Άσκηση 2 στο μάθημα Ρομποτικά Συστήματα

Θέμα: Κάλυψη περιοχής από κινούμενα ρομπότ

Ραπτάκης Αθανάσιος – 6623   Κακούρος Νικόλαος - 6519

1. Σκοπός της άσκησης

Σε εφαρμογές στις οποίες περιλαμβάνεται η ανάπτυξη ενός ρομποτικού συστήματος στον χώρο προκύπτει η ανάγκη για βελτιστοποίηση κάποιων παραμέτρων. Ως "ρομποτικό σύστημα", βέβαια, εννοούμε και οτιδήποτε μπορεί να μοντελοποιηθεί ως τέτοιο, πχ η συμπεριφορά κάποιων ζώων που ζουν σε αγέλες. Παραδείγματα παραμέτρων προς βελτιστοποίηση είναι η συνάντηση, τελικά, των επιμέρους μονάδων που αποτελούν το ρομποτικό σύστημα και η συντονισμένη κίνηση τους έπειτα (όπως σε σμήνη αποδημητικών πουλιών) ή η αυτόνομη κίνηση των μονάδων στον χώρο αλλά με αποφυγή της μεταξύ τους σύγκρουσης.
Στην προκειμένη, έχοντας τέσσερα robot εξετάζουμε την συνεργατική τους κίνηση ώστε να επιτευχθεί το μεγαλύτερο δυνατό ποσοστό κάλυψης μιας επιφάνειας. Η "κάλυψη" εδώ ορίζεται αφηρημένα και μπορεί να είναι σε μια πραγματική εφαρμογή η συνολική περιοχή που καλύπτουν οι αισθητήρες των robot (πχ κάμερες, μικρόφωνα, κλπ)

2. Παραδοχές

Στο ιδανικό σενάριο της άσκησης κάναμε τις παρακάτω παραδοχές.

1. Τα robot είναι ιδεατά και χωρίς δυναμική. Γιαυτό εμφανίζονται ως τελείες στο χώρο. Επιπλέον, με αυτήν την υπόθεση οι αρχικές γωνίες προσανατολισμού δεν χρησιμοποιούνται αφού τα robot μπορούν και κινούνται ως προς οποιαδήποτε κατεύθυνση κάθε φορά. Με άλλα λόγια, τα robot που θεωρούμε είναι ολόνομα.
2. Τα robot μπορούν να ανιχνεύουν με την απαιτούμενη ακρίβεια τη θέση τους στο χώρο.
3. Τα robot μπορούν να ανταλλάζουν αυτόματα και άμεσα τις πληροφορίες για τη θέση τους μεταξύ τους, χωρίς κανέναν περιορισμό.
4. Τα robot έχουν αρκετή ενέργεια ώστε να εκτελέσουν όλη την κίνηση που προβλέπει ο αλγόριθμος.

3. Ο αλγόριθμος

Από τους διάφορους αλγορίθμους που υπάρχουν για την βελτιστοποίηση της κάλυψης επιλέξαμε αυτόν που κάνει χρήση r-limited voronoi cells. Ο λόγος είναι η ευκολότερη κατανόηση της λειτουργίας του και η πολυπλοκότητα της υλοποίησης του. Τα βήματα του αλγορίθμου αυτού, μαζί με τα αρχεία matlab που τα υλοποιούν είναι:

1. Ορισμός αρχικών παραμέτρων (conf.m)
2. Υπολογισμός voronoi κελιών (calc_voronoi_.m)
3. Υπολογισμός των κέντρων των voronoi κελιών (voronoi_centers.m)
4. Κίνηση των robot προς το κέντρο του voronoi κελιού τους (move.m)

5. Έλεγχος τερματισμού (main.m)

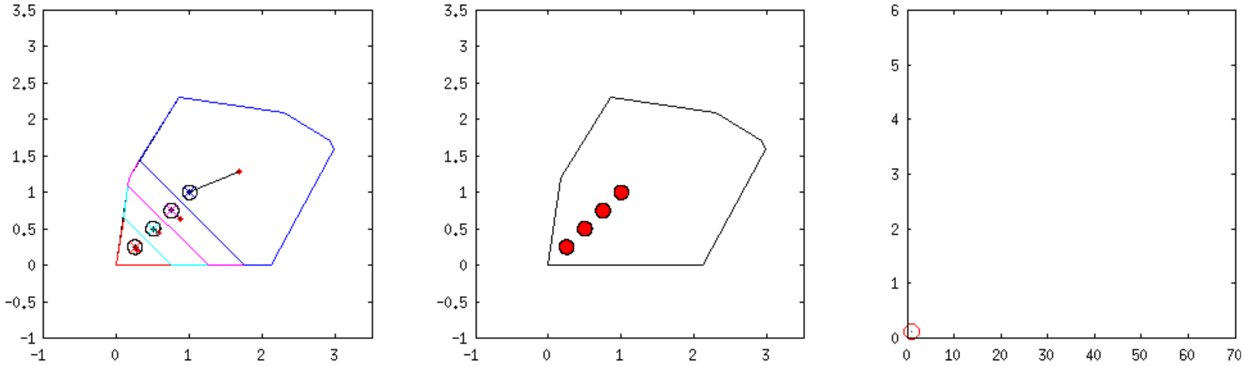
Το τελευταίο βήμα, όπως υλοποιήθηκε, ελέγχει αν τα robot έχουν αλλάξει θέση από την προηγούμενη επανάληψη, αν η ακτίνα δράσης ενός robot τέμνει της ακτίνα δράσης ενός άλλου ή τα όρια του χώρου και αν, μετά από αρκετές επαναλήψεις, η απόσταση που μετακινούνται τα robot είναι μεγαλύτερη από το ορισμένο βήμα. Αν κάτι από όλα αυτά δεν ισχύει, ο αλγόριθμος τερματίζει έχοντας υπολογίσει την βέλτιστη ανάπτυξη των robot για την μέγιστη κάλυψη.

Πρέπει να σημειωθεί πως ο αλγόριθμος κάλυψης, όπως υλοποιήθηκε, δεν είναι βέλτιστος, υπό την έννοια ότι υπάρχουν και διαφορετικές υλοποιήσεις που συγκλίνουν και πιο γρήγορα και καταλήγουν σε τελική διάταξη των robot που δίνει μεγαλύτερη κάλυψη της επιφάνειας. Επιπλέον, ο αλγόριθμος που βασίζεται στα voronoi δεν αποδεικνύεται μαθηματικά ότι συκλίνει.

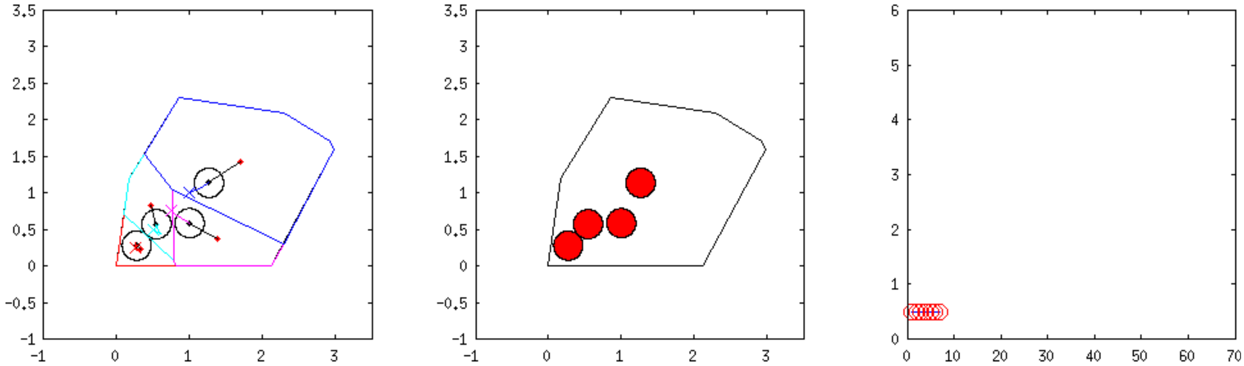
** Σχολιασμός για την ακριβή λειτουργία του κώδικα υπάρχει στα m αρχεία.*
*** Ο συνολικός χρόνος εκτέλεσης του αλγορίθμου είναι περίπου μισή ώρα.*

4. Αποτελέσματα

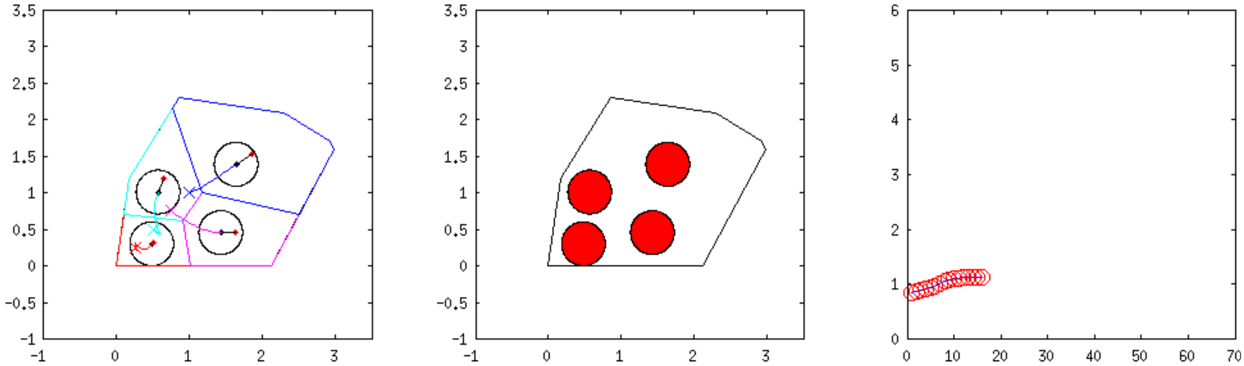
Παρακάτω παρουσιάζονται συνοπτικά και εποπτικά τα αποτελέσματα του αλγορίθμου για τις 10 διαφορετικές ακτίνες δράσης. Για κάθε R, στην αριστερή γραφική δίνεται η κάτοψη του επιπέδου με την τελική θέση των robot και το ίχνος της τροχιάς τους, στη μεσαία το τμήμα του επιπέδου που τελικά καλύφθηκε (με κόκκινο χρώμα) και, τέλος, στην δεξιά γραφική, μια γραφική παράσταση που δείχνει το εμβαδόν που καλύπτεται σε κάθε επανάληψη του αλγορίθμου.



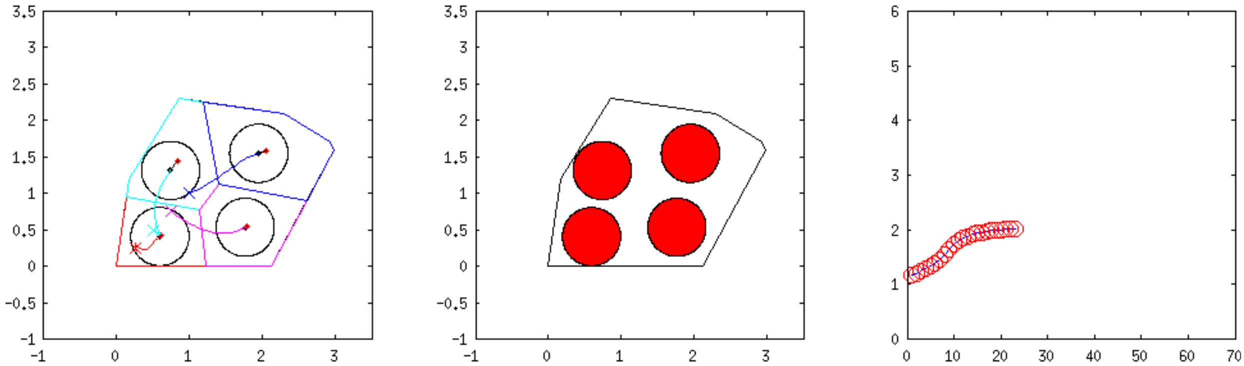
Εικόνα 1: Αποτελέσματα για r=0.1



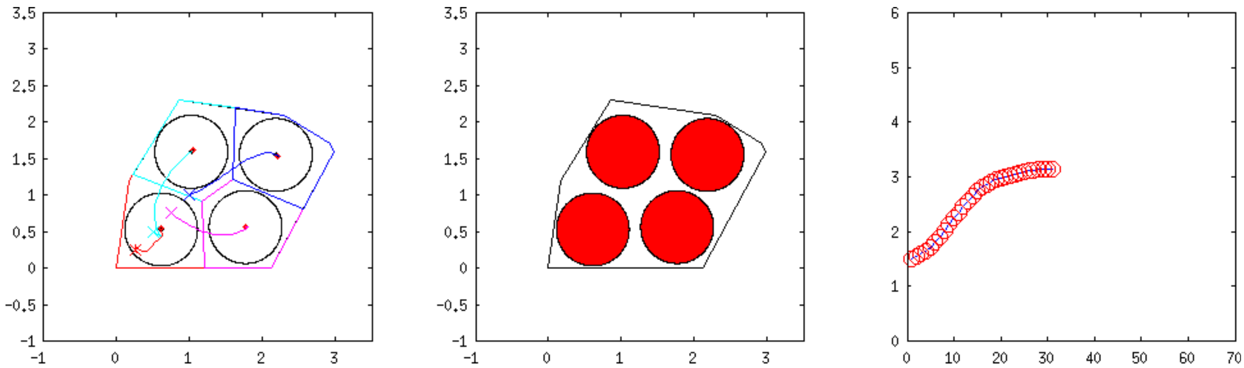
Εικόνα 2: Αποτελέσματα για r=0.2



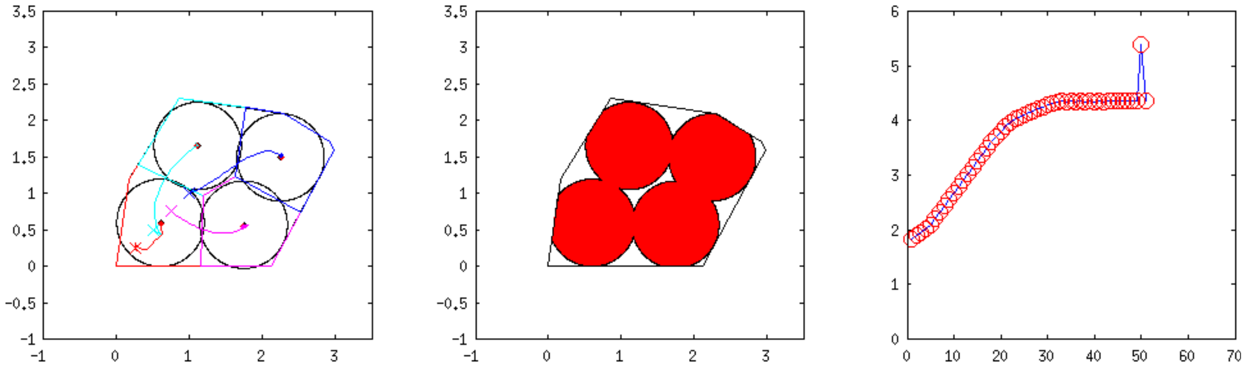
Εικόνα 3: Αποτελέσματα για r=0.3



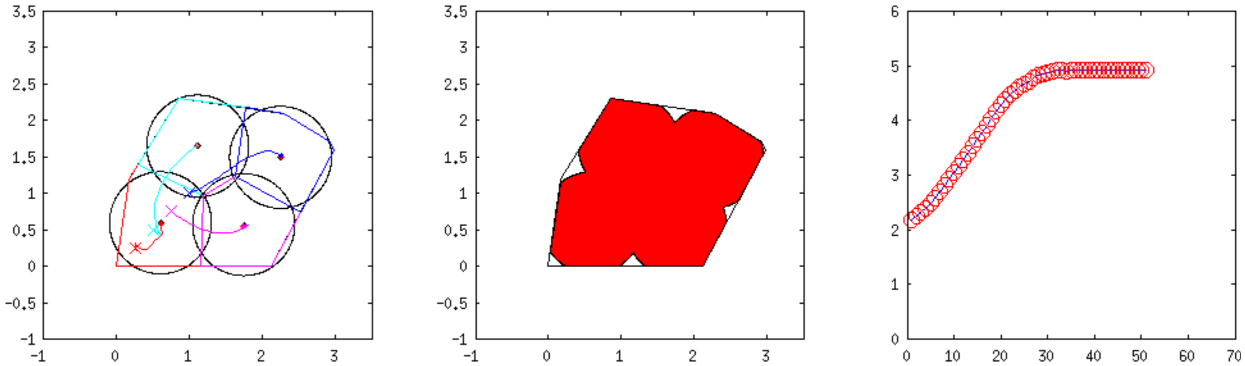
Εικόνα 4: Αποτελέσματα για r=0.4



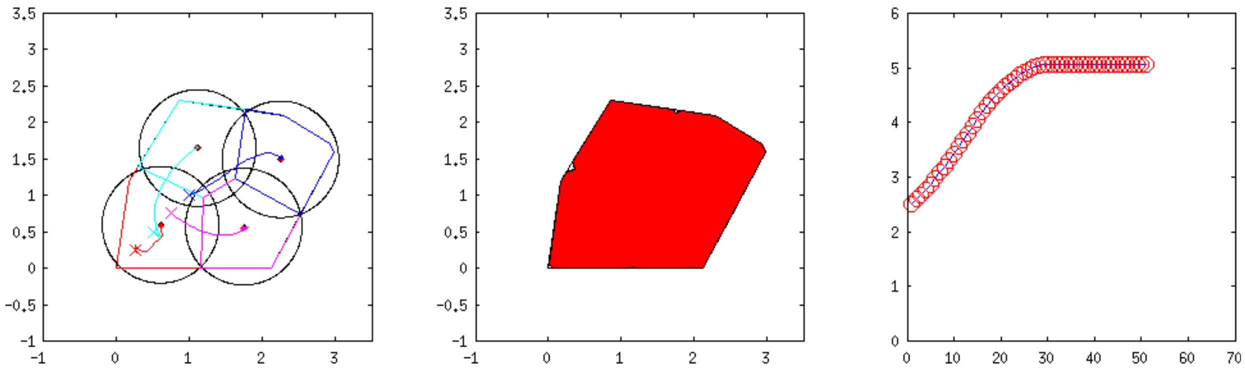
Εικόνα 5: Αποτελέσματα για r=0.5



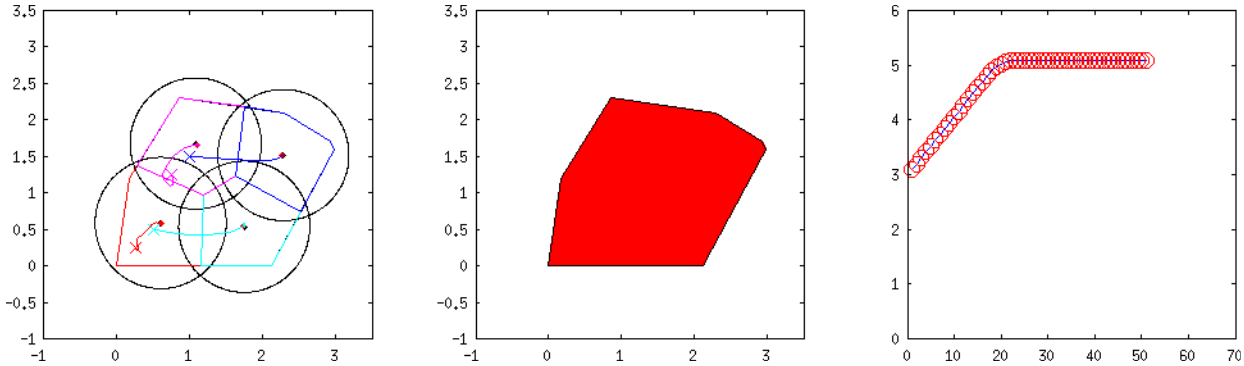
Εικόνα 6: Αποτελέσματα για r=0.6



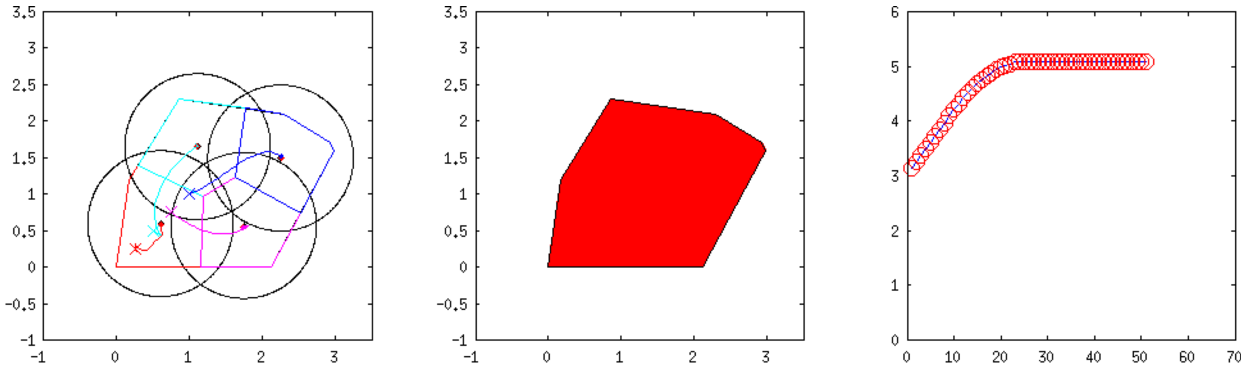
Εικόνα 7: Αποτελέσματα για r=0.7



Εικόνα 8: Αποτελέσματα για r=0.8



Εικόνα 9: Αποτελέσματα για r=0.9



Εικόνα 10: Αποτελέσματα για r=1

** Στην δεξιά εικόνα 6 μια υπερβολική γραμμή που φαίνεται είναι λάθος.*
*** Το εμβαδόν αυτής καθεαυτής της επιφάνειας είναι 5.0809*

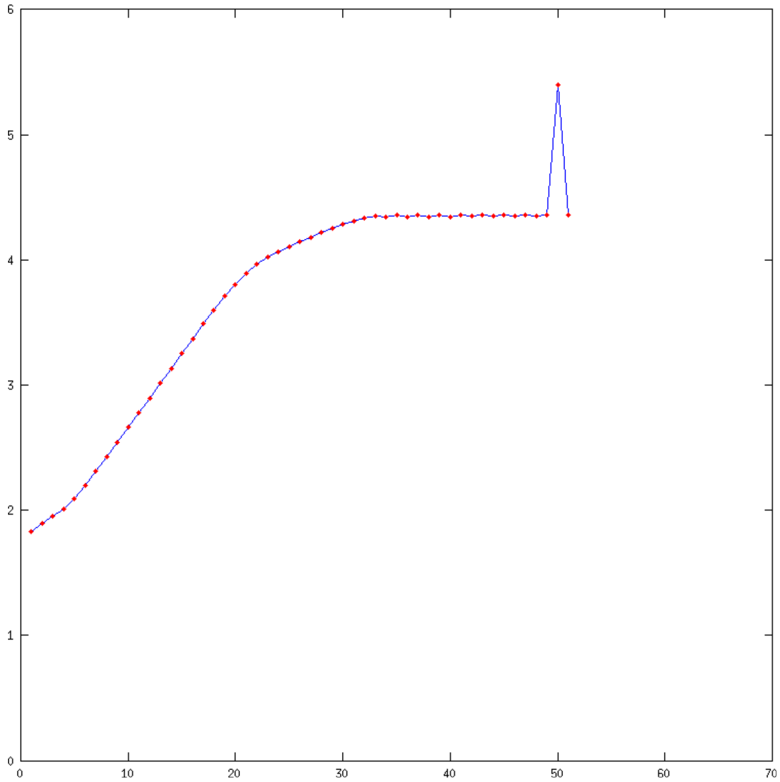
Γενικά σχόλια

Παρατηρούμε τα εξής:

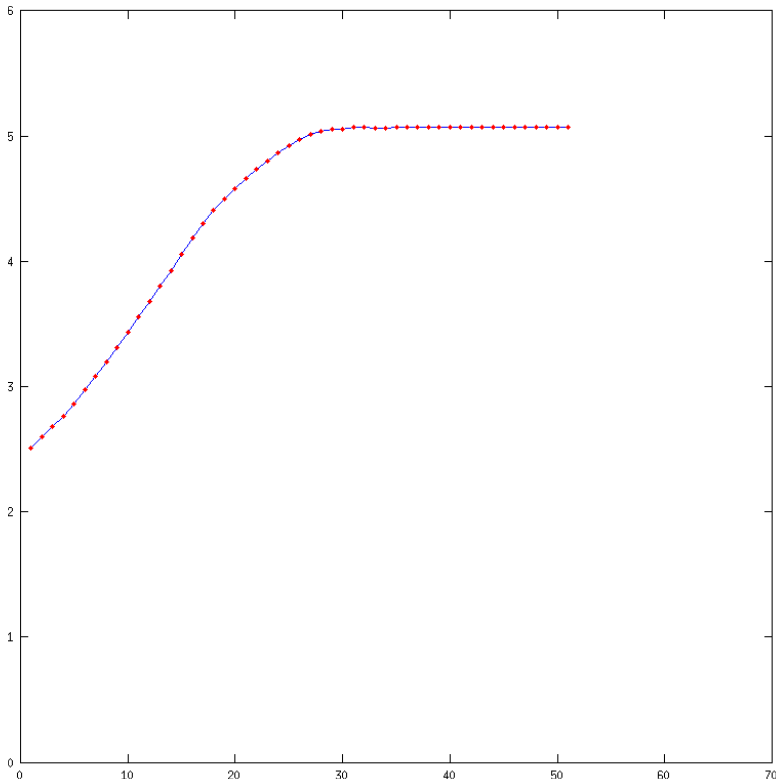
- Η τροχιά των robot εξαρτάται βεβαίως μόνο από την κατασκευή του διαγράμματος voronoi , η οποία είναι ανεξάρτητη των ακτίνων δράσης. Έτσι, και στα 10 παραπάνω σχήματα, οι τροχιές που φαίνονται κάθε φορά είναι τμήματα της ίδιας γενικής τροχιάς.
- Όσο μεγαλώνει η ακτίνα και για R>0.5 βλέπουμε πως ο διαμερισμός του επιπέδου σε voronoi κελιά είναι αρκετά καλός, αφού τα τέσσερα κελιά έχουν,

περίπου το ίδιο εμβαδόν (ιδανικά το ίδιο).

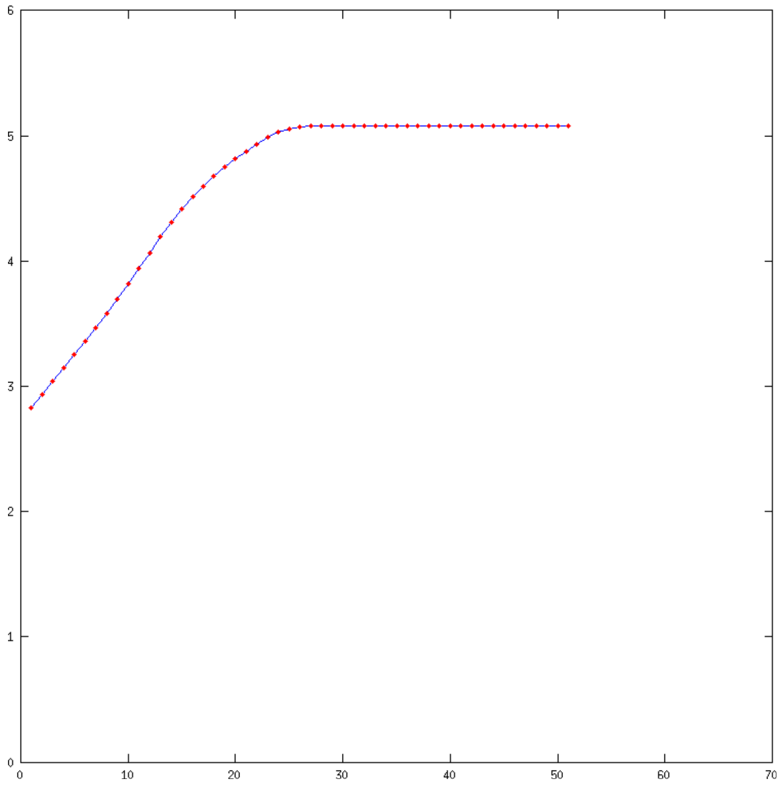
- Όσο μεγαλώνει η ακτίνα δράσης τόσο μεγαλύτερο ποσοστό καλύπτεται τελικά, κάτι αναμενόμενο. Παρατηρώντας τον κάθετο άξονα των δεξιών γραφημάτων βλέπουμε ότι η αύξηση του τελικού καλυπτομένου εμβαδού καθώς το r αυξάνει είναι σχεδόν εκθετική (*εμβαδόν κύκλου=2πr^2).* Αυτό, όμως, μέχρι r=0.8 (πρακτικά) αφού μετά, οποιαδήποτε αύξηση του εμβαδού είναι περιττή.
- Μέχρι r=0.5, η συνθήκη τερματισμού του αλγορίθμου είναι η μη αλληλοεπικάλυψη των ακτίνων δράσης (και των περιθωρίων του χώρου). Επιλέχτηκε ένα τέτοιο κριτήριο, ώστε τα robot να μην κάνουν περιττές κινήσεις, για οικονομία.
- Κρίνοντας από τα δεξιά γραφήματα, η ταχύτητα σύγκλισης του αλγορίθμου είναι σχεδόν γραμμική (στα πρώτα βήματα), κάτι αναμενόμενο λόγω του r-limted voronoi αλγόριθμου.
- Για r>0.5, βλέπουμε ότι ο αλγόριθμος εκτελεί, από ένα σημείο και μετά περιττές επαναλήψεις, αφού το καλυπτόμενο ποσοστό της επιφάνειας παραμένει το ίδιο. Αυτό έγινε σκόπιμα για να δειχτεί ότι ο αλγόριθμος είναι ευσταθής. Για οικονομία, θα μπορούσαμε να τερματίσουμε τον αλγόριθμο αμέσως μετά αφότου το εμβαδόν κάλυψης σε μία επανάληψη γίνει μικρότερο από ότι στην προηγούμενη επανάληψη.
- Ο αλγόριθμος παράγει μια αύξουσα συνάρτηση για το εμβαδόν κάλυψης ως προς τον αριθμό των επαναλήψεων, δηλαδή δεν κάνει ενδιάμεσα ταλαντώσεις. Αυτό ενισχύει την ευστάθεια του αλγορίθμου αλλά και την οικονομία του.
- Το βήμα κίνησης των robot επιλέχθηκε μετά από δοκιμές αρκετά μικρό ώστε ο αλγόριθμος να είναι αρκετά ευσταθής και να συγκλίνει σε ένα πολύ καλό ποσοστό κάλυψης, αλλά και αρκετά μεγάλο ώστε ο αλγόριθμος να συγκλίνει σχετικά γρήγορα.
- Για μεγαλύτερο βήμα κίνησης, για μικρά r (έως 0.3) ο αλγόριθμος ωφελείται και τρέχει πολύ πιο γρήγορα συγκλίνοντας, μάλιστα, στο μέγιστο εμβαδόν κάλυψης. Εντούτοις, για μεγαλύτερα r, ο αλγόριθμος γίνεται εξαιρετικά ασταθής και ταλαντώνεται συνεχώς, με το τελικό αποτέλεσμα να είναι "τυχαίο".
- Για την ακρίβεια, για r>5, μετά από αρκετές επαναλήψεις, ο αλγόριθμος δεν τοποθετεί τα robot σε μια τελική θέση, αλλά κάνει μια πολύ μικρή ταλάντωση ανάμεσα σε 2 πολύ κοντινές θέσεις για κάθε robot. Παρόλα αυτά, το εμβαδόν κάλυψης, για r>0.7, δεν κάνει κάποια ταλάντωση και σταθεροποιείται ενώ για r=0.6, 0.7 κάνει μια πολύ μικρή ταλάντωση (βλ. παρακάτω σχήματα). Για να εξαλειφθεί ακόμα και αυτή, θα έπρεπε να χρησιμοποιήσουμε ακόμα μικρότερο βήμα, που, όμως θα συντελούσε σε μεγάλη αύξηση του χρόνου εκτέλεσης.



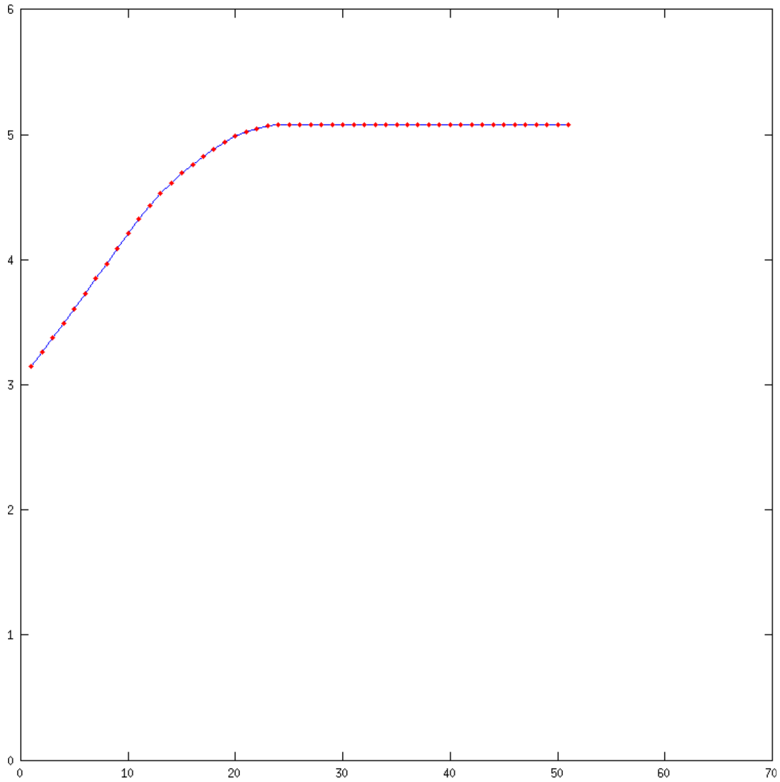
Εικόνα 11: Εμβαδόν κάλυψης συναρτήσει αριθμού επαναλήψεων για r=0.6



Εικόνα 13: Εμβαδόν κάλυψης συναρτήσει αριθμού επαναλήψεων για r=0.8



Εικόνα 14: Εμβαδόν κάλυψης συναρτήσει αριθμού επαναλήψεων για r=0.9



Εικόνα 15: Εμβαδόν κάλυψης συναρτήσει αριθμού επαναλήψεων για r=1