

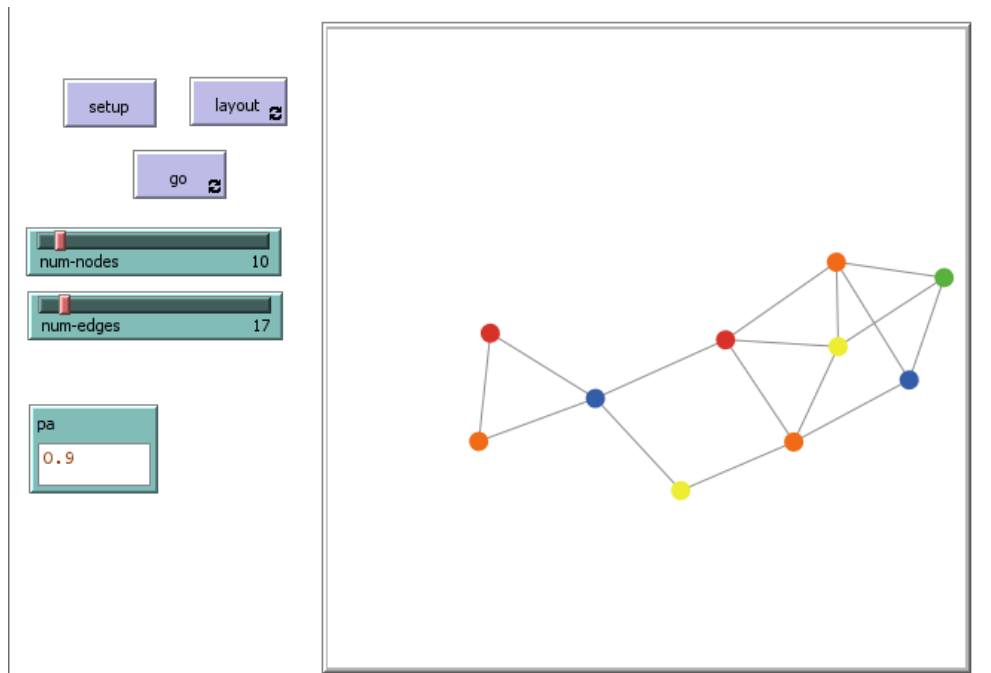
Προγραμματιστικές Ασκήσεις

Άσκηση 1:

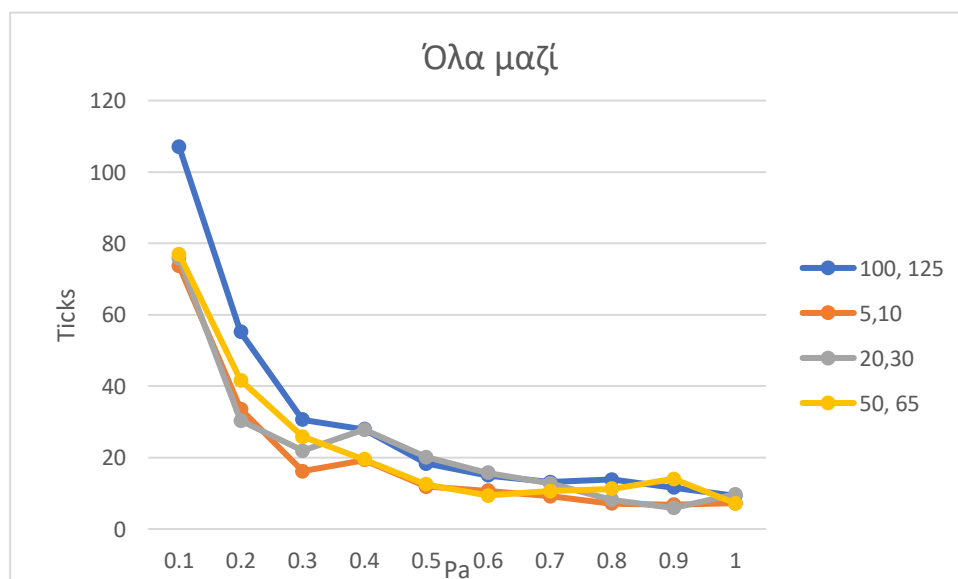
Ο κώδικας αποτελείται από την συνάρτηση:

- setup, αρχικοποιεί την προσομοίωση.
- layout, που κάνει πιο αναγνώσιμο το γράφημα.
- Την go, που «προχωράει» την προσομοίωση κατά ένα βήμα.

Πχ.



Αποτελέσματα:



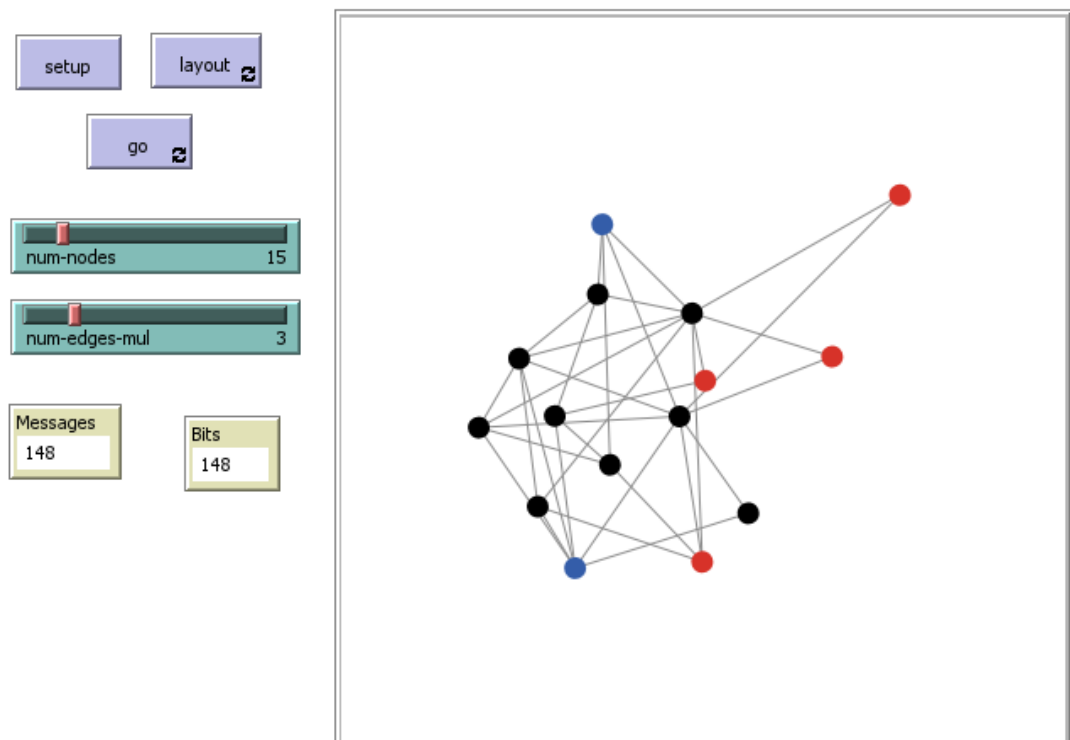
Όπως ήταν αναμενόμενο, παρατηρούμε ότι όσο αυξάνεται η πιθανότητα P_a , τόσο λιγότερος χρόνος απαιτείται για να τερματίσει ο αλγόριθμος. Αυτό οφείλεται στο ότι κάθε κόμβος έχει μεγαλύτερες πιθανότητες σε κάθε βήμα να είναι ένα αποδεκτό χρώμα.

Στο υπόμνημα του διαγράμματος, βλέπουμε πως κάθε γραφική χαρακτηρίζεται από δυο αριθμούς. Ο πρώτος είναι ο αριθμός των κόμβων και ο δεύτερος ο αριθμός των ακμών.

Άσκηση 2:

Luby's MIS Distributed Algorithm:

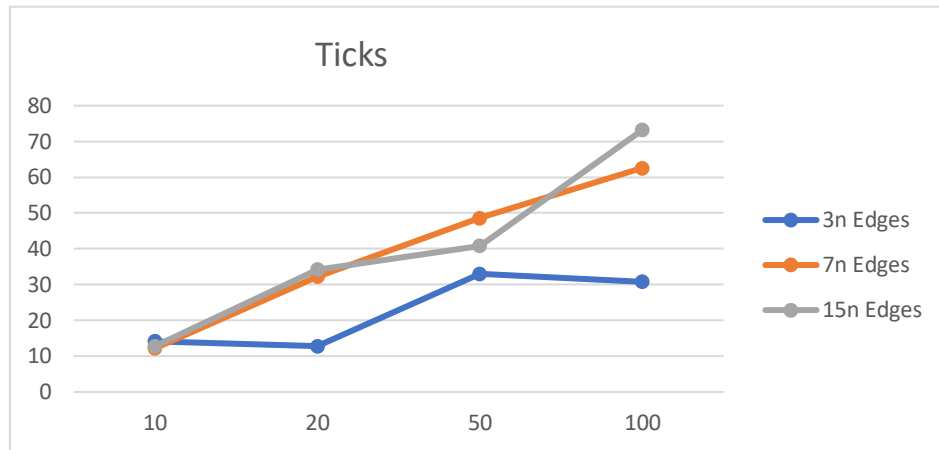
Το περιβάλλον προσομοίωσης είναι πανομοιότυπο με την πρώτη άσκηση. Η μόνη διαφορά είναι τα monitors προβολής των μηνυμάτων και των bits. Κάθε φορά που



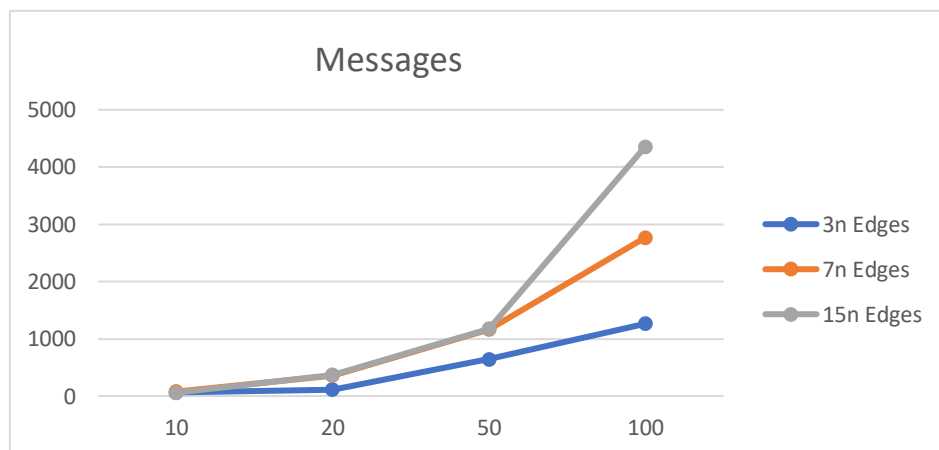
ένας κόμβος σε κάθε βήμα γίνεται active, «στέλνει» στους γειτονικούς του κόμβους μήνυμα «1» για να μην ενεργοποιηθούν. Σε αυτόν τον αλγόριθμο έχω βάλει με μπλε χρώμα να εμφανίζονται οι κόμβοι που μπήκαν στον MIS μετά από σύγκριση του βαθμού τους με τους γειτονικούς active κόμβους.

Αποτελέσματα:

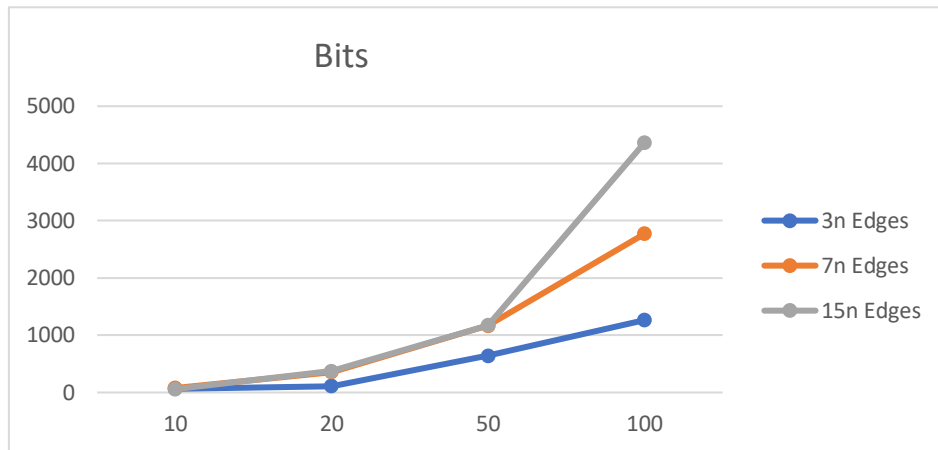
Αποτελέσματα χρόνο ολοκλήρωσης (n ο αριθμός των κόμβων):



Αποτελέσματα μηνυμάτων που στάλθηκαν σε γειτονικούς κόμβους:



Αποτελέσματα bits που στάλθηκαν μέσω των μηνυμάτων:

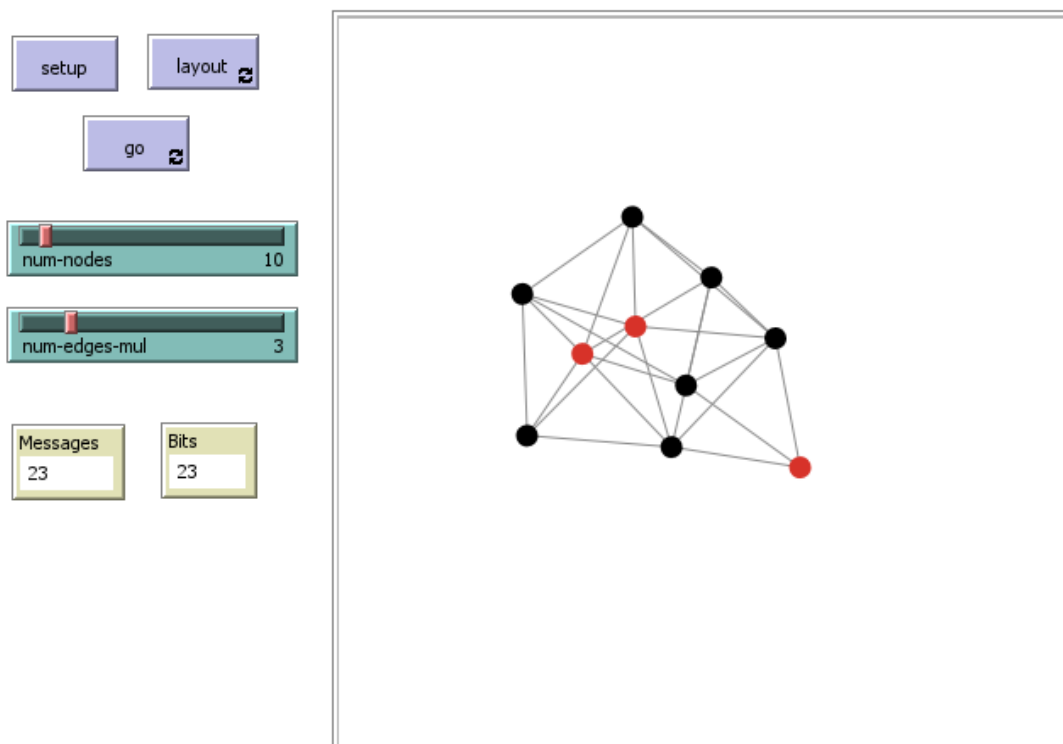


Παρατηρούμε ότι ο χρόνος εκτέλεσης είναι ανάλογος του αριθμού των κόμβων και των ακμών του γραφήματος.

Επίσης παρατηρούμε πως ανάλογα αυξάνεται και ο αριθμός των μηνυμάτων. Ιδιαίτερα στα πυκνά γραφήματα, κάθε κόμβος «στέλνει » πολλά μηνύματα, αφού αυξάνεται ο αριθμός των γειτονικών κόμβων.

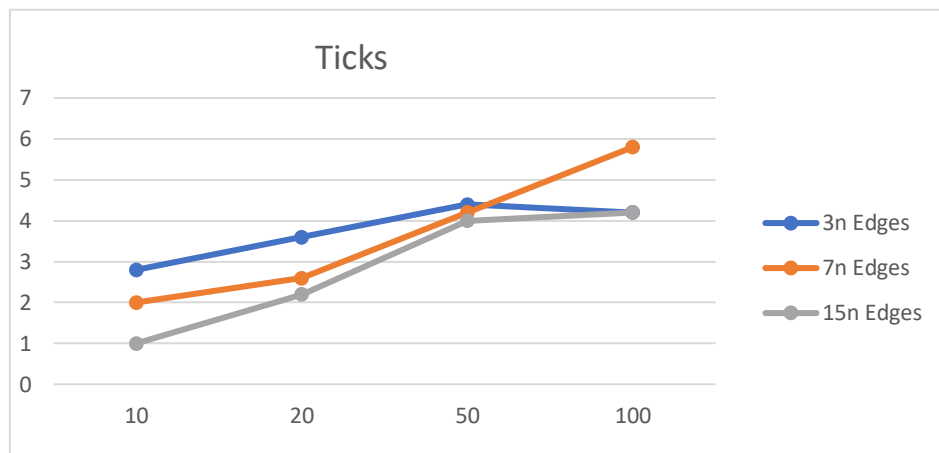
Τα bits είναι ανάλογα των μηνυμάτων και για αυτό συμπεριφέρονται ανάλογα.

Distributed Biologically-Inspired Synchronous Algorithm for MIS improved (σελ 31):

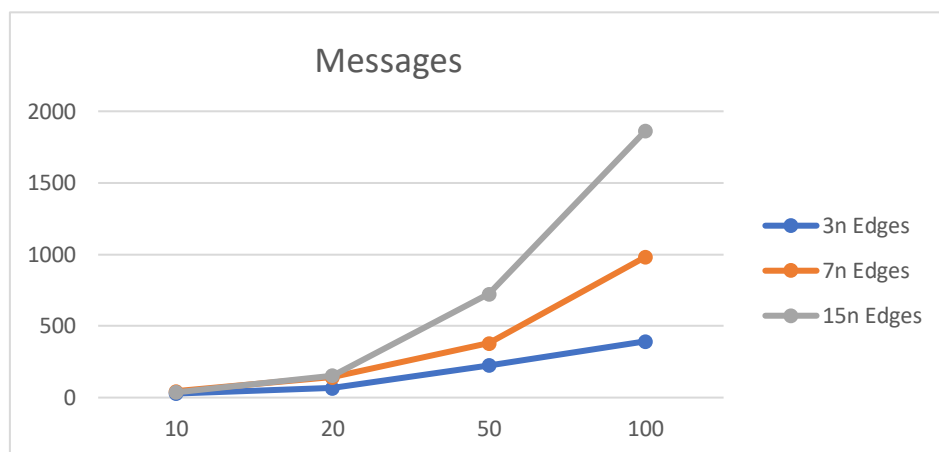


Αποτελέσματα:

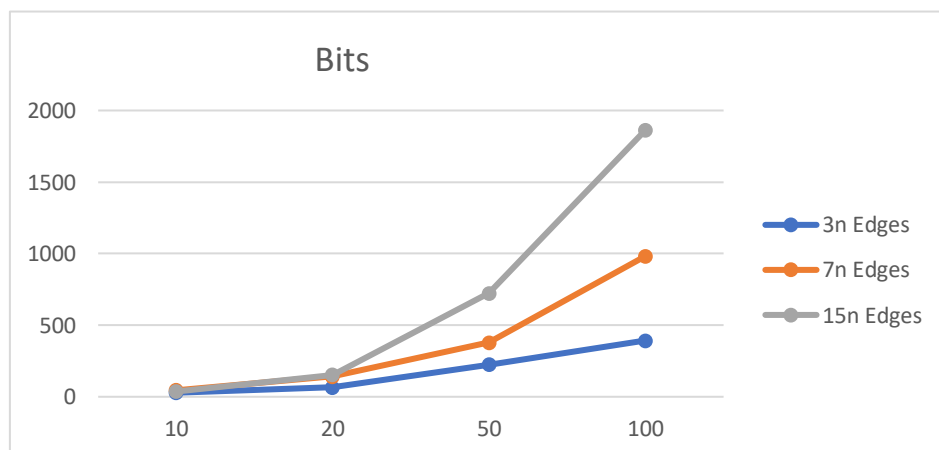
Αποτελέσματα χρόνο ολοκλήρωσης (n ο αριθμός των κόμβων):



Αποτελέσματα μηνυμάτων που στάλθηκαν σε γειτονικούς κόμβους:



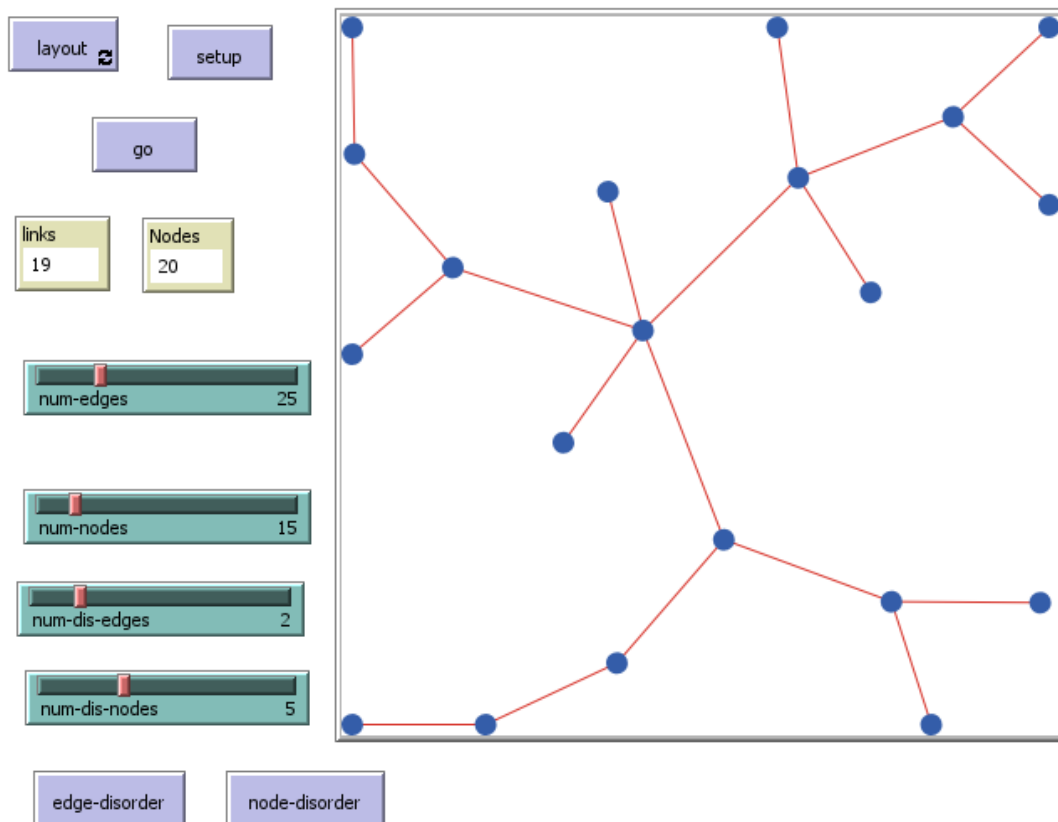
Αποτελέσματα bits που στάλθηκαν μέσω των μηνυμάτων:



Παρατηρούμε ότι και αυτός ο αλγόριθμος έχει παρόμοιες συμπεριφορές, αλλά σε αντίθεση με τον προηγούμενο είναι πολύ πιο γρήγορος και απαιτεί πολύ μικρότερο αριθμό μηνυμάτων και bits. Επομένως είναι εμφανές πως είναι καλύτερος.

Άσκηση 3:

Το περιβάλλον βασίζεται στο ίδιο με τα προηγούμενα, αλλά εδώ εμφανίζονται και 2 κουμπιά που χρησιμοποιούνται για να πραγματοποιηθεί διαταραχή στον αριθμό των κόμβων ή των ακμών του γραφήματος, καθώς και τα 2 sliders που ελέγχουν πόσο μεγάλη θα είναι η διαταραχή αυτή:



Αποτελέσματα:

Δεν κατάφερα να βγάλω αποτελέσματα, διότι για κάποιον λόγο η προσομοίωση δεν έκανε «stop». Αλλά προβλέπω πως όσο αυξάνεται ο αριθμός των κόμβων και της διαταραχής, τόσο θα αυξάνεται και ο χρόνος του Self-stabilization.