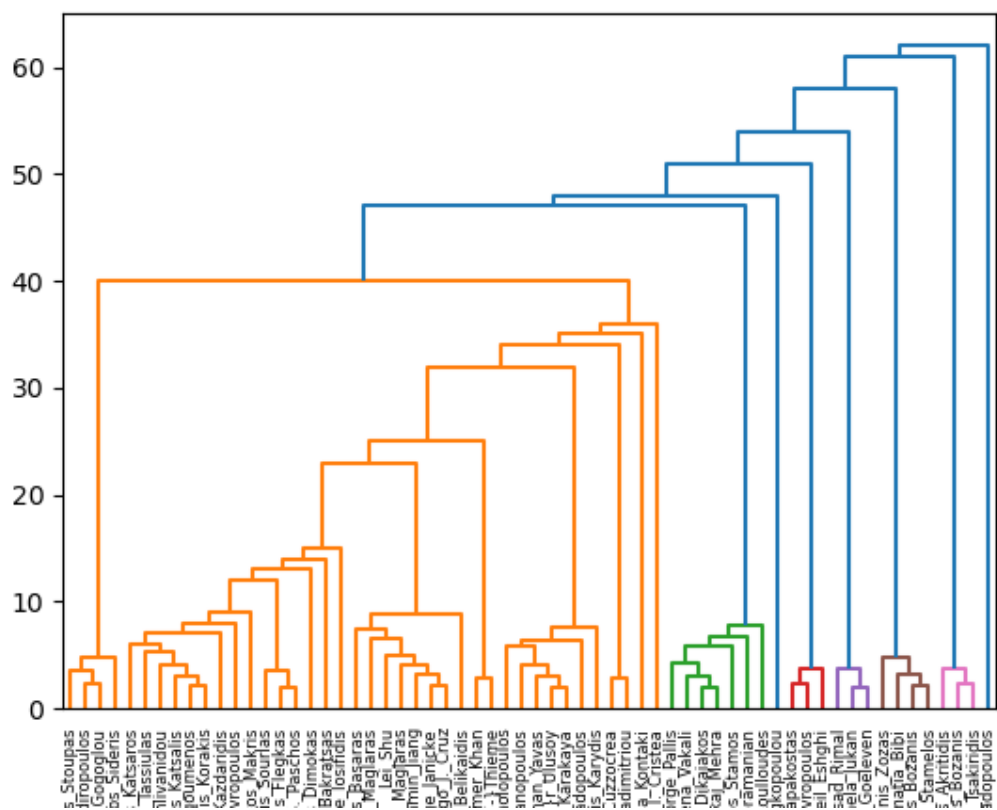


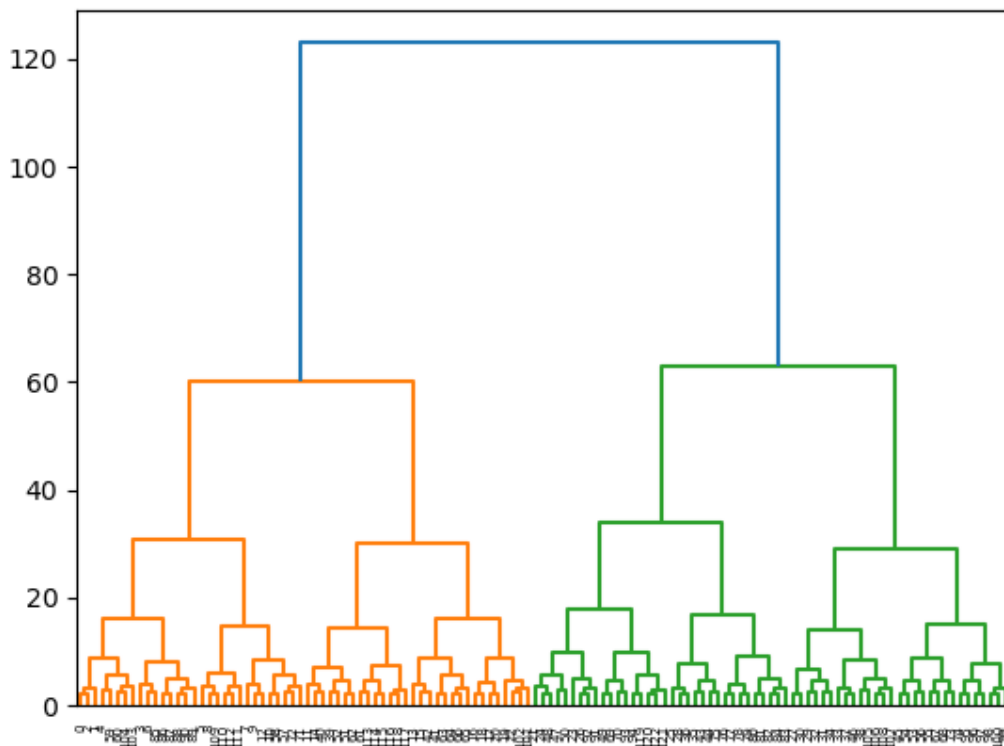


Αυτό που περιγράφεται στο άρθρο είναι ένας παραλληλοποιημένος αλγόριθμος σαν τον CPM αλλά ονομάζεται COS και κάνει τη χρήση παραπάνω νημάτων αν η μνήμη το επιτρέπει ώστε να επέξεργάζονται τον πίνακα με τους κόμβους και να υπολογίζουν τις κοινότητες με καλύτερες αποδόσεις. Δεν το επιτρέπουν όλα τα συστήματα αλλά αν επιτρέπεται έχει σαφώς καλύτερες επιδόσεις από τον CPM.

## Πρόβλημα-02

Αρχικά το δειροδιάγραμμα για τον αλγόριθμο grivan newman με υολογισμό της betweenness .μετά απο κάθε αφαίρεση ακμής είναι:





Παρατηρούμε ότι σειρά με την οποία αφαιρούνται η ακμές επιφέρει σημαντικές αλλαγές στο αποτέλεσμα βλέπουμε οτι στην περίπτωση που δεν επαναυπολογίζουμε την betweenness μετά από κάθε αφαίρεση ακμής το αποτέλεσμα είναι πιο συμμετρικό γιατί εξ' αρχής ο γράφος μας δείχνει να έχει κόμβους με παρόμοιο αριθμο συνδέσεων και σχεδόν οι περισσότεροι κόμβοι συνδέονται με όμοιους τους οπότε αρχικά οι betweenness είναι παρόμοιες όμως βλέπουμε οτι αφαιρώντας μια ακμή μένουν λίγοι κόμβοι στην μια κοινότητα και πολλοί στην άλλη(α' περίπτωση) δηλαδή αφαιρώντας μια ακμή αποκόπτει λίγους κόμβους από πολλούς πράγμα που δεν συμβαίνει στην β' περίπτωση αφού από την αρχή που περίπου όλες οι ακμές έχουν την ίδια μετρική προλαθορίζεται το αποτέλεσμα. Αλγόριθμοι για σύγκριση δύο δέντρων

- **Classic Tree Edit Distance** βασίζεται στο String Matching

όπου η ιδέα είναι ότι προσπαθούμε να κάνουμε δύο strings όμοια με τρεις κινήσεις , εισαγωγή γράμματος διαγραφή και μεταφορά κάθε πράξη έχει ένα κόστος και η edit distance ορίζεται ως το ελάχιστο κόστος μετατροπής. Στην περίπτωση του δέντρου , αυτό πρέπει να είναι ordered και labeled δηλαδή κάθε δέντρο διατρέχεται με συγκεκριμένο order (post past pre) και κάθε κόμβος έχει ένα γράμμα ορίζονται πάλι 3 λειτουργίες διαγραφή , μετονομασία και εισαγωγή , η ομοιότητα των δέντρων είναι αντιστρόφως ανάλογη του κόστους για την μετατροπή του ενός δέντρου στο άλλο

- **belief propagation (BP)** εισάγει μια μετρική ομοιότητας μεταξύ του 0 και 1 και βασίζεται στο ότι ο ίδιος κόμβους στα δύο δέντρα είναι όμοιοι εάν και οι γείτοντες τους είναι όμοιοι, και αναδρομικά οι γείτονες είναι όμοιοι αν και οι γείτονες τους είναι όμοιοι.
- **Feature extraction** Η βασική ιδέα πίσω από αυτές τις μεθόδους είναι ότι παρόμοια γραφήματα μοιράζονται πιθανώς ορισμένες ιδιότητες, όπως κατανομή βαθμού, διάμετρος, ιδιοτιμές. Μετά την εξαγωγή αυτών των χαρακτηριστικών, εφαρμόζεται μια μετρική ομοιότητας προκειμένου να εκτιμηθεί η ομοιότητα μεταξύ των συγκεκριμένων χαρακτηριστικών και, ισοδύναμα, η ομοιότητα μεταξύ των γραφημάτων.

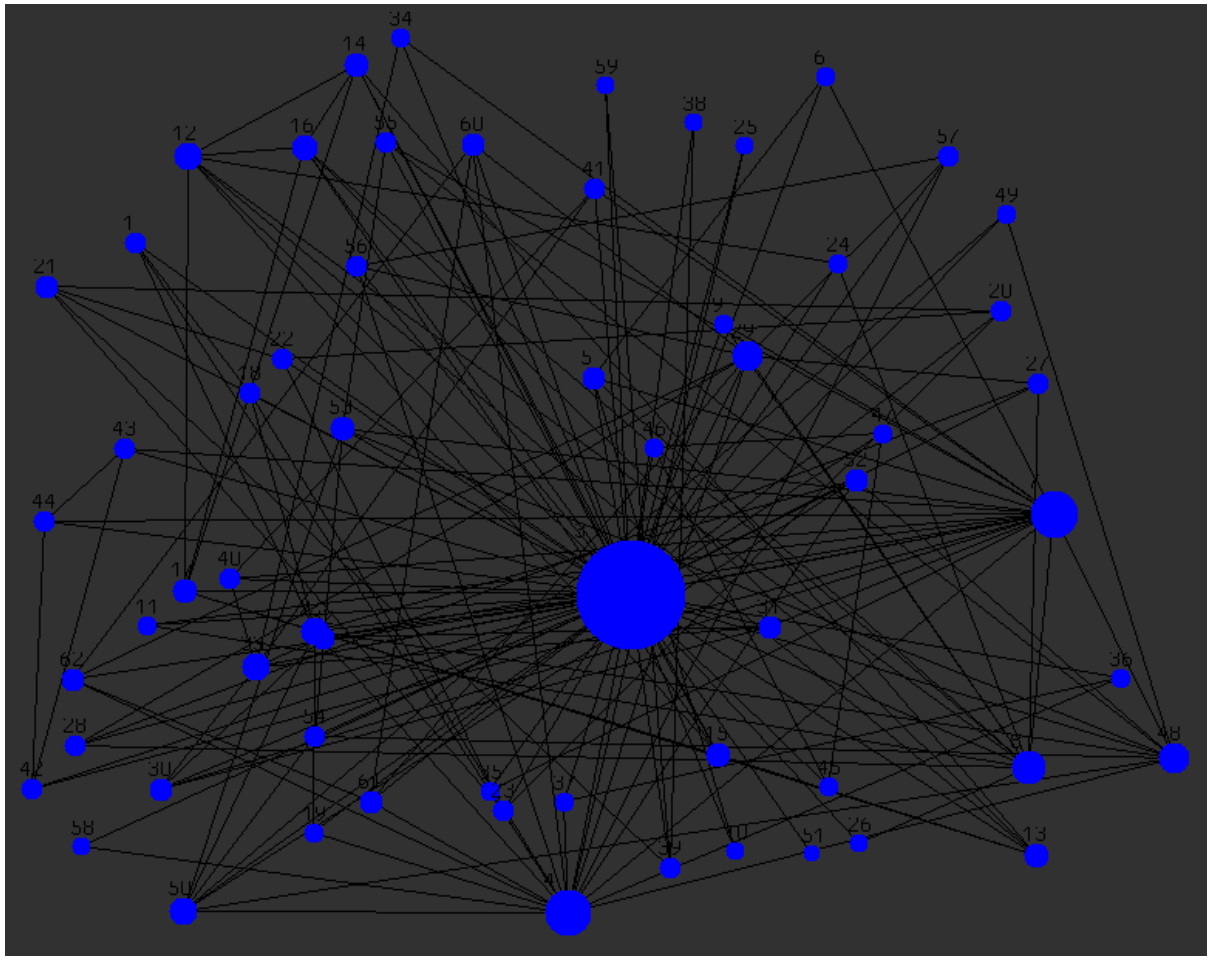
Στην δική μας περίπτωση ταιριάζει μια μίξη μεταξύ της πρώτης και της δεύτερης μεθόδου δηλαδή θα διατρέξουμε τα δέντρα με συγκεκριμένη σειρά και θα αναθέσουμε στους κόμβους ένα γράμμα με την σειρά που θα γίνει η διαπέραση και το κόστος που θα έχει η μετατροπή του ενός string στο άλλο η ομοιότητα θα είναι αντιστρόφως ανάλογη

### Πρόβλημα-03

Στο δίκτυο των co-authors φια το Δημήτριο Κατσαρό μας ζητείται να υπολογίσουμε όλες τις κοινότητες με τον αλγόριθμο CiBC.

Όσον αφορά τον αλγόριθμό αυτό αποτελείται από 3 φάσεις. Στην πρώτη φάση γίνεται υπολογισμός του node betweenness centrality που μετράει πόσο κεντρικός είναι ο κόμβος στο γράφημα. Στο τέλος αυτής της φάσης γίνεται κατάταξη των κόμβων με βάση το node BC. Στη δεύτερη ξεκινώντας από τους κόμβους με τη μικρότερη BC και μαζεύοντας τους 1-hop γείτονες τους γίνεται η αρχικοποίηση των αρχικών κλικών. Θεωρητικά θα μπορούσε ο αόριθμος να αρχίζει και από τον κόμβο με το μεγαλύτερο BC αλλά επειδή θα ήταν ο πιο κεντρικός, δηλαδή ο πιο συνδεδεμένος θα υπήρχε μεγάλη πιθανότητα να σχηματιστεί από την αρχή μόνο μια κοινότητα. Στην τρίτη φάση σχηματίζονται πίνακες τετραγωνικοί όπου στην διαγώνιο έχουν τα link μεταξύ των κόμβων τις κοινότητες ενώ στα υπόλοιπα κελιά τα link προς κόμβους εκτός την κοινότητας. Υπολογίζουμε τα διάφορα κλάσματα  $\frac{link_{out}}{link_{in}}$  για να δούμε ποια κοινότητα έλκεται από ποια. Ο αλγόριθμος συνεχίζεται μέχρι  $\frac{link_{out}}{link_{in}} < s$  όπου  $s$  είναι ένας παράγοντας για να προσδιοριστεί η ισχύ των κοινοτήτων. Συνήθως τίθεται ίσο με 1.

Παρακάτω βλέπουμε την αντιστοιχία ονόματος συγγραφέα και του αριθμού που του αντιστοιχεί στο γράφημα και στις υπολογισμένες κοινότητες υπολογισμό των communities:



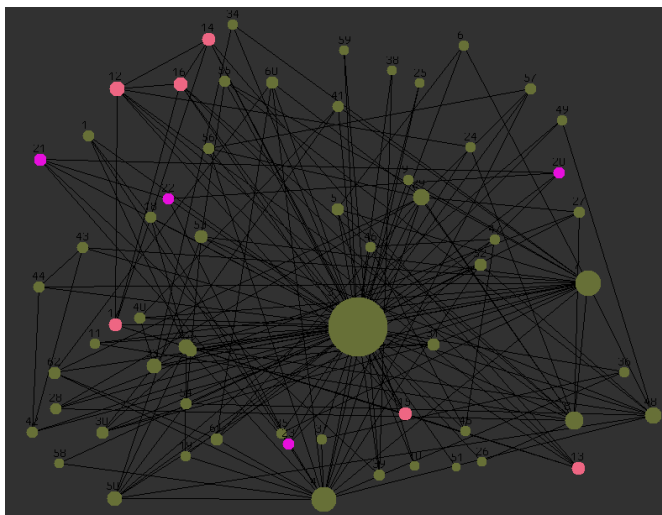
Author's Name	Name
Georgios Stoupas	1
Antonis Sidiropoulos	2
Dimitrios Katsaros	3
Yannis Manolopoulos	4
Dimitrios Papakostas	5
Soheil Eshghi	6
Leandros Tassiulas	7
Pavlos Basaras	8
George Iosifidis	9
George Stavropoulos	10
Marios Bakratsas	11
Leandros A. Maglaras	12
Lei Shu	13

Author's Name	Name
Athanasios Maglaras	14
Jianmin Jiang	15
Helge Janicke	16
Tiago J. Cruz	17
Antonia Gogoglou	18
Georgios Sideris	19
Ioannis Zozas	20
Stamatia Bibi	21
Panagiotis Bozanis	22
Ioannis Stamelos	23
Ioannis- Prodromos Belikaidis	24
Alexandra I. Cristea	25
Alexandra Stagkopoulou	26
Muhammad Umer Khan	27
Lars Schmidt- Thieme	28
Alexandros Nanopoulos	29
Katerina Pechlivanidou	30
Kostas Katsalis	31
Ioannis Igoumenos	32
Thanasis Korakis	33
Nikos Makris	34
Nikos Dimokas	35
Alfredo Cuzzocrea	36
Alexis Papadimitriou	37
Leonidas Akritidis	38
Panayiotis Bozanis	39
Donatos Stavropoulos	40
Giannis Kazdaridis	41
Vasilis Surlas	42
Paris Flegkas	43
Georgios S. Paschos	44

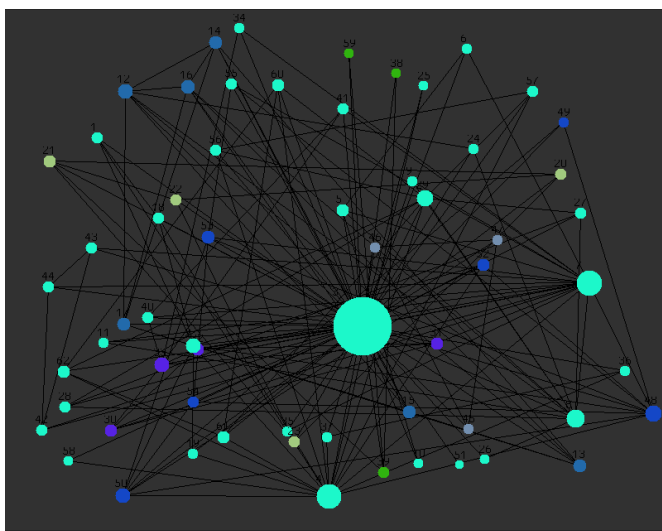
Author's Name	Name
Bhaskar Prasad Rimal	45
Admela Jukan	46
Yves Goeleven	47
George Pallis	48
S. Sivasubramanian	49
Athena Vakali	50
Andreas Papadopoulos	51
Konstantinos Stamos	52
Marios D. Dikaiakos	53
Pankaj Mehra	54
Nicholas Loulloudes	55
Ioannis Karydis	56
Apostolos N. Papadopoulos	57
Maria Kontaki	58
Fotis Tsakiridis	59
Gökhan Yavas	60
Özgür Ulusoy	61
Murat Karakaya	62

Μετά τη χρήση του λογισμικού AviNet υπολογίστηκαν οι παρακάτω κοινότητες για  $s=1$  και μετά έγιναν δοκιμές για διαφορετικές τιμές του  $s$ .

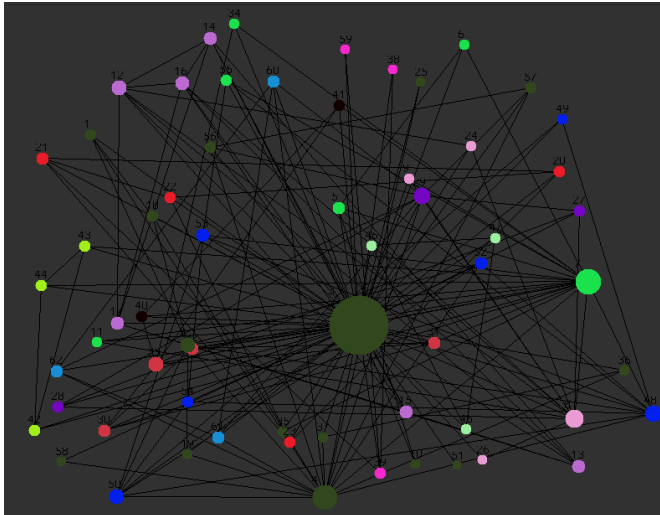



 $S = 1$ 

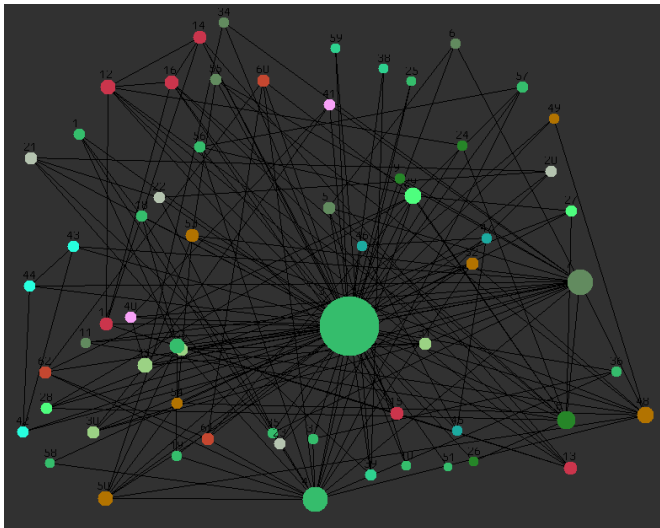
community	nodes
1	13, 12, 14, 15, 16, 17
2	20, 21, 22, 23
3	45, 46, 47, 59, 38, 39, 54, 53, 52, 49, 48, 50, 30, 31, 32, 33, 42, 43, 44, 40, 41, 55, 34, 11, 6, 5, 7, 26, 24, 9, 8, 60, 61, 62, 27, 28, 29, 56, 57, 36, 37, 58, 51, 35, 25, 19, 10, 1, 2, 3, 4, 18


 $S = 2$ 

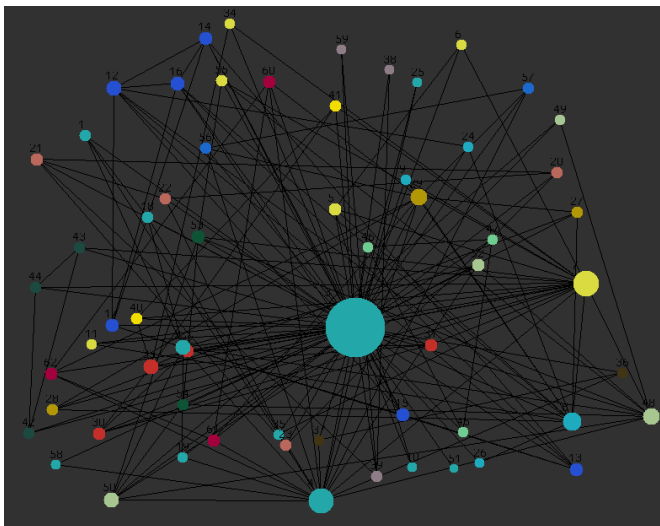
community	nodes
1	13, 12, 14, 15, 16, 17
2	20, 21, 22, 23
3	30, 31, 32, 33
4	42, 43, 44, 40, 41, 55, 34, 11, 6, 5, 7, 26, 24, 9, 8, 60, 61, 62, 27, 28, 29, 56, 57, 36, 37, 58, 51, 35, 25, 19, 10, 1, 2, 3, 4, 18
5	45, 46, 47
6	54, 53, 52, 49, 48, 50
7	59, 38, 39


 $S = 3$ 

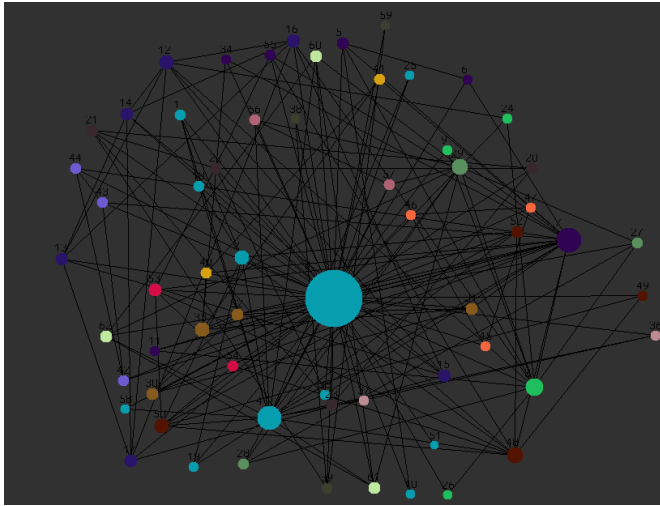
community	nodes
1	13, 12, 14, 15, 16, 17
2	20, 21, 22, 23
3	26, 24, 9, 8
4	27, 28, 29
5	30, 31, 32, 33
6	40, 41
7	42, 43, 44
8	45, 46, 47
9	54, 53, 52, 49, 48, 50
10	55, 34, 11, 6, 5, 7
11	56, 57, 36, 37, 58, 51, 35, 25, 19, 10, 1, 2, 3, 4, 18
12	59, 38, 39
13	60, 61, 62


 $S = 4$ 

community	nodes
1	13, 12, 14, 15, 16, 17
2	20, 21, 22, 23
3	26, 24, 9, 8
4	27, 28, 29
5	30, 31, 32, 33
6	40, 41
7	42, 43, 44
8	45, 46, 47
9	54, 53, 52, 49, 48, 50
10	55, 34, 11, 6, 5, 7
11	56, 57, 36, 37, 58, 51, 35, 25, 19, 10, 1, 2, 3, 4, 18
12	59, 38, 39
13	60, 61, 62

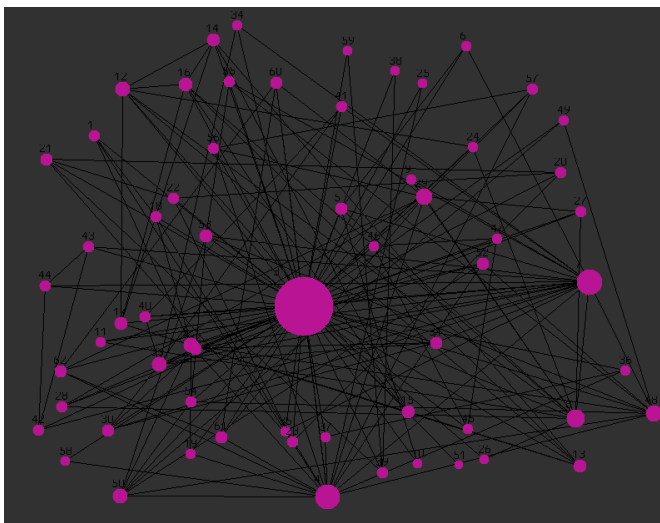

 $S = 5$ 

community	nodes
1	13, 12, 14, 15, 16, 17
2	20, 21, 22, 23
3	26, 24, 9, 8
4	27, 28, 29
5	30, 31, 32, 33
6	36, 37
7	40, 41
8	42, 43, 44
9	45, 46, 47
10	52, 49, 48, 50
11	54, 53
12	55, 34, 11, 6, 5, 7
13	56, 57
14	58, 51, 35, 25, 19, 10, 1, 2, 3, 4, 18
15	59, 38, 39
16	60, 61, 62



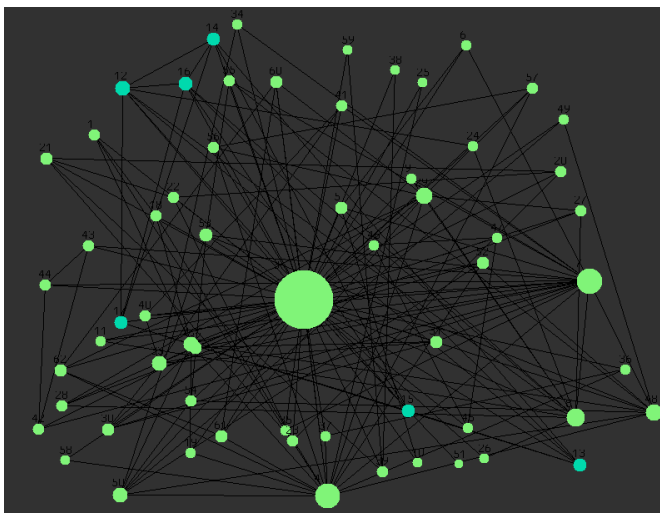
$$S = 10$$

community	nodes
1	13, 12, 14, 15, 16, 17
2	20, 21, 22, 23
3	26, 24, 9, 8
4	27, 28, 29
5	30, 31, 32, 33
6	36, 37
7	40, 41
8	42, 43, 44
9	45, 46, 47
10	52, 49, 48, 50
11	54, 53
12	55, 34, 11, 6, 5, 7
13	56, 57
14	58, 51, 35, 25, 19, 10, 1, 2, 3, 4, 18
15	59, 38, 39
16	60, 61, 62



$$S = 0.5$$

community	nodes
1	13, 12, 14, 15, 16, 17, 20, 21, 22, 23, 45, 46, 47, 59, 38, 39, 54, 53, 52, 49, 48, 50, 30, 31, 32, 33, 42, 43, 44, 40, 41, 55, 34, 11, 6, 5, 7, 26, 24, 9, 8, 60, 61, 62, 27, 28, 29, 56, 57, 36, 37, 58, 51, 35, 25, 19, 10, 1, 2, 3, 4, 18



$$S = 0.85$$

community	nodes
1	13, 12, 14, 15, 16, 17
2	20, 21, 22, 23, 45, 46, 47, 59, 38, 39, 54, 53, 52, 49, 48, 50, 30, 31, 32, 32, 33, 42, 43, 44, 40, 41, 55, 34, 11, 6, 5, 7, 26, 24, 9, 8, 60, 61, 62, 27, 28, 29, 56, 57, 36, 37, 58, 51, 35, 25, 19, 10, 1, 2, 3, 4, 18

Παρατηρούμε ότι όσο μεγαλώνει το  $s$  τόσο περισσότερες κοινότητες δημιουργούνται ενώ όσο μικρότερο γίνεται το  $s$  τόσο λιγότερες γίνονται οι κοινότητες. Στο δικό μας δίκτυο για  $s=0.5$  παρατηρούμε

οτι δημιουργείται μόνο μια κοινότητα. Ενώ μετά το  $s=5$  οι κοινότητες σταθεροποιούνται στις 16.