



Τ.Ε.Ι ΠΕΙΡΑΙΑ

ΝΟΗΜΟΣΥΝΗ ΣΜΗΝΟΥΣ ΜΕΛΕΤΗ ΑΞΙΟΛΟΓΗΣΗ & ΥΛΟΠΟΙΗΣΗ ΑΛΓΟΡΙΘΜΩΝ ΓΙΑ ΤΟΝ ΕΛΕΓΧΟ ΜΗ ΕΠΑΝΔΡΩΜΕΝΩΝ ΟΧΗΜΑΤΩΝ(UAVs)

Φοιτητές: ΚΑΨΟΥΛΗΣ ΙΩΣΗΦ

ΜΑΝΩΛΟΠΟΥΛΟΣ ΧΡΗΣΤΟΣ

ΜΕΛΚΩΝΗΣ ΑΝΤΩΝΗΣ

Επιβλέπων: ΠΑΠΟΥΤΣΙΔΑΚΗΣ ΜΙΧΑΗΛ

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

ΣΧΟΛΗ ΤΕΧΝΟΛΟΓΙΚΩΝ ΕΦΑΡΜΟΓΩΝ

ΤΜΗΜΑ ΑΥΤΟΜΑΤΙΣΜΟΥ

ΙΟΥΝΙΟΣ 2016

“Η επιστήμη είναι ο πιο ειλικρινής φίλος του ανθρώπου
κι ο πιο ευγενικός του βοηθός”

Χάρβεϋ

ΔΗΛΩΣΗ ΣΥΓΓΡΑΦΕΑ ΠΤΥΧΙΑΚΗΣ ΕΡΓΑΣΙΑΣ

Ο/Η κάτωθι υπογεγραμμένος/η Μελκωνός Αντώνιος,
του Αιμίλιου, με αριθμό μητρώου 40832 φοιτητής / τριά του
Τμήματος **Μηχανικών Αυτοματισμού Τ.Ε.** του Α.Ε.Ι. Πειραιά Τ.Τ. πριν αναλάβω την
εκπόνηση της Πτυχιακής Εργασίας μου, δηλώνω ότι ενημερώθηκα για τα παρακάτω:

«Η Πτυχιακή Εργασία (Π.Ε.) αποτελεί προϊόν πνευματικής ιδιοκτησίας τόσο του
συγγραφέα, όσο και του Ιδρύματος και θα πρέπει να έχει μοναδικό χαρακτήρα και
πρωτότυπο περιεχόμενο.

Απαγορεύεται αυστηρά οποιοδήποτε κομμάτι κειμένου της να εμφανίζεται
αυτούσιο ή μεταφρασμένο από κάποια άλλη δημοσιευμένη πηγή. Κάθε τέτοια πράξη
αποτελεί προϊόν λογοκλοπής και εγείρει θέμα Ηθικής Τάξης για τα πνευματικά δικαιώματα
του άλλου συγγραφέα. Αποκλειστικός υπεύθυνος είναι ο συγγραφέας της Π.Ε., ο οποίος
φέρει και την ευθύνη των συνεπειών, ποινικών και άλλων, αυτής της πράξης.

Πέραν των όποιων ποινικών ευθυνών του συγγραφέα σε περίπτωση που το ίδρυμα
του έχει απονεμίσει Πτυχίο, αυτό ανακαλείται με απόφαση της Συνέλευσης του Τμήματος. Η
Συνέλευση του Τμήματος με νέα απόφασης της, μετά από αίτηση του ενδιαφερόμενου, του
αναθέτει εκ νέου την εκπόνηση της Π.Ε. με άλλο θέμα και διαφορετικό επιβλέποντα
καθηγητή. Η εκπόνηση της εν λόγω Π.Ε. πρέπει να ολοκληρωθεί εντός τουλάχιστον ενός
ημερολογιακού βμήνου από την ημερομηνία ανάθεσης της. Κατά τα λοιπά εφαρμόζονται τα
προβλεπόμενα στο άρθρο 18, παρ. 5 του ισχύοντος Εσωτερικού Κανονισμού.»

Επίσης δηλώνω υπεύθυνα ότι έχω παρακολουθήσει το σεμινάριο συγγραφής και
εκπόνησης πτυχιακής εργασίας που διοργανώνεται από το Τμήμα Μηχανικών
Αυτοματισμού Τ.Ε. κατά το Χειμερινό/Εαρινό Εξάμηνο του Ακ. Έτους 2015-2016

Ο Δηλών

Μελκωνός

Ημερομηνία

27-6-2016

ΔΗΛΩΣΗ ΣΥΓΓΡΑΦΕΑ ΠΤΥΧΙΑΚΗΣ ΕΡΓΑΣΙΑΣ

Ο/Η κάτωθι υπογεγραμμένος / η Μανωλόπουλος Χρήστος,
του Μικροβίου, με αριθμό μητρώου 39769 φοιτητής / τρια του
Τμήματος **Μηχανικών Αυτοματισμού Τ.Ε.** του Α.Ε.Ι. Πειραιά Τ.Τ. πριν αναλάβω την
εκπόνηση της Πτυχιακής Εργασίας μου, δηλώνω ότι ενημερώθηκα για τα παρακάτω:

«Η Πτυχιακή Εργασία (Π.Ε.) αποτελεί προϊόν πνευματικής ιδιοκτησίας τόσο του
συγγραφέα, όσο και του Ιδρύματος και θα πρέπει να έχει μοναδικό χαρακτήρα και
πρωτότυπο περιεχόμενο.

Απαγορεύεται αυστηρά οποιοδήποτε κομμάτι κειμένου της να εμφανίζεται
αυτούσιο ή μεταφρασμένο από κάποια άλλη δημοσιευμένη πηγή. Κάθε τέτοια πράξη
αποτελεί προϊόν λογοκλοπής και εγείρει θέμα Ηθικής Τάξης για τα πνευματικά δικαιώματα
του άλλου συγγραφέα. Αποκλειστικός υπεύθυνος είναι ο συγγραφέας της Π.Ε., ο οποίος
φέρει και την ευθύνη των συνεπειών, ποινικών και άλλων, αυτής της πράξης.

Πέραν των όποιων ποινικών ευθυνών του συγγραφέα σε περίπτωση που το Ίδρυμα
του έχει απονεμίσει Πτυχίο, αυτό ανακαλείται με απόφαση της Συνέλευσης του Τμήματος. Η
Συνέλευση του Τμήματος με νέα απόφασης της, μετά από αίτηση του ενδιαφερόμενου, του
αναθέτει εκ νέου την εκπόνηση της Π.Ε. με άλλο θέμα και διαφορετικό επιβλέποντα
καθηγητή. Η εκπόνηση της εν λόγω Π.Ε. πρέπει να ολοκληρωθεί εντός τουλάχιστον ενός
ημερολογιακού 6μήνου από την ημερομηνία ανάθεσης της. Κατά τα λοιπά εφαρμόζονται τα
προβλεπόμενα στο άρθρο 18, παρ. 5 του ισχύοντος Εσωτερικού Κανονισμού.»

Επίσης δηλώνω υπεύθυνα ότι έχω παρακολουθήσει το σεμινάριο συγγραφής και
εκπόνησης πτυχιακής εργασίας που διοργανώνεται από το Τμήμα Μηχανικών
Αυτοματισμού Τ.Ε. κατά το Χειμερινό/Εαρινό Εξάμηνο του Ακ. Έτους 2015-2016

Ο Δηλών



Χρήστος
Μανωλόπουλος

Ημερομηνία

27/6/2016

ΔΗΛΩΣΗ ΣΥΓΓΡΑΦΕΑ ΠΤΥΧΙΑΚΗΣ ΕΡΓΑΣΙΑΣ


Ο / Η κάτωθι υπογεγραμμένος / η Καψαλίας Γεώργιος,
του Μ.Κ.Ο.Λ.Α.Ο., με αριθμό μητρώου 40226..... φοιτητής / τρια του
Τμήματος **Μηχανικών Αυτοματισμού Τ.Ε.** του Α.Ε.Ι. Πειραιά Τ.Τ. πριν αναλάβω την
εκπόνηση της Πτυχιακής Εργασίας μου, δηλώνω ότι ενημερώθηκα για τα παρακάτω:

«Η Πτυχιακή Εργασία (Π.Ε.) αποτελεί προϊόν πνευματικής ιδιοκτησίας τόσο του
συγγραφέα, όσο και του Ιδρύματος και θα πρέπει να έχει μοναδικό χαρακτήρα και
πρωτότυπο περιεχόμενο.

Απαγορεύεται αυστηρά οποιοδήποτε κομμάτι κειμένου της να εμφανίζεται
αυτούσιο ή μεταφρασμένο από κάποια άλλη δημοσιευμένη πηγή. Κάθε τέτοια πράξη
αποτελεί προϊόν λογοκλοπής και εγείρει θέμα Ηθικής Τάξης για τα πνευματικά δικαιώματα
του άλλου συγγραφέα. Αποκλειστικός υπεύθυνος είναι ο συγγραφέας της Π.Ε., ο οποίος
φέρει και την ευθύνη των συνεπειών, ποινικών και άλλων, αυτής της πράξης.

Πέραν των όποιων ποινικών ευθυνών του συγγραφέα σε περίπτωση που το Ίδρυμα
του έχει απονείμει Πτυχίο, αυτό ανακαλείται με απόφαση της Συνέλευσης του Τμήματος. Η
Συνέλευση του Τμήματος με νέα απόφασης της, μετά από αίτηση του ενδιαφερόμενου, του
αναθέτει εκ νέου την εκπόνηση της Π.Ε. με άλλο θέμα και διαφορετικό επιβλέποντα
καθηγητή. Η εκπόνηση της εν λόγω Π.Ε. πρέπει να ολοκληρωθεί εντός τουλάχιστον ενός
ημερολογιακού βμήνου από την ημερομηνία ανάθεσης της. Κατά τα λοιπά εφαρμόζονται τα
προβλεπόμενα στο άρθρο 18, παρ. 5 του ισχύοντος Εσωτερικού Κανονισμού.»

Επίσης δηλώνω υπεύθυνα ότι έχω παρακολουθήσει το σεμινάριο συγγραφής και
εκπόνησης πτυχιακής εργασίας που διοργανώνεται από το Τμήμα Μηχανικών
Αυτοματισμού Τ.Ε. κατά το Χειμερινό/Εαρινό Εξάμηνο του Ακ. Έτους 2015-2016

Ο Δηλών
Γεώργιος Καψαλίας


Ημερομηνία
27/6/2016

1^Ο ΚΕΦΑΛΑΙΟ

| | |
|-------------------|---|
| 1. Περίληψη | 1 |
| 2. Εισαγωγή | 2 |

2^Ο ΚΕΦΑΛΑΙΟ

Βιβλιογραφική Ανασκόπηση

| | |
|--|----|
| 2.1 Θεωρία βελτιστοποίησης | 4 |
| 2.1.1 Εισαγωγή | 4 |
| 2.1.1.1 Πριν από Το 1900 | 4 |
| 2.1.1.2 Εικοστός Αιώνας | 7 |
| 2.1.2 Κλασικές μέθοδοι βελτιστοποίησης | 10 |
| 2.2 Αλγόριθμος | 11 |
| 2.2.1 Προσεγγιστικοί αλγόριθμοι | 12 |
| 2.2.2 Ευρετικοί και μεταευρετικοί | 12 |
| 2.2.3 Γενετικοί Αλγόριθμοι | 14 |

3^Ο ΚΕΦΑΛΑΙΟ

Αλγόριθμοι Νοημοσύνης Σμηνών

| | |
|--|----|
| 3.1 Εισαγωγή | 16 |
| 3.2 Χαρακτηριστικά της νοημοσύνης των σμηνών | 18 |

4^Ο ΚΕΦΑΛΑΙΟ

Αλγόριθμοι Νοημοσύνης Σμηνών

| | |
|---|----|
| 4.1 Αλγόριθμος της πυγολαμπίδας | 20 |
| 4.2 Αλγόριθμος Ant Colony Optimization (ACO) | 21 |
| 4.3 Αλγόριθμος τεχνητής αποικίας μελισσών | 22 |
| 4.4 Αλγόριθμος βελτιστοποίησης ζευγαρώματος μελισσών | 22 |
| 4.5 Αλγόριθμος της νυχτερίδας | 23 |
| 4.6 Αλγόριθμος Βελτιστοποίησης Σμήνους Σωματιδίων (Particle Swarm Optimization - PSO) | 23 |
| 4.7 Εφαρμογές αλγορίθμων νοημοσύνης σμήνους | 28 |

5^ο ΚΕΦΑΛΑΙΟ

Μη Επανδρωμένα Οχήματα(UAVs)

| | |
|---|----|
| 5.1 Εισαγωγή..... | 31 |
| 5.2 Ιστορική αναδρομή..... | 31 |
| 5.3 Τύποι UAVs | 34 |
| 5.4 Εφαρμογές UAVs..... | 36 |
| 5. 5 Κατηγορίες UAVs | 36 |
| 5. 6 Μαθηματικό μοντέλο UAVs | 36 |
| 5.7 Κινηματικό μοντέλο UAVs..... | 39 |
| 5.7.1 Χωρόδετοι (αδρανειακοί) άξονες..... | 39 |
| 5.7.2 Σωματόδετοι άξονες αεροσκάφους | 40 |
| 5. 8 Επικοινωνίες UAVs..... | 43 |
| 5.8.1 Ασύρματη επικοινωνία..... | 43 |
| 5.8.2 Ασύρματες τεχνολογίες στην επικοινωνία των UAVs..... | 43 |
| 5.8.3 Τα βασικά ενός συστήματος επικοινωνίας..... | 45 |
| 5.8.4 Πρωτόκολλα ασύρματης επικοινωνίας | 45 |
| 5.8.5 Παράγοντες που επηρεάζουν την επικοινωνία των UAVs..... | 46 |
| 5. 9 Αισθητήρια του UAVs..... | 47 |
| 5.9.1 Γυροσκόπιο..... | 47 |
| 5.9.2 Επιταχυνσιόμετρο..... | 47 |
| 5.9.3 Μαγνητόμετρο..... | 48 |
| 5.9.4 Βαρόμετρο/Αισθητήρας πίεσης..... | 48 |
| 5.9.5 GPS..... | 49 |
| 5.10 Ελεγκτής πτήσης | 49 |
| 5. 11 Πως λειτουργεί το UAV | 50 |
| 5. 12 Σμήνος UAVs | 51 |

6^ο ΚΕΦΑΛΑΙΟ

Εφαρμογή και Υλοποίηση Αλγορίθμου

| | |
|--------------------------------|----|
| 6.1 Test Functions | 52 |
| 6.1.1 Beale Function..... | 52 |
| 6.1.2 Eggholder Function | 53 |

| | |
|--|-----------|
| 6.1.3 Ackley Function | 54 |
| 6.1.4 Booth Function | 56 |
| 6.1.5 Rosenbrock Function..... | 57 |
| 6.2 Εφαρμογή και υλοποίηση αλγορίθμου..... | 59 |
| 6.2.1 Σχηματισμός από ευθεία σε εξάγωνο | 59 |
| 6.2.2 Σχηματισμός από ευθεία σε τρίγωνο | 69 |
| 6.2.3 Σχηματισμός από ευθεία σε ημικόκλιο | 72 |
| 6.2.4 Σχηματισμός από εξάγωνο σε τρίγωνο..... | 75 |
| 6.2.5 Σχηματισμός από μετατοπισμένο εξάγωνο σε πεντάγωνο | 78 |
| 6.2.6 Σχηματισμός από πεντάγωνο σε ημικόκλιο | 81 |
| ΠΑΡΑΡΤΗΜΑ..... | 84 |
| Λίστα Εικόνων..... | 84 |
| Λίστα Πινάκων | 87 |
| ΒΙΒΛΙΟΓΡΑΦΙΑ..... | 89 |
| Ηλεκτρονικές Πηγές..... | 91 |

Περίληψη

Στην παρούσα εργασία γίνεται αναφορά στις βασικές αρχές της νοημοσύνης σμήνους και ειδικότερα στη μελέτη αξιολόγηση και υλοποίηση αλγορίθμων για τον έλεγχο σμήνους μη επανδρωμένων οχημάτων.

Σκοπό έχει να βοηθήσει τον εκάστοτε αναγνώστη να κατανοήσει την απήχηση που θα έχει στο εγγύς μέλλον το βασικό υπόβαθρο της εργασίας, το οποίο είναι η Νοημοσύνη Σμήνους. Παρουσιάζεται μία βιβλιογραφική ανασκόπηση της Νοημοσύνης Σμήνους καθώς και τους αλγόριθμους που συναντάμε. Τέλος γίνεται μία μελέτη για τα μη επανδρωμένα αεροσκάφη, πιο συγκεκριμένα στα αυτόνομα αεροσκάφη που πετούν σε σχηματισμό ενός σμήνους που η μετάβαση από το αρχικό σχηματισμό στο τελικό επιθυμητό σχηματισμό και ο έλεγχος τους επιταχύνεται χρησιμοποιώντας κατάλληλους αλγορίθμους.

Abstract

This paper refers to the basic principles of swarm intelligence and in particular the study, evaluation and implementation of algorithms for the cluster control unmanned vehicles.

Its aim is to help the reader understand the impact that will, swarm intelligence, have in the near future, the basic idea of this thesis.

There is presented a bibliographical review of Intelligence Cluster, the algorithms found and unmanned aircrafts. More precisely, a detailed analysis follows concerning the autonomous aircraft flying in formation of a cluster, the transition from initial formation to the final desired while their control is accelerated using appropriate algorithms.

Εισαγωγή

Το 1992, ο Marco Dorigo τελείωσε τη διδακτορική του διατριβή για τη βελτιστοποίηση και τους φυσικούς αλγόριθμους, στην οποία περιέγραψε το καινοτόμο έργο του σχετικά με τη βελτιστοποίηση αποικίας μυρμηγκιών (ACO). Αυτή η τεχνική αναζήτησης είναι εμπνευσμένη από τη νοημοσύνη σμήνους των κοινωνικών μυρμηγκιών, που χρησιμοποιούν τη φερομόνη ως χημικό αγγελιοφόρο. Στη συνέχεια, το 1992, ο John R. Koza του Πανεπιστημίου του Στάνφορντ δημοσίευσε μια πραγματεία περί γενετικού προγραμματισμού, η οποία έθεσε τα θεμέλια μιας ολόκληρης νέας περιοχής της μηχανικής μάθησης, κι έφερε επανάσταση στον προγραμματισμό υπολογιστών. Ήδη από το 1988, ο Koza εφάρμοσε το πρώτο του δίπλωμα ευρεσιτεχνίας για το γενετικό προγραμματισμού. Η βασική ιδέα είναι η χρήση της γενετικής αρχής για να αναπαράγονται προγράμματα ηλεκτρονικών υπολογιστών, έτσι ώστε να παράγουν τα καλύτερα προγράμματα για ένα δεδομένο είδος προβλήματος.

Λίγο αργότερα, το 1995, έγινε ακόμα ένα σημαντικό βήμα προόδου στην ανάπτυξη της βελτιστοποίησης σμήνους σωματιδίων (PSO) από τον Αμερικανό κοινωνικό ψυχολόγο James Kennedy, και το μηχανικό Russell C. Eberhart. Σε αδρές γραμμές η PSO είναι ένας αλγόριθμος βελτιστοποίησης εμπνευσμένος από τη νοημοσύνη σμήνους των ψαριών και πουλιών καθώς και από την ανθρώπινη συμπεριφορά. Οι πολλαπλοί παράγοντες, που ονομάζονται σωματίδια, δημιουργούν ένα σμήνος γύρω από το χώρο αναζήτησης ξεκινώντας από κάποια αρχική τυχαία εικασία. Το σμήνος επικοινωνεί τις τρέχουσες καλύτερες λύσεις και μοιράζεται την καθολικά καλύτερη, έτσι ώστε να επικεντρωθεί στις ποιοτικές λύσεις. Από το σχεδιασμό της, υπήρξαν περίπου 20 διαφορετικές παραλλαγές σμήνους σωματιδίων, και έχουν εφαρμοστεί σε όλους σχεδόν τους τομείς των δύσκολων προβλημάτων βελτιστοποίησης. Υπάρχουν κάποιες ισχυρές ενδείξεις ότι η PSO είναι καλύτερη από τις παραδοσιακούς αλγόριθμους αναζήτησης και ακόμη καλύτερα από τους γενετικούς αλγόριθμους για τα περισσότερα είδη των προβλημάτων, αν και αυτό απέχει από το να είναι οριστικό συμπέρασμα.

Το 1997, η δημοσίευση του «θεωρήματα βελτιστοποίησης για όχι δωρεάν γεύματα» από τον D. H. Wolpert και τον W. G. Macready σόκαρε την κοινότητα βελτιστοποίησης. Οι ερευνητές πάντα προσπαθούν να βρουν καλύτερους αλγόριθμους, ή ακόμη και με καθολική ισχύ αλγόριθμους, για τη βελτιστοποίηση, ειδικά για δύσκολα NP-hard προβλήματα βελτιστοποίησης. Ωστόσο, αυτά τα θεωρήματα δηλώνουν ότι εάν ο αλγόριθμος A αποδίδει καλύτερα από τον αλγόριθμο B για ορισμένες λειτουργίες βελτιστοποίησης, ο αλγόριθμος B θα ξεπεράσει A για άλλες λειτουργίες. Αυτό σημαίνει ότι στο εύρος όλων των πιθανών

λειτουργιών, τόσο ο αλγόριθμος A όσο και ο αλγόριθμος B θα αποδώσουν κατά μέσο όρο εξίσου καλά. Ή με άλλα λόγια, δεν υπάρχει καθολικά καλύτερος αλγόριθμος. Αυτό είναι απογοητευτικό, έτσι δεν είναι; Στη συνέχεια, οι άνθρωποι συνειδητοποίησαν για ένα δεδομένο πρόβλημα βελτιστοποίησης δεν χρειαζόμαστε το μέσο όρο όλων των πιθανών λειτουργιών. Αυτό δεν έχει καμία σχέση με το γενικό μέσο όρο όλου του εύρους λειτουργίας. Επιπλέον, μπορούμε να δεχτούμε το γεγονός ότι δεν υπάρχει ένα καθολικό ή μαγικό εργαλείο, αλλά γνωρίζουμε από την εμπειρία μας ότι ορισμένοι αλγόριθμοι ξεπερνούν τους άλλους για συγκεκριμένους τύπους προβλημάτων βελτιστοποίησης. Έτσι, η έρευνα επικεντρώνεται τώρα στην εξεύρεση των καλύτερων και πιο αποτελεσματικών αλγορίθμων για ένα δεδομένο πρόβλημα. Ο στόχος είναι ο σχεδιασμός των καλύτερων αλγορίθμων για τους περισσότερους τύπους προβλημάτων, όχι για όλα τα προβλήματα. Ως εκ τούτου, η έρευνα συνεχίζεται.

Στην αυγή του εικοστού πρώτου αιώνα, τα πράγματα έγιναν ακόμα πιο συναρπαστικά. Κατ' αρχάς, ο Zong Woo Geem et al. το 2001 ανέπτυξε τον αλγόριθμο Harmony Search (HS), ο οποίος έχει εφαρμοστεί ευρέως στην επίλυση διαφόρων προβλημάτων βελτιστοποίησης, όπως η διανομή νερού, η μοντελοποίηση και ο προγραμματισμός των μεταφορών.

Το 2004, οι S. Nakrani και C. Tovey πρότειναν τον αλγόριθμο Honey Bee και τις εφαρμογές του για τη βελτιστοποίηση των κέντρων φιλοξενίας στο διαδίκτυο, η οποία ακολουθείται από την ανάπτυξη ενός νέου αλγορίθμου μέλισσα (bee algorithm) από τον D. T. Pham et al. το 2005 και τον Artificial Bee Colony (ABC) από τον Karaboga το 2005. Το 2008, ο συγγραφέας του βιβλίου αυτού, ανέπτυξε τον Αλγόριθμο Firefly (FA). Ακολούθησαν αρκετά ερευνητικά άρθρα σχετικά με τον αλγόριθμο Firefly, και ο αλγόριθμος αυτός έχει προσελκύσει έντονο ενδιαφέρον.



Εικόνα 1.1: Ένα μη επανδρωμένο αεροσκάφος

ΚΕΦΑΛΑΙΟ 2

Βιβλιογραφική Ανασκόπηση

2.1 Θεωρία Βελτιστοποίησης

2.1.1 Εισαγωγή

Η βελτιστοποίηση υπάρχει παντού, από το μηχανολογικό σχέδιο ως τις οικονομικές αγορές, από τις καθημερινές μας δραστηριότητες ως τον προγραμματισμό των διακοπών και από τις επιστήμες των υπολογιστών ως τις βιομηχανικές εφαρμογές. Πάντα έχουμε στόχο να μεγιστοποιήσουμε ή να ελαχιστοποιήσουμε κάτι. Μια οργάνωση θέλει να μεγιστοποιήσει τα κέρδη της, να ελαχιστοποιήσει τα έξοδά της και να μεγιστοποιήσει την απόδοσή της. Ακόμα κι όταν απλώς σχεδιάζουμε τις διακοπές μας, θέλουμε να μεγιστοποιήσουμε τη διασκέδαση και την ευχαρίστησή μας με τα ελάχιστα έξοδα (ή ακόμα καλύτερα δωρεάν). Στην πραγματικότητα ψάχνουμε διαρκώς για την καλύτερη λύση σε κάθε πρόβλημα που αντιμετωπίζουμε, παρόλο που δεν είμαστε πάντοτε σε θέση να βρούμε τέτοιες λύσεις.

Δεν θα ήταν υπερβολή αν λέγαμε ότι η αναζήτηση λύσεων σε προβλήματα βελτιστοποίησης είναι τόσο παλιά όσο και η ιστορία του ανθρώπου. Για παράδειγμα η αρχή της ήσσονος προσπάθειας μπορεί να εξηγήσει πολλές φορές την ανθρώπινη συμπεριφορά. Γνωρίζουμε ότι η μικρότερη απόσταση μεταξύ δυο οποιωνδήποτε διαφορετικών σημείων ενός επιπέδου είναι μια ευθεία γραμμή, παρόλο που συχνά απαιτούνται σύνθετοι μαθηματικοί υπολογισμοί όπως ο λογισμός των μεταβολών για να αποδειχθεί και τυπικά ότι ένα ευθύγραμμο τμήμα μεταξύ δυο σημείων είναι πράγματι το μικρότερο.

Στην πραγματικότητα, πολλά φυσικά φαινόμενα διακατέχονται από τη λεγόμενη αρχή της ελάχιστης δράσης ή παραλλαγές της. Για παράδειγμα το φως κατά τη διαδρομή του μέσα σε ένα μέσον από ένα σημείο σε ένα άλλο, ακολουθεί τη διαδρομή που θα του επιτρέψει να την διανύσει στο μικρότερο δυνατό χρόνο (αρχή του Φερμά ή αρχή του ελάχιστου χρόνου), κι έχει σαν αποτέλεσμα το νόμο του Snell. Ολόκληρη η αναλυτική μηχανική βασίζεται στην αρχή της ελάχιστης δράσης.

2.1.1.1 ΠΡΙΝ ΑΠΟ ΤΟ 1900

Η μελέτη της βελτιστοποίησης των προβλημάτων είναι τόσο παλιά όσο και η επιστήμη. Είναι γνωστό ότι οι αρχαίοι Έλληνες μαθηματικοί έλυσαν πολλά προβλήματα βελτιστοποίησης. Για παράδειγμα ο Ευκλείδης περίπου το 300 π.Χ. απέδειξε ότι το εμβαδόν του τετραγώνου είναι μεγαλύτερο από το εμβαδόν κάθε άλλου ορθογώνιου

παραλληλόγραμμου με το ίδιο συνολικό μήκος των τεσσάρων πλευρών. Αργότερα, γύρω στο 100 π.Χ., ο Ήρων υπέδειξε την αρχή του ελαχίστου δρόμου: «Όταν μια ακτίνα φωτός ανακλάται σε ένα κάτοπτρο, η πραγματική διαδρομή που ακολουθεί είναι συντομότερη ως προς οποιαδήποτε άλλη δυνατή διαδρομή της ανακλώμενης ακτίνας» ότι η απόσταση μεταξύ δυο σημείων κατά μήκος της διαδρομής αντανάκλασης από ένα κάτοπτρο, είναι μικρότερη όταν το κάτοπτρο είναι συμμετρικό και η γωνία προσπτώσεως είναι ίση με τη γωνία ανακλάσεως. Πρόκειται για ένα γνωστό πρόβλημα βελτιστοποίησης, το πρόβλημα του Ήρωνα, που περιεγράφηκε πρώτα στα Κατοπτρικά του Ήρωνα.

Ο περίφημος Γερμανός αστροφυσικός Γ. Κέπλερ, έγινε γνωστός κυρίως για την ανακάλυψη των τριών νόμων της κίνησης των πλανητών. Ωστόσο το 1613, έλυσε ένα πρόβλημα βελτιστοποίησης στο καλούμενο πρόβλημα του γάμου ή πρόβλημα της γραμματέως όταν άρχισε να ψάχνει για τη δεύτερη σύζυγό του. Περιγράφει τη μέθοδό του σ' ένα γράμμα του προς το Βαρόνο Strahlendorf (23/10/1613), όπου σταθμίζει τις αρετές και τα μειονεκτήματα της κάθε υποψήφιας, την προίκα, τις αναστολές της και τις συμβουλές των φίλων. Μεταξύ των έντεκα υποψηφίων από τις οποίες πήρε συνέντευξη ο Κέπλερ, επέλεξε την Πέμπτη, παρόλο που οι φίλοι του τον συμβούλευαν να πάρει την τέταρτη. Αυτό μπορεί να σημαίνει ότι ο Κέπλερ προσπαθούσε να βελτιστοποιήσει κάποια συνάρτηση χρησιμότητας. Το πρόβλημα αυτό εισήχθη επίσημα από τον Martin Gardner το 1960 στη στήλη του μαθηματικά παιχνίδια στο τεύχος του Φεβρουάριου του 1960 του περιοδικού Scientific American. Από τότε, έχει εξελιχθεί σε έναν τομέα της βελτιστοποίησης πιθανοτήτων όπως είναι τα βέλτιστα προβλήματα διακοπής.

Ο W. van Royen Snell ανακάλυψε το 1621 το νόμο της διάθλασης, ο οποίος δεν δημοσιεύτηκε. Αργότερα ο Christiaan Huygens ανέφερε τα αποτελέσματα του Snell στα Διοπτρικά του, το 1703. Ο νόμος αυτός ξανά-ανακαλύφτηκε ανεξάρτητα από τον Rene Descartes (Καρτέσιος) και δημοσιεύτηκε στην πραγματεία του Λόγος περί της Μεθόδου το 1673. Είκοσι χρόνια αργότερα, όταν μαθητές του Καρτέσιου ήρθαν σε επαφή με τον Φερμά προκειμένου να συγκεντρώσουν την αλληλογραφία του με τον Καρτέσιο, ο Φερμά ξανακοίταξε το 1657 την διαφωνία του για τη μη ικανοποιητική περιγραφή της διάθλασης του φωτός από τον Καρτέσιο, και κατέληξε στα ίδια αποτελέσματα με τον Snell και τον Descartes θεμελιώνοντάς τα αξιωματικά - το φως κατά τη διαδρομή του μέσα σε οποιοδήποτε μέσον από ένα σημείο σε ένα άλλο, ακολουθεί τη διαδρομή που θα του επιτρέψει να την διανύσει στο μικρότερο δυνατό χρόνο και αυτή η αρχή είναι πλέον σήμερα γνωστή ως αρχή του Φερμά, και αποτελεί τη βάση της σύγχρονης οπτικής.

Στο έργο του Principia Mathematica που δημοσιεύθηκε το 1687, ο Sir Isaac Newton έλυσε το πρόβλημα του σχήματος του σώματος της ελάχιστης αντίστασης που είχε θέσει νωρίτερα το 1685 ως ένα πρωτοποριακό πρόβλημα βελτιστοποίησης, που πλέον είναι ένα πρόβλημα του λογισμού των παραλλαγών. Ο κύριος στόχος ήταν να βρεθεί το σχήμα ενός συμμετρικού σώματος, ώστε να ελαχιστοποιείται η αντίσταση του στην κίνηση σε ένα ρευστό. Ακολούθως, ο Νεύτων κατέληξε στη διατύπωση του νόμου της αντίστασης ενός σώματος. Είναι ενδιαφέρον ότι και ο Galileo Galilei εντελώς ανεξάρτητα διατύπωσε ένα παρόμοιο πρόβλημα το 1683 στο έργο του Discursi.

Τον Ιούνιο του 1696, ο J. Bernoulli σημείωσε κάποια σημαντική πρόοδο στο λογισμό. Σε ένα άρθρο στο Acta Eruditorum, προκάλεσε τους μαθηματικούς σε όλο τον κόσμο να βρουν το σχήμα ή την καμπύλη που συνδέει δύο σημεία σε διαφορετικά ύψη, ώστε το σώμα που θα πέσει κατά μήκος της καμπύλης αυτής λόγω βαρύτητας, να κάνει το συντομότερο χρόνο(η γραμμή της ταχύτερης πτώσης), αν και Bernoulli γνώριζε ήδη τη λύση. Στις 29/1/1697 ο Νεύτων απάντησε στην πρόκληση, όταν γύρισε σπίτι του στις 4 το απόγευμα και δεν κοιμήθηκε μέχρι που έλυσε το πρόβλημα, περίπου στις 4 τα ξημερώματα της επομένης, οπότε και έστειλε τη λύση του. Παρόλο που ο Νεύτων το έλυσε σε λιγότερο από 12 ώρες, όταν έγινε Φύλακας του Βασιλικού Νομισματοκοπείου στις 29/3/1696, βρέθηκαν ορισμένοι που είπαν ότι μια τέτοια ιδιοφυία έπρεπε να το έχει λύσει σε μισή ώρα, ενώ άλλοι εξέφρασαν φόβους ότι η πολλή διοικητική δουλειά θα επιβράδυνε την πρωτότυπη εργασία του και την πρόοδό του. Η λύση όπως τώρα ξέρουμε είναι τμήμα ενός κυκλοειδούς. Αυτή η απότομη κατάβαση καλείται σήμερα Βραχυστόχρονο πρόβλημα και είναι αυτό που ενέπνευσε τον Euler και τον Lagrange να διατυπώσουν τη γενική θεωρία του λογισμού των μεταβολών.

Το 1746 προτάθηκε από τον P. L. de Maupertuis η αρχή της ελάχιστης δράσης, προκειμένου να ενοποιήσει διάφορους νόμους της κίνησης της φυσικής και τις εφαρμογές τους και να εξηγήσει διάφορα φαινόμενα. Στη σύγχρονη ορολογία, είναι μια μεταβολική αρχή της στατικής δράσης με όρους διαφορικής εξίσωσης (integral equation) αποτελεί αναπόσπαστο ενός λειτουργικού, στο πλαίσιο του λογισμού των μεταβολών, η οποία διαδραματίζει κεντρικό ρόλο στην Λαγκρανσιανή και Χαμιλτονιανή κλασική μηχανική. Αποτελεί επίσης βασική αρχή στα μαθηματικά και στη φυσική.

Το 1781, ο Gaspard Monge, ένας Γάλλος πολιτικός μηχανικός, διερεύνησε το πρόβλημα της μεταφοράς για τη βέλτιστη μεταφορά και την κατανομή των πόρων, αν η αρχική και η τελική χωρική κατανομή είναι γνωστές. Το 1942, ο Leonid Kantorovich έδειξε ότι αυτό το πρόβλημα συνδυαστικής βελτιστοποίησης είναι στην πραγματικότητα μια περίπτωση ενός γραμμικού προβλήματος προγραμματισμού.

Γύρω στο 1801, ο Frederich Gauss ισχυρίστηκε ότι χρησιμοποίησε τη μέθοδο των ελαχίστων τετραγώνων για να προβλέψει τη θέση της τροχιάς του αστεροειδή Ceres, αν και η εκδοχή του για τα ελάχιστα τετράγωνα με πιο αυστηρή μαθηματική θεμελίωση δημοσιεύθηκε αργότερα, το 1809. Το 1805, ο Adrien Legendre ήταν ο πρώτος που περιέγραψε τη μέθοδο των ελαχίστων τετραγώνων σε ένα παράρτημα του βιβλίου του *''Nouvelle methodes pour la determination des orbites des cometes''*, και το 1806 ο ίδιος χρησιμοποίησε την αρχή των ελαχίστων τετραγώνων για την προσαρμογή της καμπύλης. Ο Gauss αργότερα ισχυρίστηκε ότι χρησιμοποιούσε τη μέθοδο αυτή για περισσότερα από 20 χρόνια, και έθεσε τα θεμέλια για την ανάλυση ελαχίστων τετραγώνων το 1795. Αυτό τον οδήγησε σε ορισμένες πικρές διαμάχες με τον Legendre. Το 1808, ο Robert Adrain, μη γνωρίζοντας το έργο του Legendre, δημοσίευσε τη μέθοδο των ελαχίστων τετραγώνων μελετώντας την αβεβαιότητα και λάθη παρατηρήσεων, χωρίς να χρησιμοποιήσει την ίδια ορολογία με εκείνη που Legendre.

Το 1815, D. Ricardo πρότεινε το νόμο της φθίνουσας απόδοσης για την καλλιέργεια της γης, που μπορεί να εφαρμοστεί σε πολλές δραστηριότητες. Για παράδειγμα, η παραγωγικότητα ενός χωραφιού ή ενός εργοστασίου θα αυξηθεί οριακά μόνο με πρόσθετη αύξηση των εισροών. Ο νόμος αυτός καλείται νόμος της αύξησης του κόστους ευκαιρίας. Υπαγορεύει ότι υπάρχει μια θεμελιώδης σχέση μεταξύ ευκαιριών και σπανιότητας των πόρων, απαιτώντας οι διαθέσιμοι πόροι που σπανίζουν να χρησιμοποιούνται αποτελεσματικά.

Το 1847 σε ένα σύντομο σημείωμα, ο L. A Cauchy πρότεινε μια γενική μέθοδο για την επίλυση συστημάτων εξισώσεων με επαναληπτικό τρόπο. Αυτό οδηγεί ουσιαστικά σε δύο επαναληπτικές/ επαναλαμβανόμενες μεθόδους ελαχιστοποίησης: οι γνωστές σήμερα σταδιακή μέθοδος (gradient method) και απότομη κάθοδο, για ορισμένες λειτουργίες με περισσότερες από μία μεταβλητές.

2.1.1.2 Εικοστός Αιώνας

Το 1906, ο Δανός μαθηματικός J. Jensen εισήγαγε την έννοια της κυρτότητας και υπολόγισε μια ανισότητα, που τώρα αναφέρονται ως ανισότητα Jensen, η οποία παίζει σημαντικό ρόλο στην βελτιστοποίηση της καμπύλης/κυρτότητας (Convex optimization), αλλά και σε άλλους τομείς όπως η οικονομία.

Η βελτιστοποίηση της καμπύλης/κυρτότητας είναι μια ειδική, αλλά πολύ σημαντική κατηγορία των μαθηματικών βελτιστοποίησης, καθώς κάθε βέλτιστη που βρίσκεται είναι εγγυημένα και καθολική βέλτιστη. Ένα ευρύτερο φάσμα προβλημάτων βελτιστοποίησης μπορεί να επαναδιατυπωθούν με όρους βελτιστοποίησης καμπύλης/κυρτότητας (Convex optimization). Κατά συνέπεια, έχει πολλές εφαρμογές, συμπεριλαμβανομένων των συστημάτων ελέγχου, της τοποθέτησης δεδομένων και της κατασκευής μοντέλων, του

βέλτιστου σχεδιασμού, της επεξεργασίας σήματος, των χρηματοοικονομικών μαθηματικών κ.α.

Ήδη από το 1766, ο Leonhard Euler μελέτησε το πρόβλημα περιοδείας του Knight, και ο T. P. Kirkman δημοσίευσε σε άρθρο την έρευνά του σχετικά με την εύρεση ενός κυκλώματος το οποίο περνά από κάθε κορυφή ενός γραφήματος δεδομένου πολυέδρου μία φορά και μόνο μία φορά. Το 1856, ο Sir William Rowan Hamilton διέδωσε το «Το δωδεκαέδρο του ταξιδιώτη» (the icosian game). Έπειτα, το Φεβρουάριο του 1930, ο Karl Menger έθεσε στη διάρκεια ενός μαθηματικού Συνεδρίου στη Βιέννη το πρόβλημα του ταξιδιώτη, το οποίο συχνά αντιμετωπίζουν οι ταχυδρομικοί διανομείς και οι ταξιδιώτες. Το έργο του δημοσιεύθηκε αργότερα, το 1932. Σκοπός του προβλήματος είναι να βρεθεί η συντομότερη διαδρομή που συνδέει έναν πεπερασμένο αριθμό σημείων / πόλεων του οποίου οι αποστάσεις ανά ζεύγη είναι γνωστές. Αν και το πρόβλημα είναι επιλύσιμο σε πεπερασμένο αριθμό δοκιμών και παραλλαγών, δεν υπάρχει αλγόριθμος για την εύρεση τέτοιας λύσης. Σε γενικές γραμμές, ο απλός κανόνας να πηγαίνει ο διανομέας ή ο ταξιδιώτης στα πλησιέστερα σημεία, δεν έχει ως αποτέλεσμα τη συντομότερη διαδρομή. Το πρόβλημα αυτό αναφέρεται τώρα ως Το Πρόβλημα του Περιοδεύοντος Πωλητή, και είναι στενά συνδεδεμένο με πολλές διαφορετικές εφαρμογές όπως το δρομολόγιο, η κατανομή των πόρων, ο προγραμματισμός και γενικά η επιχειρησιακή έρευνα. Στην πραγματικότητα, ήδη από το 1832, το εγχειρίδιο πωλητή του 1832 περιγράφεται μια περιήγηση κατά μήκος 45 γερμανικών πόλεων με συντομότερη διαδρομή των 1.248 χιλιομέτρων, αν και οι ακριβείς μαθηματικές ρίζες αυτού του προβλήματος είναι αρκετά ασαφείς, και θα μπορούσε να βρεθούν κάποιο χρονικό διάστημα πριν από το 1930.

Είναι ενδιαφέρον, ότι ο Hancock δημοσίευσε το 1917 το πρώτο βιβλίο σχετικά με τη βελτιστοποίηση, με τίτλο " Theory of Minima and Maxima".

Το 1939, ο L. Kantorovich ήταν ο πρώτος που ανέπτυξε έναν αλγόριθμο για το γραμμικό προγραμματισμό και να τον χρησιμοποίησε στα οικονομικά. Διατύπωσε το πρόβλημα του βέλτιστου σχεδιασμού και στην των αποτελεσματικών μεθόδων για την εξεύρεση λύσεων στην παραγωγή με τη βοήθεια του γραμμικού προγραμματισμού. Για το έργο αυτό (Θεωρία της Βέλτιστης Κατανομής των Πόρων), μοιράστηκε το βραβείο Νόμπελ Οικονομικών με τον T. Koopmans το 1975. Το επόμενο σημαντικό βήμα προόδου είναι ότι ο George Dantzig εφηύρε το 1947 τη μέθοδο simplex για την επίλυση μεγάλης κλίμακας προβλημάτων γραμμικού προγραμματισμού. Ο Dorfman σε ένα άρθρο που δημοσιεύθηκε το 1984, έγραψε ότι ο γραμμικός προγραμματισμός ανακαλύφθηκε τρεις φορές, ανεξάρτητα, μεταξύ 1939 και 1947, αλλά κάθε χρόνο σε μια κάπως διαφορετική μορφή. Την πρώτη ανακάλυψη έκανε ο

Ρώσος μαθηματικός L. Kantorovich, στη συνέχεια, ο ολλανδός οικονομολόγος, Koopmans, και την τρίτη το 1947 ο Αμερικανός μαθηματικό George Dantzig. Η επαναστατική μέθοδος Simplex του Dantzig , είναι σε θέση να λύσει ένα ευρύ φάσμα προβλημάτων βέλτιστης πολιτικής απόφασης μεγάλης πολυπλοκότητας. Ένα κλασικό παράδειγμα και μία από τις πρώτες εφαρμογές του γραμμικού προγραμματισμού, όπως περιγράφεται στο βιβλίο του Dantzig του 1963 ήταν η εξεύρεση λύσης στο ειδικό πρόβλημα βέλτιστης διατροφής που περιλαμβάνει 9 εξισώσεις και 77 αγνώστους χρησιμοποιώντας χειροκίνητες επιτραπέζιες αριθμομηχανές.

Το 1951, ο Harold Kuhn και ο A. W. Tucker μελέτησαν την μη γραμμική βελτιστοποίηση του προβλήματος και ανέπτυξαν εκ νέου τη βέλτιστη συνθήκη, καθώς παρόμοιες συνθήκες είχαν προτείνει ο W. Karush το 1939 στη διατριβή του για το MSc. Στην πραγματικότητα, οι βέλτιστες συνθήκες είναι η γενίκευση των πολλαπλασιαστών Lagrange σε μη γραμμικές ανισότητες, και είναι γνωστές σήμερα ως Συνθήκες Karush-Kuhn-Tucker, ή απλά συνθήκες Kuhn-Tucker, οι οποίες είναι απαραίτητες προϋποθέσεις ώστε μια λύση να είναι βέλτιστη σε μη γραμμικό προγραμματισμό.

Στη συνέχεια, το 1957, ο Richard Bellman στο Πανεπιστήμιο του Στάνφορντ ανέπτυξε το δυναμικό προγραμματισμό και την αρχή βελτιστοποίησης όταν μελετούσε τις διαδικασίες λήψης αποφάσεων και σχεδιασμού σε πολλαπλά στάδια, ενώ παράλληλα ο ίδιος πέρασε κάποιο χρονικό διάστημα στο RAND Corporation. Αυτός έπλασε και τον όρο Δυναμικός Προγραμματισμός. Η ιδέα του δυναμικού προγραμματισμού μπορεί να χρονολογούνται από το 1944, όταν ο John von Neumann και ο O. Morgenstern μελέτησαν τα προβλήματα διαδοχικών αποφάσεων. Ο John von Neumann, επίσης, είχε σημαντική συμβολή στην ανάπτυξη της επιχειρησιακής έρευνας. Ήδη από το 1840, ο Charles Babbage μελέτησε το κόστος της μεταφοράς και της διαλογής της αλληλογραφίας , και αυτό θα μπορούσε να είναι η πρώτη μελέτη σχετικά με την επιχειρησιακή έρευνα.

Σημαντική πρόοδος σημειώθηκε κατά τη διάρκεια του Δευτέρου Παγκοσμίου Πολέμου, και από τότε επεκτάθηκε για την εξεύρεση των βέλτιστων ή σχεδόν βέλτιστων λύσεων σε ένα πολύπλοκο και ευρύ φάσμα διεπιστημονικών προβλημάτων, όπως τα δίκτυα επικοινωνίας, ο σχεδιασμός έργων, ο προγραμματισμός, το σχεδιασμός των μεταφορών και η διαχείριση.

Μετά τη δεκαετία του 1960, υπήρξε έκρηξη στη βιβλιογραφία σχετικά με τη βελτιστοποίηση, και θα χρειαζόταν ένα ολόκληρο βιβλίο για να γράψει κάποιος ακόμα και μια σύντομη ιστορία για τη βελτιστοποίηση μετά το 1960. Καθώς αυτό το βιβλίο είναι κυρίως για την εισαγωγή στους μεταερευνητικούς αλγόριθμους, θα εστιάσουμε την προσοχή μας στην

ανάπτυξη των ευρετικών και μεταευρετικών. Στην πραγματικότητα, αρκετά σημαντικός αριθμός νέων αλγορίθμων βελτιστοποίησης είναι κυρίως μεταευρετικοί.

2.1.2 Κλασικές Μέθοδοι Βελτιστοποίησης

Εκτός από τον κλασικό διαχωρισμό των μεθόδων βελτιστοποίησης σε ολικές και τοπικές υπάρχουν και άλλοι τρόποι κατηγοριοποίησής τους, ανάλογα με το αν χρησιμοποιούν παραγώγους, πληθυσμούς, αν είναι αλγοριθμικές και αν έχουν εγγυημένη σύγκλιση και ακρίβεια. Ακολουθώντας την κατηγοριοποίηση των Beasley, Bull και Martin, οι μέθοδοι βελτιστοποίησης θα μπορούσαν να ταξινομηθούν ως ακολούθως :

- Μέθοδοι δασισμένες στον Λογισμό (calculus-based methods). Οι μέθοδοι αυτές στηρίζονται κυρίως στην χρήση παραγώγων και μπορούν να χωριστούν σε άμεσες (direct) και έμμεσες (indirect). Οι άμεσες αναζητούν τοπικά ακρότατα της συνάρτησης κάνοντας μικρά άλματα (hill-climbing), ενώ οι έμμεσες συνήθως αξιοποιούν την τοπική πληροφορία με σκοπό τον εντοπισμό τοπικών ακροτάτων. Οι μέθοδοι αυτής της κατηγορίας έχουν μελετηθεί πλήρως. Βασικό τους μειονέκτημά είναι η ισχυρή εξάρτηση από τις αρχικές συνθήκες και η τοπικότητα στην εμβέλεια.
- Μέθοδοι τυχαίας αναζήτησης (random search methods). Οι μέθοδοι αυτές παράγουν τυχαία σημεία τα οποία αξιολογούνται και αποφασίζουν αν είναι επιθυμητά ή όχι. Σπανίως χρησιμοποιούνται μόνες τους γιατί είναι γνωστή η μειωμένη αποδοτικότητά τους. Συνήθως χρησιμοποιούνται συνδυαστικά με μεθόδους τοπικής αναζήτησης, στις οποίες παρέχουν τυχαία αρχικά σημεία για τις επανεκκινήσεις του αλγορίθμου, δίνοντας την επόμενη κατηγορία μεθόδων.
- Μέθοδοι επαναλαμβανόμενης αναζήτησης (iterated search methods). Αυτές οι μέθοδοι συνδυάζουν τις μεθόδους hill-climbing με τις μεθόδους τυχαίας αναζήτησης. Έτσι, όταν η πρώτη μέθοδος φτάσει σε ένα τοπικό ελάχιστο, ενεργοποιείται η δεύτερη, η οποία παρέχει ένα νέο αρχικό σημείο στον αλγόριθμο, από το οποίο ξεκινάει ένα νέο hill-climbing. Αυτές οι μέθοδοι έχουν το πλεονέκτημά της απλότητας και μπορούν να αποφεύγουν τοπικούς ελαχιστοποιητές, αλλά η απόδοσή τους μειώνεται δραματικά όσο αυξάνεται ο αριθμός των ακροτάτων της συνάρτησης.

- Μέθοδοι προσομοιωμένης ανόπτωσης (simulated annealing). Αποτελούν τροποποίηση των μεθόδων hill-climbing, όπου τα άλματα πραγματοποιούνται βάση ενός σχήματος δανεισμένου από τις φυσικές διαδικασίες ανόπτωσης (annealing procedures) στα μέταλλα. Πρόκειται για πολύ διαδεδομένη κλάση στοχαστικών αλγορίθμων, με πολλές εφαρμογές. Βασικά της μειονεκτήματα είναι ότι σε κάθε βήμα μελετάται.

2.2 Αλγόριθμος

Η ονομασία Αλγόριθμος προέρχεται από το όνομα του Άραβα Μαθηματικού Al-Khowârizmi (με καταγωγή από το Ουζμπεκιστάν, που έζησε στη Βαγδάτη τον 9ο αιώνα μ.Χ.). Ήταν ο πρώτος που διατύπωσε τους κανόνες για τις 4 βασικές αριθμητικές πράξεις (από δικό του βιβλίο προέρχεται και η Άλγεβρα).

Υπενθυμίζουμε ότι σκοπός ενός αλγορίθμου είναι η επίλυση ενός προβλήματος. Σε έναν αλγόριθμο μας ενδιαφέρουν: ορθότητα, πολυπλοκότητα, αν εφαρμόζεται γενικά (δηλαδή για όλα τα πιθανά στιγμιότυπα εισόδου), αν είναι βέλτιστος, ακριβής ή προσεγγιστικός, πιθανοτικός κ.τ.λ.

Ως αλγόριθμος ορίζεται μια πεπερασμένη σειρά ενεργειών, αυστηρά καθορισμένων και εκτελέσιμων σε πεπερασμένο χρόνο που αποτελείται από μία σειρά υπολογιστικών διεργασιών, που στοχεύουν στην επίλυση ενός προβλήματος. Πιο απλά, αλγόριθμο ονομάζουμε μία σειρά από εντολές που έχουν αρχή και τέλος, είναι σαφείς και εκτελέσιμες και έχουν σκοπό την επίλυση κάποιου προβλήματος. Τα αντικείμενα πάνω στα οποία επενεργούν αυτές οι διεργασίες λέγονται δεδομένα (data).

Ο αλγόριθμος χαρακτηρίζεται από τα παρακάτω πέντε στοιχεία:

- Κάθε εκτέλεση είναι πεπερασμένη ,δηλαδή τελειώνει ύστερα από έναν πεπερασμένο αριθμό διεργασιών ή βημάτων (finiteness).
- Κάθε κανόνας του ορίζεται επακριβώς και η αντίστοιχη διεργασία είναι συγκεκριμένη (definiteness).
- Έχει μηδέν ή περισσότερα μεγέθη εισόδου που δίδονται εξαρχής, πριν αρχίσει να εκτελείται ο αλγόριθμος (input).
- Δίνει τουλάχιστον ένα μέγεθος σαν αποτέλεσμα (έξοδο - output) που εξαρτάται κατά κάποιο τρόπο από τις αρχικές εισόδους.

- Είναι μηχανιστικά αποτελεσματικός, δηλαδή όλες οι διαδικασίες που περιλαμβάνει μπορούν να πραγματοποιηθούν με ακρίβεια και σε πεπερασμένο χρόνο “με μολύβι και χαρτί” (effectiveness)

2.2.1 Προσεγγιστικοί Αλγόριθμοι

Εκτός από τους ακριβείς αλγόριθμους υπάρχουν και οι προσεγγιστικοί αλγόριθμοι, οι οποίοι δεν δίνουν απαραίτητα τη βέλτιστη λύση, αλλά εγγυώνται μια συγκεκριμένη απόδοση (Vazirani, 2003). Στην βιβλιογραφία υπάρχουν αρκετοί προσεγγιστικοί αλγόριθμοι για επίλυση προβλημάτων τύπου ODP (Li & Cheng, 1990 - Miyazawa & Wakabayashi, 2007 - Frederickson & Solis-Oba, 2006 - Bansal, et al., 2007) όπως επίσης και για προβλήματα τύπου SSSBPP από τους Miyazawa και Wakabayashi (1999, 2007, 2009). Το κυριότερο μειονέκτημα τους είναι ότι οι αλγόριθμοι αυτοί δεν έχουν δοκιμασθεί με κάποιο περιορισμό. Η σημαντικότερη κατηγορία των προσεγγιστικών αλγορίθμων είναι οι ευρετικοί αλγόριθμοι (heuristic algorithms). Οι ευρετικοί αλγόριθμοι διαχωρίζονται στις ακόλουθες κατηγορίες:

- Κλασικοί ευρετικοί αλγόριθμοι
- Μεταευρετικοί αλγόριθμοι (αλγόριθμοι υπολογιστικής ευφυΐας)

Ένας κλασικός ευρετικός αλγόριθμος αποτελεί έναν αποτελεσματικό αλγόριθμο (εύρεση λύσεων εντός αποδεκτού χρονικού διαστήματος) που παράγει λύσεις σε προβλήματα συνδυαστικής βελτιστοποίησης, όχι απαραίτητα βέλτιστες, και του οποίου η αποδοτικότητα εξαρτάται κάθε φορά από το βαθμό ενσωμάτωσης των ιδιοτήτων του υπό εξέταση προβλήματος στον μηχανισμό λειτουργίας του.

2.2.2 Ευρετικοί Και Μεταευρετικοί

Ευρετικοί είναι μια στρατηγική επίλυσης με δοκιμή-και-σφάλμα μέχρι να προκύψουν αποδεκτές λύσεις σε ένα σύνθετο πρόβλημα σε πρακτικό χρόνο. Η πολυπλοκότητα του προβλήματος καθιστά αδύνατη την αναζήτηση όλων των δυνατών λύσεων ή συνδυασμού αυτών, έτσι ο στόχος είναι να βρούμε καλές, εφικτές λύσεις σε ένα αποδεκτό χρονοδιάγραμμα. Δεν υπάρχει καμία εγγύηση ότι μπορούν να βρεθούν οι άριστες λύσεις, κι επίσης δεν γνωρίζουμε αν ένας αλγόριθμος θα λειτουργήσει και γιατί. Η ιδέα είναι να βρεθεί ένας αποτελεσματικός αλλά πρακτικός αλγόριθμος που θα λειτουργεί στην πλειοψηφία των περιπτώσεων και θα παράγει καλής ποιότητας λύσεις. Μεταξύ των ποιοτικών λύσεων που προκύπτουν, αναμένεται ότι ορισμένες από αυτές είναι σχεδόν βέλτιστες, αν και δεν υπάρχει καμία εγγύηση γι' αυτό.

Ο Alan Turing ήταν ίσως ο πρώτος που χρησιμοποίησε ευρετικούς αλγόριθμους κατά τη διάρκεια του Β' Παγκοσμίου Πολέμου, όταν έσπασε το γερμανικό αλγόριθμο κρυπτογράφησης Enigma στο Bletchley Park, όπου ο Turing, μαζί με τον Βρετανό μαθηματικό Gordon Welchman, σχεδίασαν το 1940 μια ηλεκτρομηχανολογική μηχανή κρυπτογράφησης, το Bombe, για να τους βοηθήσει να σπάσουν τους κώδικες. Η bombe χρησιμοποίησε ένα ευρετικό αλγόριθμο, όταν ο Turing κλήθηκε να ψάξει, μεταξύ περίπου 1022 πιθανών συνδυασμών, για να αποκωδικοποιήσει ένα μήνυμα της Enigma. Ο Turing ονόμασε τη μέθοδο αναζήτησης του ευρετική αναζήτηση, και όπως αναμένετο λειτούργησε με επιτυχία τις περισσότερες φορές, αλλά όχι πάντα. Σε κάθε περίπτωση ήταν μια τεράστια επιτυχία. Το 1945, ο Turing είχε προσληφθεί στο Εθνικό Εργαστήριο Φυσικής (NPL), στο Ηνωμένο Βασίλειο, όπου ξεκινά το σχέδιό του για τη Αυτόματη Υπολογιστική Μηχανή [Automatic Computing Engine (ACE)]. Σε μια έκθεση του NPL με τίτλο «Ευφυείς μηχανές» το 1948, περιέγραψε πρωτοποριακή του ιδέα για τη νοημοσύνη των μηχανών και τη μάθηση, τα νευρωνικά δίκτυα και τους εξελικτικούς αλγόριθμους ή μια πρόωμη έκδοση των γενετικών αλγορίθμων.

Οι δεκαετίες του 1980 και του 1990 ήταν η πιο συναρπαστική στιγμή για τους μεταευρετικούς αλγόριθμους. Το επόμενο μεγάλο βήμα είναι η ανάπτυξη της προσομοιωμένης ανόπτησης/ simulated annealing (SA) το 1983, μια τεχνική βελτιστοποίησης, η οποία ξεκίνησε από τον S. Kirkpatrick, τον C.D. Gellat και τον M.P. Vecchi, εμπνευσμένη από τη διαδικασία ανόπτησης των μετάλλων. Πρόκειται για αναζήτηση αλγόριθμου με βάση την τροχιά, ξεκινώντας με μια αρχική εικασία λύσης σε υψηλή θερμοκρασία, και σταδιακή ψύξη του συστήματος. Η πραγματική πρώτη χρήση των μεταευρετικών είναι πιθανόν να οφείλεται στον Fred Glover και στη δική του αναζήτηση Tabu το 1986, αν και το πρωτόλειο βιβλίο του σχετικά με την αναζήτηση Tabu δημοσιεύθηκε αργότερα, το 1997.

Αρκετοί Μεταευρετικοί αλγόριθμοι είναι εμπνευσμένοι από τη φύση όπως έχουν αναπτυχθεί με βάση κάποια στοιχεία της φύσης. Η Φύση έχει εξελιχθεί κατά τη διάρκεια εκατομμυρίων ετών και έχει βρει εκπληκτικές λύσεις για σχεδόν όλα τα προβλήματα που συνάντησε. Έτσι μπορούμε με αυτόν τον τρόπο να μάθουμε την επιτυχία της επίλυσης προβλημάτων από τη φύση και την ανάπτυξη αλγορίθμων ευρετικών ή μεταευρετικών εμπνευσμένους από τη φύση. Πιο συγκεκριμένα, μερικοί αλγόριθμοι που είναι εμπνευσμένοι από τη φύση είναι εμπνευσμένοι από την εξελικτική θεωρία του Δαρβίνου. Κατά συνέπεια, έχουν πει να είναι

βιολογία-εμπνευσμένο ή απλά bio-εμπνευσμένο. Ως εκ τούτου, λέγονται αλλιώς βίο-εμπνευσμένοι.

Δύο βασικά στοιχεία του κάθε Μεθευρετικού αλγόριθμου είναι: η επιλογή της καλύτερης λύσης και η τυχαιοποίηση. Η επιλογή της καλύτερης λύσης διασφαλίζει ότι οι λύσεις θα συγκλίνουν προς το βέλτιστο, ενώ με την τυχαιοποίηση αποφεύγουμε οι λύσεις να εγκλωβιστούν σε τοπικά βέλτιστα και, ταυτόχρονα, να αυξήσει την ποικιλομορφία των λύσεων. Ο καλός συνδυασμός αυτών των δύο στοιχείων εξασφαλίζει συνήθως ότι το ολικό βέλτιστο είναι εφικτό.

Οι Μεθευρετικοί αλγόριθμοι μπορούν να ταξινομηθούν με πολλούς τρόπους. Ένας τρόπος είναι να ταξινομηθούν:

- με βάση τον πληθυσμό και με βάση την τροχιά. Για παράδειγμα, οι γενετικοί αλγόριθμοι είναι με βάση τον πληθυσμό που χρησιμοποιούν ένα σύνολο συμβολοσειρών, έτσι είναι η βελτιστοποίηση σμήνους σωματιδίων (BSS), η οποία χρησιμοποιεί πολλούς πράκτορες ή σωματίδια. Η BSS είναι επίσης γνωστό ως αλγόριθμος βάσει πρακτόρων.
- Από την άλλη, η προσομοιωμένη ανόπτηση χρησιμοποιεί ένα μεμονωμένο παράγοντα ή λύση που κινείται μέσα από το χώρο σχεδίασης ή του χώρου αναζήτησης σε *piecewise* στυλ. Μια καλύτερη κίνηση ή λύση είναι πάντα αποδέκτες, ενώ μια όχι και τόσο καλή κίνηση μπορεί να είναι αποδεκτή με ορισμένη πιθανότητα. Τα βήματα ή οι κινήσεις εντοπίζουν μια τροχιά στον χώρο αναζήτησης, με μη μηδενική πιθανότητα ότι η πορεία αυτή μπορεί να φτάσει το ολικό βέλτιστο.

Οι μεταευρετικοί αλγόριθμοι κατηγοριοποιούνται ως εξής:

- Οι αλγόριθμοι που βασίζονται στην ευφυΐα του σμήνους (*swarm intelligence based algorithms*).
- Οι εξελικτικοί αλγόριθμοι (*evolutionary algorithms*)

2.2.3 Γενετικοί Αλγόριθμοι

Οι γενετικοί αλγόριθμοι (GA) είναι μια μέθοδος αναζήτησης με βάση τη θεωρία του Δαρβίνου περί της εξέλιξης και της φυσικής επιλογής των βιολογικών συστημάτων και την αντιπροσωπεύουν στους μαθηματικούς τελεστές: διασταύρωση ή ανασυνδυασμός, μετάλλαξη, καλή φυσική κατάσταση, και επιλογή του ισχυρότερου. Από τότε, οι γενετικοί

αλγόριθμοι είχαν τόσο μεγάλη επιτυχία στην επίλυση ενός ευρέος φάσματος προβλημάτων βελτιστοποίησης, ενώ χιλιάδες ερευνητικά άρθρα και εκατοντάδες βιβλία έχουν γραφτεί. Μερικές στατιστικές δείχνουν ότι η συντριπτική πλειοψηφία των εταιρειών του Fortune 500, τους χρησιμοποιούν τώρα καθημερινά για την επίλυση δύσκολων προβλημάτων συνδυαστικής βελτιστοποίησης, όπως στον σχεδιασμό, data-fitting και τον προγραμματισμό.

Κατά την ίδια περίοδο, ο Ingo Rechenberg και ο Hans-Paul Schwefel όντας και οι δυο στο Τεχνικό Πανεπιστήμιο του Βερολίνου, ανέπτυξαν το 1963 μια τεχνική έρευνας για επίλυση του προβλήματος βελτιστοποίησης στην αεροδιαστημική μηχανική, που ονομάζεται στρατηγική εξέλιξης. Αργότερα, ο Bienert συνεργάστηκε μαζί τους και άρχισε να κατασκευάζει ένα αυτόματο πειραματιστή (automatic experimenter) με τη χρήση απλών κανόνων της μετάλλαξης και επιλογής. Δεν υπάρχει διασταύρωση σε αυτή την τεχνική. Χρησιμοποιήθηκε μόνο μετάλλαξη για την παραγωγή ενός κλάδου (offspring) και διατηρήθηκε μια βελτιωμένη λύση σε κάθε γενεά.

Πρόκειται ουσιαστικά για ένα απλό αλγόριθμο με βάση την πορεία hill-climbing στίλ με τυχαιοποίηση. Ήδη το 1960, ο Lawrence J. Fogel σκόπευε να χρησιμοποιήσει προσομοίωση εξέλιξης ως διαδικασία μάθησης και ως εργαλείο για τη μελέτη της τεχνητής νοημοσύνης. Στη συνέχεια, το 1966, ο L. J. Fogel με τον A. J. Owen και τον M. J. Walsh, ανέπτυξαν την τεχνική του εξελικτικού προγραμματισμού εκπροσωπώντας λύσεις όπως μηχανήματα πεπερασμένων καταστάσεων και μεταλλάσσοντας τυχαία ένα από αυτά τα μηχανήματα. Οι παραπάνω καινοτόμες ιδέες και μέθοδοι έχουν εξελιχθεί σε ένα πολύ ευρύτερο επιστημονικό τομέα, που ονομάζεται εξελικτικοί αλγόριθμοι και εξελικτική υπολογιστική.



Εικόνα 2.2.3: Η βασική λειτουργική δομή ενός τυπικού ΓΑ

ΚΕΦΑΛΑΙΟ 3

Νοημοσύνη Σμήνους

3.1 Εισαγωγή

Με τον όρο Νοημοσύνη Σκηνών (ΝΣ) (swarm intelligence) στην Επιστήμη των Υπολογιστών χαρακτηρίζουμε όλους εκείνους τους αλγορίθμους και τις καταναμημένες μεθόδους επίλυσης προβλημάτων που έχουν σαν πηγή έμπνευσης την συλλογική συμπεριφορά (collective behavior) και την εκδηλούμενη νοημοσύνη (emergent intelligence) που εμφανίζεται σε πληθυσμούς. Ο όρος αυτός χρησιμοποιήθηκε αρχικά σε τεχνολογικές εφαρμογές για να χαρακτηρίσει την συμπεριφορά ομαδοποιημένων αυτόματων μηχανισμών (robots), τα οποία υπακούοντας καθένα από αυτά σε κάποιους απλούς κανόνες, εμφάνιζαν με το πέρασμα των χρόνων κάποιου είδους συλλογική συμπεριφορά. Αυτές οι συμπεριφορές είχαν παρατηρηθεί πιο νωρίς και στην φύση, δηλαδή σε κοινωνικές ομάδες εντόμων και ζώων καθώς επίσης και σε ανθρώπινες ομάδες. Επομένως μπορούμε να πούμε ότι ο όρος νοημοσύνη σμηνών χρησιμοποιούνταν και από κοινωνιολογικής άποψης με σκοπό να καταδείξει το αποτέλεσμα της αλληλεπίδρασης μεταξύ διαφόρων οντοτήτων, από την πλευρά της ομάδας ή και του σμήνους. Αξιοσημείωτο παράδειγμα αποτελούν τα σμήνη των μυρμηγκιών όπου ενώ το κάθε μυρμηγκι λειτουργώντας σαν μονάδα εμφανίζει ένα περιορισμένο αριθμό δυνατοτήτων αλλά και αντιδράσεων στο περιβάλλον του, οι ομάδες των μυρμηγκιών αντιθέτως παρουσιάζουν μια αξιοθαύμαστη οργάνωση και δομή. Η ΝΣ αποτελεί μια γενική έννοια που χρησιμοποιείται ακόμη και σημειολογικά για τον χαρακτηρισμό των ιδεών που διέπουν την ερμηνεία συμβόλων και συμβολισμών κατά την επικοινωνία δύο ατόκων. Ο πρώτος αλγόριθμος ΝΣ που εμφανίστηκε ήταν η Βελτιστοποίηση με Αποικία Μυρμηγκιών (BAM) (Ant Colony Optimization), ο οποίος μιμείται τον τρόπο με τον οποίο οι τερμίτες αναζητούν τροφή επιλέγοντας την συντομότερη διαδρομή μεταξύ της πηγής τροφής και της βάσης τους. Ο αλγόριθμος δοκιμάστηκε σε πολλά προβλήματα συνδυαστικής βελτιστοποίησης με εξαιρετικά αποτελέσματα και μέχρι σήμερα έχει χρησιμοποιηθεί και εφαρμοστεί στα πλαίσια του πραγματικού κόσμου, όπως η οργάνωση δρομολογίων αεροσκαφών και η δρομολόγηση τηλεπικοινωνιακών δορυφόρων (εφαρμόζεται στους δορυφόρους του τηλεπικοινωνιακού κολοσσού France Telecom). Επίσης, έχει αποτελέσει την αφετηρία για το πρόγραμμα ANTS: Autonomous Nano-Technology Swarm της NASA (National Aeronautics and Space Administration, USA) με στόχο την εξερεύνηση ζωνών

αστεροειδών με μικρά, μη-επανδρωμένα διαστημικά σκάφη τα οποία διαθέτουν την ικανότητα αυτό-οργάνωσης και κατανεμημένης, αυτόνομης λειτουργίας και λήψης αποφάσεων σε ομάδες με ιεραρχία (περισσότερες πληροφορίες υπάρχουν διαθέσιμες στην διεύθυνση του διαδικτύου της NASA, <http://ants.gsfc.nasa.gov/>). Ωστόσο, στις εφαρμογές της BAM υπάρχει ένα κενό στα προβλήματα αριθμητικής βελτιστοποίησης και αυτός είναι ο λόγος που δεν επεκτείνονται στα προβλήματα αυτά. Το κενό αυτό έρχεται να καλύψει η μέθοδος βελτιστοποίησης με σμήνος σωματιδίων, η οποία αποτέλεσε την πρώτη μέθοδο ΝΣ για τέτοιου είδους προβλήματα. Παρόλο που η φιλοσοφία πίσω από την βελτιστοποίηση σμήνους σωματιδίων(ΒΣΣ) διαφέρει από αυτή των εξελικτικών αλγορίθμων, αρχικά κατατάσσονταν στην ίδια κατηγορία λόγω των ομοιοτήτων που εμφάνιζε στην λειτουργία της με τους εξελικτικούς αλγόριθμους, όπως για παράδειγμα η στοχαστικότητα, η χρήση πληθυσμών, η ανάγκη για γνώση μονάχα της συναρτησιακής τιμής κλπ. Αργότερα, όταν η ΝΣ εμφανίστηκε σαν αυτόνομη κατηγορία αλγορίθμων, η ΒΣΣ αποτέλεσε το κύριο εργαλείο αυτής της κατηγορίας αλγορίθμων για την αντιμετώπιση προβλημάτων αριθμητικής βελτιστοποίησης.



Εικόνα 3.1: Ένα σμήνος πουλιών

3.2 Χαρακτηριστικά Της Νοημοσύνης Των Σμηνών

Τα πιο σημαντικά θετικά χαρακτηριστικά της εφαρμογής μιας προσέγγισης που βασίζεται στη νοημοσύνη των σμηνών, αναφορικά με τον έλεγχο μιας ομάδας ρομποτικών διαμεσολαβητών είναι τα εξής:

- **Επεκτασιμότητα (scalability)**, που αφορά την διατήρησης της δομής ελέγχου της ομάδας, η οποία αποτελείται από λίγες είτε από εκατοντάδες μονάδες. Παρατηρήθηκε ότι τα κοινωνικά έντομα λειτουργούν συλλογικά για ένα μεγάλο εύρος πληθυσμών ομάδων και συμπεραίνεται ότι ο μηχανισμός συντονισμού και συνεργασίας λειτουργεί ανεξάρτητα του αριθμού ατόμων στην ομάδα.
- **Ευελιξία (flexibility)**, υπό την έννοια ότι μονάδες μπορούν να προστεθούν ή να αφαιρεθούν δυναμικά, ενώ παράλληλα υπάρχει η δυνατότητα αναδιοργάνωσης ή ανακατανομής των ατόμων της ομάδας με κάποια σχετική μορφή αυτό οργάνωσης .
- **Ευρωστία (robustness)**, υπό την έννοια ότι το τελικό συλλογικό σύστημα είναι εύρωστο, όχι μόνο στη μείωση των μονάδων (σφάλματα κλπ.) αλλά και στην ελαχιστοποίηση στη σχεδίαση της μονάδας (minimalistic design).
- **Αποκέντρωση (Decentralization)**, εννοούμε ότι οι υπομονάδες λειτουργούν αυτόνομα (δεν υπάρχει κεντρικό σύστημα ελέγχου), ενώ το ίδιο παράλληλα ισχύει και για το περιβάλλον.
- **Αυτό-οργάνωση (Self-Organization)**, με την έννοια της εύρεσης (μονοπατιού) λύσης η οποία προκύπτει και δεν καθορίζεται εκ των προτέρων.
- **Απλότητα (Simplicity)**, δηλαδή οι διαμεσολαβητές σχεδιάζονται, προγραμματίζονται και ελέγχονται (σε ατομικό επίπεδο) πολύ εύκολα, έχουν την δυνατότητα να εφαρμοστούν σε πολύ μικρές πλατφόρμες (όπως για παράδειγμα δίκτυα smart dust) και τέλος να εκπαιδευτούν με χρήση γενετικών αλγορίθμων ή particle-swarm μεθόδων σε αντίθεση με άλλους αναλυτικούς αλγορίθμους μάθησης.

- Ενοποίηση με το περιβάλλον (environmental integration), εννοούμε την δυνατότητα χρήσης των δυναμικών του περιβάλλοντος για τον έλεγχο του συστήματος (στιγματική συμπεριφορά). Το επίπεδο οργάνωσης ενός συστήματος είναι αντιστρόφως ανάλογο της συμμετρίας του συστήματος. Ο θόρυβος αποτελεί το μεγαλύτερο μειονέκτημα όταν πρόκειται για ένα συμβατικό σύστημα, ενώ ένα στιγματικό σύστημα τον εκμεταλλεύεται σαν ένα φυσικό τρόπο, προκειμένου να σπάσει τις συμμετρίες των οντοτήτων και να τους δώσει την δυνατότητα να αυτόοργανωθούν.

Τα πιο σημαντικά αρνητικά γνωρίσματα της νοημοσύνης των σμηγών είναι τα εξής:

- Συμπεριφορά: Δεν μπορούμε να εξάγουμε συμπεράσματα από την ατομική συμπεριφορά με βάση τη συλλογική συμπεριφορά. Απλά παρατηρώντας μεμονωμένους διαμεσολαβητές δεν μπορεί κανείς να καταλήξει σε ασφαλή συμπεράσματα.
- Γνώση: η έλλειψη καθολικής γνώσης για το περιβάλλον αλλά και τη λειτουργία της ομάδας μπορεί να θεωρηθεί αρνητικός παράγοντας σε κάποιες περιπτώσεις.
- Σχεδιασμός: δεν υπάρχουν (παρά ελάχιστοι και συγκεκριμένοι) αναλυτικοί μηχανισμοί – μεθοδολογίες για το σχεδιασμό τέτοιων συστημάτων.
- Δράση: Η συμπεριφορά και οι δράσεις του ατόμου μοιάζουν με θόρυβο, ειδικά όταν η επιλογή δράσης είναι στοχαστική.
- Παράμετροι: Το σύστημα είναι εξαιρετικά ευαίσθητο στις παραμέτρους λειτουργίας του (π.Χ. διαφορετική ακτίνα αντίχενωσης εμποδίων), κάτι που σημαίνει ότι η αλλαγή μιας παραμέτρου μπορεί να επιφέρει μεγάλες αλλαγές στην εμφάνιση(ή όχι) συλλογικών συμπεριφορών. Το γεγονός αυτό συνεπάγεται εξονυχιστικό έλεγχο της συμπεριφοράς του συστήματος για διαφορετικά σύνολα παραμέτρων.

ΚΕΦΑΛΑΙΟ 4

Αλγόριθμοι Νοημοσύνης Σμηνών

Οι αλγόριθμοι που βασίζονται στην ευφυΐα του σμήνους έχουν εμπνευστεί από την φύση. Η ευφυΐα του σμήνους είναι ένα πρότυπο εμπνευσμένο από την φύση και πιο συγκεκριμένα από την συμπεριφορά διαφόρων κοινωνικών οργανισμών όπως είναι τα μυρμήγκια, οι μέλισσες, τα ψάρια, τα πουλιά. Μιμείται τον τρόπο ζωής αυτών των οργανισμών αφού αυτός μπορεί να μας δώσει λύσεις σε διάφορα υπολογιστικά προβλήματα. Οι διάφορες κοινωνίες οργανισμών συνεργάζονται μεταξύ τους για την ολοκλήρωση κοινών σκοπών, όπως είναι η συλλογή της τροφής ή η κατασκευή της φωλιάς τους. Η ευφυΐα του σμήνους αποτελεί συχνό αντικείμενο έρευνας τα τελευταία χρόνια. Ο όρος “σμήνος” χρησιμοποιείται πιο γενικά για την αναφορά σε οποιοδήποτε περιορισμένο σύνολο αλληλοεπιδρώντων πρακτόρων ή ατόμων. Το πιο κλασικό παράδειγμα είναι αυτό των μελισσών, ωστόσο μπορεί να γενικευθεί εύκολα και σε άλλα συστήματα παρόμοιας αρχιτεκτονικής, όπως π.χ. ένα ‘ανοσοποιητικό σύστημα’ (De Castro, Von Zuben, 1999).

4.1 Αλγόριθμος Της Πυγολαμπίδας

Η συμπεριφορά των πυγολαμπίδων όταν έλκονται ή όταν εκπέμπουν λάμπες ενέπνευσε τον Yang το 2007 για την δημιουργία του αλγορίθμου της πυγολαμπίδας. Ως μέσο επικοινωνίας για να προσελκύσουν άλλες πυγολαμπίδες, χρησιμοποιούν τη λάμψη τους. Ο αλγόριθμος της νυχτερίδας (Firefly Algorithm – FA) ακολουθεί τρεις κανόνες:

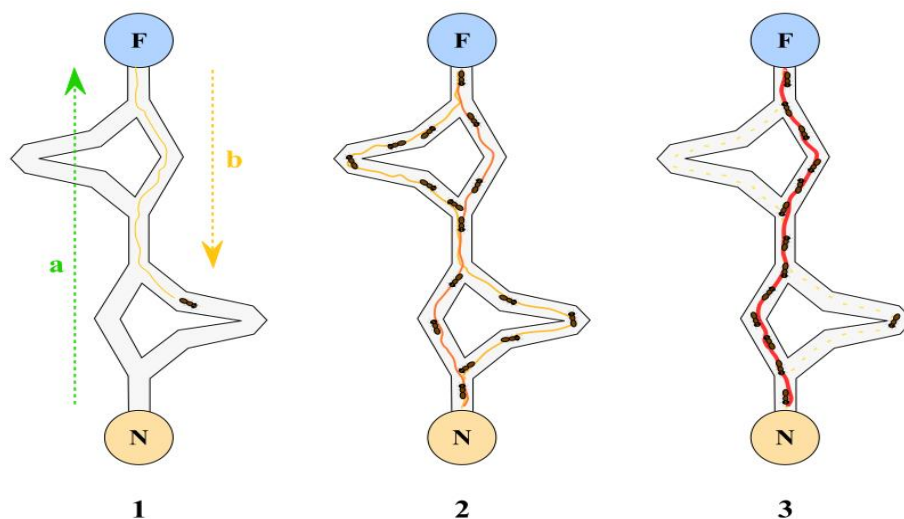
- Οι πυγολαμπίδες είναι μονογενείς, δεν είναι απαραίτητο να μπορεί να προσελκύσει οποιαδήποτε άλλη χωρίς να εξετάζεται το φύλο τους.
- Η ελκυστικότητα των πυγολαμπίδων είναι ανάλογη με τη φωτεινότητά τους. Η φωτεινότητα μιας πυγολαμπίδας δεν μειώνεται εφόσον παραμένουν σε κοντινή απόσταση. Η πυγολαμπίδα κινείται προς μία πιο φωτεινή πυγολαμπίδα. Αν δεν βρει φωτεινότερη, η πυγολαμπίδα θα κινηθεί προς τυχαία πορεία.
- Μία πυγολαμπίδα δεν μπορεί να προσελκυστεί από μία άλλη που δεν είναι τόσο φωτεινή όσο αυτή.

Η αντικειμενική συνάρτηση ενός προβλήματος σχετίζεται κυρίως από την φωτεινότητα μιας πυγολαμπίδας. Η φωτεινότητα είναι ανάλογη με την τιμή της αντικειμενικής συνάρτησης σε

ένα πρόβλημα μεγιστοποίησης. Μπορούν να προσδιοριστούν και άλλα είδη φωτεινότητας παρόμοια με τη συνάρτηση καταλληλότητας των γενετικών αλγορίθμων. Για τον υπολογισμό της απόστασης μεταξύ των πυγολαμπίδων μπορεί να χρησιμοποιηθεί η καρτεσιανή απόσταση. Επίσης διαμορφώνεται μία συνάρτηση ελκυστικότητας (attractiveness function) από τη φωτεινότητα, την απόσταση κι έναν παράγοντα απορρόφησης της λάμψης. Η ελκυστικότητα, η τρέχουσα θέση και μία τυχαία διαδρομή συνθέτουν τη συνάρτηση κίνησης (movement function) για μία πυγολαμπίδα.

4.2 Αλγόριθμος Ant Colony Optimization (ACO)

Η βελτιστοποίηση αποικίας μυρμηγκιών - Ant Colony Optimization είναι μια μεθευρετική μέθοδος οποία βασίζεται στον πληθυσμό που μπορεί να χρησιμοποιηθεί προκειμένου να επιλύουν κατά προσέγγιση δύσκολα προβλήματα βελτιστοποίησης. Εμπνεύστηκε από την συμπεριφορά που αναζητούν τροφή τα μυρμήγκια στις αποικίες τους. Στην παρακάτω εικόνα περιγράφεται ο τρόπος με τον οποίο βρίσκουν την τροφή τους οι αποικίες μυρμηγκιών. Στο 1 το πρώτο μυρμήγκι βρίσκει την τροφή ακολουθώντας το μονοπάτι και επιστρέφει στη φωλιά από το μονοπάτι αφήνοντας πίσω του φερομόνες. Στο 2 τα υπόλοιπα μυρμήγκια ακολουθούν ένα από τα 4 δυνατά μονοπάτια αλλά το συντομότερο μονοπάτι ενισχύεται περισσότερο από φερομόνες και έτσι γίνεται πιο ελκυστικό. Στο 3 τα μυρμήγκια ακολουθούν το συντομότερο μονοπάτι ενώ οι φερομόνες στα υπόλοιπα εξασθενούν.



Εικόνα 4.2: 1) Το πρώτο μυρμήγκι επιλέγει μια διαδρομή προς την τροφή (F) και επιστρέφει στη φωλιά (N) αφήνοντας ίχνη φερομόνης 2) τα άλλα μυρμήγκια ακολουθούν κάποια από τις τέσσερις διαδρομές 3) η διαδρομή με την μεγαλύτερη εναπόθεση φερομόνης, που είναι και η συντομότερη, γίνεται προτιμητέα.

Στην βελτιστοποίηση αποικίας μυρμηγκιών (ACO- Ant Colony Optimization), ένα σύνολο παραγόντων λογισμικού που ονομάζονται «τεχνητά μυρμηγκία» αναζητούν τις καλύτερες λύσεις για ένα πρόβλημα βελτιστοποίησης το οποίο έχει μετατραπεί σε πρόβλημα εύρεσης του ελάχιστου κόστους ενός μονοπατιού σε ένα σταθμισμένο γράφημα. Σταδιακά τα μυρμηγκία βρίσκουν λύσεις με την κίνησή τους στο γράφημα. Η διαδικασία αυτή είναι μία στοχαστική επίλυση και επηρεάζεται από ένα μοντέλο φερομόνης, δηλαδή ένα σύνολο παραμέτρων που σχετίζονται με τα χαρακτηριστικά του γραφήματος (κόμβους ή ακμές), οι τιμές των οποίων με την πάροδο του χρόνου εκτέλεσης του αλγορίθμου αλλάζουν από τα τεχνητά μυρμηγκία.

Η ACO έχει εφαρμοστεί με επιτυχία σε πολλά κλασικά προβλήματα συνδυαστικής βελτιστοποίησης, όπως και σε προβλήματα διακεκριμένης βελτιστοποίησης που έχουν στοχαστικές ή και δυναμικές συνιστώσες. Επιπλέον, η ACO έχει επεκταθεί έτσι ώστε να μπορεί να χρησιμοποιηθεί για την επίλυση προβλημάτων βελτιστοποίησης συνεχών και μικτών μεταβλητών. Η βελτιστοποίηση αποικίας μυρμηγκιών είναι ίσως ένα από τα πιο επιτυχημένα παραδείγματα πληροφοριακού συστήματος νοημοσύνης σμήνους με πολλές εφαρμογές σε προβλήματα του πραγματικού κόσμου.

4.3 Αλγόριθμος Τεχνητής Αποικίας Μελισσών

Ο αλγόριθμος των μελισσών (Bee Algorithms) είναι μία κατηγορία μεταερευτικών μεθόδων που είναι βασισμένες στον τρόπο που οι μέλισσες αναζητούν την τροφή τους. Το 1996 ο Yonezawa και ο Kikuchi και το 1997 ο Sato και ο Hagiwara ασχολήθηκαν με τεχνικές που περιέχουν τα στοιχεία της συμπεριφοράς των μελισσών. Το 2004 ο Nakrani και ο Tovey ανέπτυξαν τον αλγόριθμο των μελιτοφόρων μελισσών ως ένα ξεχωριστό πεδίο μεταερευτικών αλγορίθμων. Μέχρι και σήμερα έχουν αναπτυχθεί πολλοί αλγόριθμοι των μελισσών μεταξύ των οποίων και η αποικία των τεχνητών μελισσών (Artificial Bee Colony-ABC).

4.4 Αλγόριθμος Βελτιστοποίησης Ζευγαρώματος Μελισσών

Η διαδικασία ζευγαρώματος στην κυψέλη από την βασίλισσα των μελισσών και η ευφυΐα ενός σμήνους είναι τα βασικά κριτήρια του αλγορίθμου Βελτιστοποίησης Ζευγαρώματος Μελισσών. Η μέθοδος αρχίζει όταν η βασίλισσα αποφασίζει να απομακρυνθεί από τη φωλιά και να πετάξει, πράττοντας την τυχαία πτήση ζευγαρώματος, ακολουθούμενη από το σμήνος

των κηφών και ζευγαρώνουν με αυτή στον αέρα. Η βασίλισσα επιλέγει με βάση την ταχύτητα της, τον κηφήνα με τον οποίο θα ζευγαρώσει.

4.5 Αλγόριθμος Της Νυχτερίδας

Ο αλγόριθμος της νυχτερίδας (Bat Algorithm – BA) είναι ένας από τους πιο πρόσφατους μεταερευνητικούς αλγορίθμους και αναπτύχθηκε από τον Xin-She Yang. Έχει επηρεαστεί από το σύστημα ηχοεντοπισμού που χρησιμοποιούν οι νυχτερίδες κατά την αναζήτηση της τροφής τους. Έχει αναδειχθεί σε έναν ιδιαίτερα αποδοτικό πληθυσμιακό αλγόριθμο νοημοσύνης σμήνους με την απόδοσή του να μπορεί να αυξάνεται περαιτέρω μετά από κατάλληλη τροποποίηση ορισμένων παραμέτρων ή ένταξη στοιχείων από άλλους μεταερευνητικούς αλγορίθμους.

4.6 Αλγόριθμος Βελτιστοποίησης Σμήνους Σωματιδίων (Particle Swarm Optimization - PSO)

Ο Αλγόριθμος Βελτιστοποίησης Σμήνους Σωματιδίων (Particle Swarm Optimization - PSO) είναι ένας απ' τους πιο σημαντικούς αλγορίθμους που προτάθηκε από τους Kennedy και Eberhard προκειμένου να προσομοιωθεί η κοινωνική συμπεριφορά κάποιων ομάδων όπως το πέταγμα των πουλιών σε μορφή σμήνους και η ομαδική κίνηση των ψαριών.

Η αρχική ιδέα ήταν να προσομοιωθεί γραφικά η κίνηση την οποία κάνει ένα σμήνος πουλιών κατά πρώτο λόγο, κατά δεύτερο λόγο να βρεθούν και να αναλυθούν οι κανόνες που οδηγούν τα πουλιά να κινούνται ως ένα σμήνος καθώς επίσης το γιατί αλλάζουν ξαφνικά κατεύθυνση τα πουλιά χωρίς όμως να καταστρέφεται ο σχηματισμός τον οποίον είχαν.

Ο αλγόριθμος χρησιμοποιεί τη φυσική κίνηση των ατόμων ενός πληθυσμού μέσα στο σμήνος και έχει πολύ ευέλικτο και καλά ισορροπημένο μηχανισμό που προσαρμόζεται στις ικανότητες εξερεύνησης των ατόμων του σμήνους. Οι αλλαγές ενός ατόμου μέσα στο σμήνος επηρεάζονται από την εμπειρία και τις γνώσεις των γειτονικών του ατόμων και ως εκ τούτου ο αλγόριθμος βελτιστοποίησης σμήνους σωματιδίων μπορεί να θεωρηθεί ως ένας συνεργατικός αλγόριθμος.

Έχει αποδειχτεί πως λόγω της εύκολης υλοποίησης της η συγκεκριμένη μέθοδος καθίσταται πολύ χρήσιμη σε ένα μεγάλο φάσμα προβλημάτων αλλά και λόγω των πολύ καλών αποτελεσμάτων που έχει δώσει σε προβλήματα που έχουν συνεχείς μεταβλητές, καθώς και τώρα τελευταία και με εφαρμογές σε προβλήματα συνδυαστικής βελτιστοποίησης.

Σε σύγκριση με τους γενετικούς αλγόριθμους, ο PSO εμπεριέχει μνήμη, πράγμα που σημαίνει πως η γνώση που προκύπτει από προηγούμενες καλές λύσεις κληροδοτείται στις επόμενες γενιές και δεν χάνεται από τις διαρκείς επαναλήψεις. Επίσης διαπιστώνεται μια καλή συνεργασία μεταξύ των σωματιδίων του σμήνους αφού μεταξύ των μελών της ομάδας υπάρχει άριστη συνεργασία στην κατασκευή των λύσεων.

Για την δημιουργία του αρχικού σμήνους, δημιουργείται ένας αρχικός αριθμός από σωματίδια που το κάθε ένα από τα οποία έχει μια συγκεκριμένη θέση μέσα στο χώρο λύσεων και κινείται επίσης με συγκεκριμένη ταχύτητα. Αντιστοιχίζοντας με έναν γενετικό αλγόριθμο θα διαπιστώναμε πως το σμήνος (swarm) αντιστοιχεί στον πληθυσμό (population) και το σωματίδιο (particle) αντιστοιχεί σε ένα άτομο (individual).

Οι θέσεις των σωματιδίων αντιπροσωπεύουν τις λύσεις του προβλήματος. Δεδομένου αυτού, αν N είναι το μέγεθος του πληθυσμού και n ο αριθμός των διαστάσεων, x_{ij} θα είναι οι λύσεις με $i=1,2,\dots,N$ και $j=1,2,\dots,n$. Η απόδοση της θέσης-λύσης εκτιμάται από την προκαθορισμένη συνάρτηση καταλληλότητας $f(x_{ij})$ (fitness function). Επίσης η ταχύτητα u_{ij} αντιπροσωπεύει τις αλλαγές που θα γίνουν έτσι ώστε να κινηθεί το σωματίδιο από μια θέση σε μια άλλη. Η κατεύθυνση που θα κινηθεί το σωματίδιο υπολογίζεται από την δυναμική αλληλεπίδραση της δικής του εμπειρίας αλλά και της εμπειρίας ολόκληρου του σμήνους.

Κατά την κίνηση τους τα σωματίδια, έχουν τρεις επιλογές, ή να επιλέξουν να ακολουθήσουν μια δική τους διαδρομή, είτε να κινηθούν προς την βέλτιστη θέση που είχαν κατά τη διάρκεια των επαναλήψεων $pbest_{ij}$, είτε να κινηθούν προς την θέση που έχει το βέλτιστο σωματίδιο στον πληθυσμό ($gbest_j$)

Οι ταχύτητες και οι θέσεις των σωματιδίων όταν έχουμε συνεχείς μεταβλητές, υπολογίζονται βάση των παρακάτω εξισώσεων:

$$u_{ij}(t+1) = u_{ij}(t) + c_1 \text{rand}_1(pbest_{ij} - x_{ij}(t)) + c_2 \text{rand}_2(gbest_j - x_{ij}(t))$$

$$x_{ij}(t+1) = x_{ij}(t) + u_{ij}(t+1)$$

Όπου t είναι ο μετρητής των επαναλήψεων, c_1 και c_2 είναι οι μεταβλητές επιτάχυνσης, rand_1 και rand_2 είναι δύο τυχαίες μεταβλητές στο διάστημα $[0,1]$. Οι μεταβλητές επιτάχυνσης ελέγχουν το πόσο μακριά μπορεί να κινηθεί το σωματίδιο κατά τη διάρκεια μιας απλής επανάληψης. Χαμηλές τιμές επιτρέπουν στα σωματίδια να βρεθούν πολύ μακριά από τη στοχευμένη περιοχή πριν βρεθούν κοντά σε τοπικό ελάχιστον, ενώ πολύ υψηλές τιμές έχουν

ως αποτέλεσμα να κάνουν μια απότομη κίνηση προς τις στοχευμένες περιοχές. Τυπικά και οι δύο αυτές μεταβλητές παίρνουν την τιμή 2. Αυτός ο αλγόριθμος ονομάζεται και gbest – PSO γιατί η γειτονιά αναζήτησης που έχουν τα σωματίδια αντιστοιχεί σε ολόκληρο το σμήνος.

Η βέλτιστη θέση $pbest_{ij}$ ενός σωματιδίου στο σμήνος υπολογίζεται από την ακόλουθη εξίσωση:

Σε περίπτωση προβλήματος ελαχιστοποίησης:

$$pbest_{ij} \{x_{ij}(t+1), \text{ εάν } f(x_{ij}(t+1)) < f(x_{ij}(t)) pbest_{ij},$$

αλλιώς σε περίπτωση προβλήματος μεγιστοποίησης:

$$pbest_{ij} \{x_{ij}(t+1), \text{ εάν } f(x_{ij}(t+1)) > f(x_{ij}(t)) pbest_{ij},$$

αλλιώς η βέλτιστη θέση όλου του σμήνους τη χρονική στιγμή t υπολογίζεται από την εξίσωση:

$$gbest_j \in \{pbest_{1j}, pbest_{2j}, \dots, pbest_{Nj} | f(gbest_j)\} = \min\{f(pbest_{1j}), f(pbest_{2j}), \dots,$$

$$f(pbest_{Nj})\}$$

Ο υπολογισμός της ταχύτητας σε μια χρονική στιγμή $t+1$ αποτελείται από τρεις όρους. Αρχικά από την προηγούμενη ταχύτητα ($u_{ij}(t)$) που στην ουσία λειτουργεί ως μνήμη των προηγούμενων κατευθύνσεων που είχε χρησιμοποιήσει το σωματίδιο κατά την αμέσως προηγούμενη χρονική στιγμή. Αυτή η μνήμη εμποδίζει το σωματίδιο να αλλάξει δραστικά την κατεύθυνσή του κατά τη διάρκεια μιας επανάληψης.

Ο δεύτερος όρος ($c1rand1(pbest_{ij} - x_{ij}(t))$) ο οποίος αναφέρεται και ως γνωστικός όρος (cognitive component) χρησιμοποιείται για να συγκρίνει την απόδοση κάθε σωματιδίου σε σχέση με τις αποδόσεις του κατά το παρελθόν. Στην ουσία αυτός ο όρος έχει το ρόλο της επαναφοράς του σωματιδίου σε προηγούμενες καταστάσεις που ήταν πολύ αποδοτικές για το ίδιο το σωματίδιο.

Ο τρίτος όρος ($c2rand2(gbest_j - x_{ij}(t))$) ο οποίος αναφέρεται και ως κοινωνικός όρος (social component) χρησιμοποιείται για να συγκρίνει την απόδοση κάθε σωματιδίου σε σχέση με τις αποδόσεις ολόκληρου του σμήνους ή μιας γειτονιάς σωματιδίων. Στην ουσία αυτός ο όρος έχει το ρόλο της αναζήτησης του καλύτερου μέλους του σμήνους από το σωματίδιο .

Όταν μεταφέρουμε στην αρχική τους κατάσταση τα σωματίδια, είναι πολύ πιθανό να αφήσουμε κάποια μέρη του χώρου λύσεων ακάλυπτα και αυτό μπορεί να οδηγήσει είτε σε αργή σύγκλιση προς το βέλτιστο είτε να οδηγηθεί ο αλγόριθμος σε τοπικό ελάχιστο. Ο συνηθέστερος τρόπος είναι να αρχικοποιήσουμε τις θέσεις των σωματιδίων με τυχαίο τρόπο στο χώρο λύσεων. Έτσι, ανάλογα με το πρόβλημα που θέλουμε να επιλύσουμε πρέπει να προσπαθούμε να αρχικοποιούμε κατάλληλα τον πληθυσμό ώστε να καλύπτουμε όσο το δυνατόν μεγαλύτερο μέρος του χώρου αναζήτησης. Αν θέσουμε ως x_{min} την ελάχιστη τιμή και x_{max} την μέγιστη, τότε όλες οι τιμές του διανύσματος του κάθε σωματιδίου μπορούν να υπολογιστούν από την σχέση:

$$x_{ij}(0) = x_{min} + r_i(x_{max} - x_{min})$$

Με το r_i να είναι ένας τυχαίος αριθμός στο $(0,1)$.

Αντίθετα με τις αρχικές θέσεις των σωματιδίων που είναι πολύ σημαντικό να οριστούν σωστά, οι αρχικές ταχύτητες σχεδόν πάντα παίρνουν την τιμή 0. Οι αρχικές τιμές του p_{best} για κάθε σωματίδιο τίθενται ίσες με τις αρχικές θέσεις των σωματιδίων. Ο τερματισμός του αλγορίθμου επέρχεται όταν ένα από τα κλασσικά κριτήρια που ισχύουν σε όλους τους εξελικτικούς αλγορίθμους ικανοποιηθεί. Δηλαδή είτε αν έχουμε φθάσει το μέγιστο αριθμό επαναλήψεων, είτε αν η βέλτιστη λύση βρίσκεται μέσα σε κάποιο προκαθορισμένο κατώφλι, είτε αν μετά από ένα αριθμό επαναλήψεων δεν έχει εμφανιστεί κάποια βελτίωση στη λύση κτλ.

Οι παράγοντες επιτάχυνσης c_1 και c_2 μαζί με τα τυχαία διανύσματα $rand_1$ και $rand_2$ ελέγχουν την τυχαία επιρροή του γνωστικού και κοινωνικού παράγοντα στη συνολική ταχύτητα του σωματιδίου. Αν το $c_1 = c_2 = 0$ τα σωματίδια πετούν στην κατεύθυνση που τους δίνει η ταχύτητα τους. Εάν το $c_1 > 0$ και το $c_2 = 0$ τότε το κάθε μέλος του πληθυσμού επηρεάζεται μόνο από τις προηγούμενες κινήσεις του και κινείται ανεξάρτητα από τα άλλα σωματίδια του σμήνους.

Στην αντίθετη περίπτωση όπου $c_2 > 0$ και το $c_1 = 0$ ολόκληρο το σμήνος κυνηγάει ένα σωματίδιο, το βέλτιστο. Ο στόχος είναι να βρεθεί η κατάλληλη ισορροπία μεταξύ του c_1 και του c_2 . Συνήθως το $c_1 = c_2$, όπου σημαίνει ότι το σωματίδιο το έλκουν και οι δύο παράγοντες ισόποσα. Για παράδειγμα, αν θεωρήσουμε ότι το $c_{1,min}$, $c_{1,max}$, $c_{2,min}$, $c_{2,max}$ είναι οι μέγιστες και οι ελάχιστες τιμές που θα μπορούσαν να πάρουν τα c_1 και c_2 αντίστοιχα, τότε:

$$c_1 = c_{1,min} + c_{1,max} - c_{1,min} \cdot iter_{max} \cdot t$$

$$c2 = c2,min + c2,max - c2,min \text{ itermax} * t$$

Όπου t είναι ο αριθμός της τρέχουσας επανάληψης και itermax είναι ο μέγιστος αριθμός επαναλήψεων. Με αυτό τον τρόπο αρχικά οι τιμές των $c1$ και $c2$ είναι μικρές εν συνεχεία καθώς προχωρούν οι επαναλήψεις οι τιμές αυξάνονται και πλησιάζουν τη μέγιστη τιμή. Άρα αρχικά τα σωματίδια επηρεάζονται περισσότερο από τον πρώτο παράγοντα της εξίσωσης και κατόπιν αρχίζουν να επηρεάζονται από το δεύτερο και τον τρίτο παράγοντα. Αυτό γίνεται με σκοπό να υπάρχει μεγαλύτερη ελευθέρια κίνηση μέσα στο χώρο των σωματιδίων κατά τις πρώτες επαναλήψεις ώστε να γίνει ο εντοπισμός του ελάχιστου το γρηγορότερο δυνατό.

Σε μορφή ψευδοκώδικα ο αλγόριθμος βελτιστοποίησης σμήνους σωματιδίων θα έχει ως εξής:

Αλγόριθμος Βελτιστοποίησης Σμήνους Σωματιδίων

1. Αρχικοποίηση
2. Επιλογή του αριθμού των σμηνών
3. Επιλογή του αριθμού των σωματιδίων σε κάθε σμήνος
4. Αρχικοποίηση της θέσης και της ταχύτητας κάθε σωματιδίου
5. Υπολογισμός του αρχικού κόστους του κάθε σωματιδίου
6. Εύρεση Βέλτιστου σωματιδίου ολόκληρου του σμήνους
7. Εύρεση Βέλτιστης λύσης κάθε σωματιδίου
8. Κύρια φάση Εκτέλεσης έως ότου φθάσει ο μέγιστος αριθμός επαναλήψεων
9. Υπολογισμός της ταχύτητας του κάθε σωματιδίου
10. Υπολογισμός της νέας θέσης του κάθε σωματιδίου
11. Υπολογισμός της νέας συνάρτησης ποιότητας του κάθε σωματιδίου
12. Ενημέρωση της βέλτιστης λύσης του κάθε σωματιδίου
13. Εύρεση του βέλτιστου σωματιδίου ολόκληρου του σμήνους
14. Τέλος εκτέλεσης
15. Επιστροφή βέλτιστου σωματιδίου (βέλτιστης λύσης).

4.7 Εφαρμογές Αλγορίθμων Νοημοσύνης Σμήνους

Τα τελευταία χρόνια, αυξάνονται διαρκώς οι προσπάθειες ανεύρεσης τρόπων ώστε να εφαρμόζεται η νοημοσύνη σμήνους σε ποικίλα προβλήματα. Ο τρόπος που αναζητούν τροφή τα μυρμήγκια έχει αποτελέσει την αιτία για την εφαρμογή μίας νέας μεθόδου για την αναδρομολόγηση της κυκλοφορίας στα σύγχρονα τηλεπικοινωνιακά δίκτυα. Η αλληλεπίδραση των μυρμηγκιών κατά την διαδικασία της μεταφοράς της τροφής τους, μπορεί αν οδηγήσει σε πιο αποτελεσματικούς αλγόριθμους για χρήση σε ρομπότ, καθώς επίσης ο καταμερισμός εργασίας ανάμεσα στις μέλισσες ίσως βοηθήσει σε αποτελεσματικότερες διαδικασίες στις βιομηχανίες με αλυσίδα συναρμολόγησης.

Οι κυριότερες εφαρμογές του αλγορίθμου ACO σε ασύρματα δίκτυα χωρίς συγκεντρωτική δομή (MANET) τα οποία αποτελούν και την κυριότερη πλατφόρμα για εφαρμογές κινητού υπολογισμού, αφορούν στην εύρεση μεθόδων αποτελεσματικής δρομολόγησης. Αρκετοί αλγόριθμοι έχουν αναπτυχθεί τα τελευταία χρόνια βασισμένοι στον ACO για την εξάλειψη των προβλημάτων δρομολόγησης που εμφανίζονται στα δίκτυα MANET εξαιτίας της κινητικότητας των πηγών. Ο αλγόριθμος AntHocNet είναι ένας υβριδικός αλγόριθμος που δημιουργεί αρχικά τους πίνακες δρομολόγησης και κατόπιν χρησιμοποιεί πράκτορες (μυρμήγκια) για την συνεχή αξιολόγηση των μονοπατιών. Ο αλγόριθμος ARA (Ant colony-based Routing Algorithm) είναι ένας αλγόριθμος βρίσκει το συντομότερο μονοπάτι ανάμεσα στους κόμβους χρησιμοποιώντας τα πακέτα δεδομένων ως πράκτορες για να ιχνηλατήσουν τα συντομότερα μονοπάτια.

Σε άλλες περιπτώσεις χρησιμοποιούνται πράκτορες για να διατρέξουν το δίκτυο τυχαία, ιχνηλατώντας τα μονοπάτια που χρησιμοποίησαν από την πηγή προς τους κόμβους. Αλγόριθμοι υπολογισμού θέσης με τη βοήθεια του ACO έχουν επίσης αναφερθεί, όπου οι πράκτορες δίνουν την πληροφορία δρομολόγησης που έχουν ανακτήσει. Ένας υβριδικός αλγόριθμος που συνδυάζει τον ACO με το πρωτόκολλο AODV χρησιμοποιεί πράκτορες που διατρέχουν τυχαία το δίκτυο και κρατούν πληροφορίες για τους n κόμβους που επισκέφθηκαν τελευταία ενημερώνοντας παράλληλα και τους πίνακες δρομολόγησης καταλλήλως.

Αρκετοί ακόμα αλγόριθμοι έχουν προταθεί για τη δρομολόγηση σε δίκτυα MANET. Όλοι αυτοί οι αλγόριθμοι στην προσπάθεια να αντιμετωπίσουν τις ιδιαιτερότητες αυτών των δικτύων, παρεκκλίνουν από τις αρχές του ACO, και οι περισσότεροι από αυτούς δεν διαφέρουν τελικά από κοινούς αλγορίθμους δρομολόγησης. Πρέπει επίσης να αναφερθεί ότι χρησιμοποιούνται παραλλαγές του ACO προκειμένου να υπάρχει αποτελεσματικότερη

ενεργειακή διαχείριση στα ασύρματα δίκτυα και οι οποίες έχουν ως σκοπό αποτελεσματικότερους αλγορίθμους δρομολόγησης για τη σωστή διαχείριση των ενεργειακών πόρων του δικτύου.

Ο αλγόριθμος PSO έχει βρει ποικίλες εφαρμογές σε διάφορα προβλήματα βελτιστοποίησης και αυτοοργάνωσης σχετικά με τα δίκτυα MANET. Συγκεκριμένα έχουν προταθεί μέθοδοι δρομολόγησης με την χρησιμοποίηση διαφόρων τεχνικών ,όπως επίσης και μέθοδοι αυτοοργάνωσης τέτοιων δικτύων προκειμένου να υπάρχει συνεργατική συμπεριφορά του συγκεκριμένου αλγορίθμου. Μάλιστα ο βαθμός της αυτοοργάνωσης που επιτεύχθηκε με την εφαρμογή του PSO δεν απέχει και πολύ από το βαθμό αυτοοργάνωσης που θα είχε το ίδιο δίκτυο σε περίπτωση κεντρικής διαχείρισης. Ο αλγόριθμος PSO έχει βρει εφαρμογή και στην ενεργειακή διαχείριση των κόμβων ενός δικτύου MANET με τη δημιουργία clusters για εξοικονόμηση ενέργειας όσον αφορά τη συλλογή δεδομένων από τους κόμβους-αισθητήρες.

Ένας άλλος τομέας στον οποίο βρίσκουν εφαρμογή οι αλγόριθμοι Νοημοσύνης Σμήνους είναι τα Σμήνη Ρομπότ (Swarm Robots). Η χρήση swarm robots για την διεκπεραίωση διαφόρων εργασιών παρουσιάζει πολλά πλεονεκτήματα όπως απλότητα κατασκευής ρομποτικών μονάδων, υψηλή πιστότητα (μικρή πιθανότητα λάθους) και μικρό κόστος. Ένα σμήνος ρομπότ μπορεί να πραγματοποιήσει πολύπλοκες εργασίες που είναι δύσκολες ακόμη και για τα παραδοσιακά ρομποτικά συστήματα. Ένα τέτοιο μοντέλο έχει προταθεί για την ανακάλυψη και αποφυγή εμποδίων με τη βοήθεια του PSO. Άλλα μοντέλα swarm robots έχουν επίσης προταθεί για ανθεκτικότητα αποτυχίας (fault tolerance), και συνεργατική αναζήτηση.

Θα πρέπει να αναφερθεί τέλος και η χρησιμοποίηση του PSO σε σμήνη UAV (Unmanned Aerial Vehicles) και UUV (Unmanned Underwater Vehicles) δηλαδή μη επανδρωμένων εναέριων και υποβρύχιων οχημάτων που χρησιμοποιούνται κυρίως για στρατιωτικούς σκοπούς. Χαρακτηριστικά παραδείγματα είναι η εφαρμογή του PSO για της εύρεση βέλτιστης διαδρομής προς το στόχο σε τρεις διαστάσεις, ενός σμήνους από UAV έτσι ώστε να αποφεύγουν εμπόδια, εχθρικές απειλές με την ταυτόχρονη κατανάλωση ελάχιστων καυσίμων, της χρησιμοποίησης του PSO από ένα σμήνος UAV το οποίο μέσω της συνεργασίας και της αυτοοργάνωσης να προστατεύει συγκεκριμένο χώρο από εχθρικές απειλές, όπως επίσης και της αυτοοργάνωσης και πλοήγησης σμήνους UUV με τη βοήθεια μιας βελτιωμένης έκδοσης του αλγορίθμου PSO .

Άλλες τυπικές εφαρμογές είναι το covering (εξερεύνηση σε εχθρικό έδαφος), patrolling (φύλαξη μουσείων από κλοπές), environment manipulation(έλεγχος κυκλοφορίας), self-assembling (επαναρυθμιζόμενα ρομπότ) και localization (βελτίωση προσδιορισμού θέσης). Επιπλέον σημαντική ερευνητική δουλειά έχει επενδυθεί στο σχεδιασμό συστημάτων σμήνους για την εξερεύνηση περιοχών γνωστών ή άγνωστων. Στην πλειονότητα τους οι προηγούμενες ερευνητικές εργασίες σχετικές με συστήματα νοημοσύνης σμήνους θεωρούν τις πηγές πληροφορίας στατικές εκτός ελάχιστων περιπτώσεων όπου οι προσπάθειες αφορούν δυναμικά περιβάλλοντα. Η νοημοσύνη σμήνους αναφέρεται σε δυναμικά περιβάλλοντα αλλά δεν θίγει θέματα εγκυρότητας πλαισίων.

Η νοημοσύνη σμήνους είναι πολύ σημαντική και εφαρμόζεται σε πάρα πολλούς τομείς. Οι επιστήμονες δεν γνωρίζουν όλες τις λεπτομέρειες των αλληλεπιδράσεων μέσα στα σμήνη κοινωνικών εντόμων και ένεκα τούτου δεν μπορούν να κατασκευάσουν λογισμικό που να τις προσομοιώνει. Επιπλέον, η νοημοσύνη σμήνους στερείται μιας γενικής θεώρησης των πραγμάτων και έτσι δεν μπορεί να εφαρμοστεί σε προβλήματα που απαιτούν βαθιά συλλογιστική.

ΚΕΦΑΛΑΙΟ 5

Μη Επανδρωμένα Οχήματα(UAVs)

5.1 Εισαγωγή

Ένα μη επανδρωμένο αεροσκάφος (Unmanned Aerial Vehicle (UAV) ή ένα εξ αποστάσεως χειριζόμενο όχημα (remotely operated aircraft (ROA), ή ένα εξ αποστάσεως κατευθυνόμενο όχημα (remotely piloted vehicle(RPV) ορίζεται ένα αεροσκάφος το οποίο δεν περιέχει πιλότο στο χώρο του αεροσκάφους ελέγχεται είτε από αυτόνομο σύστημά πλοήγησης είτε από τηλεχειριζόμενο σύστημα πλοήγησης.

5.2 Ιστορική Αναδρομή

Το πρώτο UAV κατασκευάστηκε το 1916 από τους Elmer Sperry και Peter Cooper Hewitt. Η κατασκευή αυτή ονομάστηκε Aviation Torpedo (αέρια τορπίλη), η οποία διακρίνεται παρακάτω, ήταν ικανή να μεταφέρει εκρηκτικά και πέταξε σε απόσταση περίπου 30 μιλίων. Παρ' όλα αυτά, λόγω μηδαμινής τεχνολογικής ανάπτυξης δε χρησιμοποιήθηκαν UAV σε κάποιον από τους δύο Παγκοσμίους Πολέμους. Για να πετύχουν την αυτόματη πλοήγηση του αεροσκάφους χρησιμοποίησαν ένα γυροσκόπιο με σκοπό τη σταθεροποίηση του κυρίως σώματος του αεροσκάφους. Αυτό είναι γνωστό ως ο Αυτόματος Έλεγχος Στάσης (Attitude) ενός αεροσκάφους.



Εικόνα 5.2.1: Sperry & Hewitt Areal Torpedo 1916

Κατα τη διάρκεια του Ψυχρού Πολέμου και του Πολέμου του Βιετνάμ ξεκίνησε η ουσιαστική ανάπτυξη των UAV, με συνεχή ανάπτυξη και έρευνα με αποκορύφωμα την δεκαετία του 1970 κυρίως από την ΗΠΑ και το Ισραήλ, όπου ξεκίνησαν η χρησιμοποίηση καμερών προβάλλοντας εικόνα στο χειριστή. Μια τέτοια κατασκευή είναι το IAI Scout Ισραηλινής κατασκευής , το οποίο διακρίνεται στη παρακάτω εικόνα.



Εικόνα 5.2.2: IAI Scout 1978

Η πραγματική χρήση των UAV πραγματοποιήθηκε από τις ΗΠΑ το 1991 κατά τη διάρκεια του Πολέμου του Κόλπου. Μετά από αυτά τα γεγονότα έγινε ταχεία ανάπτυξη των UAV. Το πιο διαδεδομένο UAV είναι το Predator, το οποίο είναι στρατιωτικής χρήσης που φαίνεται στη παρακάτω εικόνα.



Εικόνα 5.2.3: Το UAV Predator

Τη δεκαετία αυτή πρωτοπόρος στην ανάπτυξη UAV για μη στρατιωτικούς σκοπούς ήταν η NASA, μερικά τέτοια παραδείγματα είναι τα Helios και Altus, τα οποία διακρίνονται στις παρακάτω εικόνες και χρησιμοποιήθηκαν για περιβαλλοντολογικές μετρήσεις έχοντας την ικανότητα να πετούν σε μεγάλα υψόμετρα.



Εικόνα 5.2.4: NASA Helios



Εικόνα 5.2.5: NASA Altus

5.3 Τύποι UAVs

Υπάρχουν διάφοροι τύποι UAV, μερικοί από αυτούς είναι :

- Μικρό εναέριο οχήματα (MAV) : Με διαστάσεις μικρότερες των 15 εκατοστών , όπως το RQ-16 T Hawk [εικόνα 5.3.1]
- Τακτικής (tactical) χρήσης UAV όπως το Pioneer [εικόνα 5.3.2] το οποίο μπορεί να φτάσει μέχρι τα 15000 πόδια και ο χρόνος πτήσης του φτάνει μέχρι τις 6 ώρες πτήσης.
- Φορητά όπως το Raven της AeroVironment [εικόνα 5.3.3]
- Μεσαίου υψόμετρου-μεγάλης αντοχής UAV (MALE-Medium Attitude Long Endurance) όπως το Ισραηλινής κατασκευής Aeronautics Defense Dominator, το οποίο μπορεί να πραγματοποιήσει μέχρι 28 ώρες πτήσης και να φτάσει έως τα 30000 πόδια. [εικόνα 5.3.4]
- Μεγάλου Υψόμετρου-μεγάλης αντοχής UAV (HALE- High Attitude Long Endurance) όπως το Αμερικάνικης κατασκευής RQ-4 Global Hawk, το οποίο μπορεί να πραγματοποιήσει μέχρι 32 ώρες πτήσης και να φτάσει ως τα 65000 πόδια. [εικόνα 5.3.5]



Εικόνα 5.3.1: RQ-16 T Hawk



Εικόνα 5.3.2: Pioneer



Εικόνα 5.3.3: AeroVironment



Εικόνα 5.3.4: Aeronautics Defense Dominator



Εικόνα 5.3.5: Black Hawk

5.4 Εφαρμογές UAVs

Κατά βάση τα UAV έχουν στρατιωτική εφαρμογή και χρησιμοποιούνται για επιτήρηση, αναγνώριση στόχου, συλλογή πληροφοριών και επιθέσεις. Έχουν αρχίσει όμως να εμφανίζονται και πολιτικές εφαρμογές οι οποίες θα αναπτυχθούν στο μέλλον, μερικές από αυτές είναι :

- Παρακολούθηση του περιβάλλοντος.
- Εφαρμογές Πυρόσβεσης
- Εναέρια χαρτογράφηση
- Επιτήρηση χερσαίων και θαλάσσιων συνόρων
- Επιβολή νόμου και εφαρμογές ασφαλείας

5.5 Κατηγορίες UAVs

Οι βασικές κατηγορίες στις οποίες χωρίζονται τα UAV είναι :

- 1) Αυτά που οδηγούνται από κάποιον απομακρυσμένο χρήστη (remotely piloted vehicle - RPV)
- 2) Αυτά τα οποία είναι αυτόνομα ή προγραμματισμένα (Autonomous/Preprogrammed)

Τα RPV ελέγχονται από σταθμό εδάφους (Ground Control Station – GCS) μέσω ανθρώπινου χειριστή. Ένα χαρακτηριστικό παράδειγμα UAV που χρησιμοποιείται από ένα σταθμό εδάφους είναι το Αμερικάνικο Predator

Τα αυτόνομα UAV έχουν στο εσωτερικό τους έναν αυτόματα ελεγκτή ο οποίος ελέγχει τη πτήση η οποία έχει φορτωθεί στη μνήμη του και προβαίνει σε διορθωτικές κινήσεις σε περίπτωση που το UAV ξεφύγει από το σχέδιο πορείας της προγραμματισμένης πτήσης του.

5.6 Μαθηματικό Μοντέλο UAVs

Σε αυτή την ενότητα παρουσιάζεται το μαθηματικό μοντέλο του UAV. Αυτό όμως γίνεται με τις εξής παραδοχές :

- Η κίνηση του UAV εξετάζεται σε κανονικές καιρικές συνθήκες

- Οι μονάδες ελέγχου κινούνται αλλά δε παραμορφώνονται
- UAV αντιμετωπίζεται ως μη-παραμορφώσιμο αντικείμενο, με έξι βαθμούς ελευθερίας ως αποτέλεσμα της μετακίνησης των πηδαλίων
- Το βάρος w του UAV ποικίλει κατά τη πτήση
- Η βαρύτητα και η αεροδυναμική επηρεάζουν το UAV
- Ο έλεγχος των κινήσεων του UAV πραγματοποιείται από τις ακόλουθες παραμέτρους: τη στροφή φ , τη κλίση θ , την εκτροπή ψ και τη ταχύτητα V_0

Για όλα αυτά χρησιμοποιούμε το σύστημα αξόνων που περιγράφηκε παραπάνω (ενότητα Σωματόδετοι Άξονες αεροσκάφους)

Η γραμμική ταχύτητα ενός UAV σε σύστημα αξόνων Oxyz ορίζεται ως εξής:

$$V_0 = U_i + V_j + W_k$$

όπου U, V, W είναι η διαμήκη, εγκάρσια και κατακόρυφη ταχύτητα αντίστοιχα, ενώ i, j, k τα διανύσματα μονάδος του συστήματος που σχετίζονται με αντικείμενο (σε αυτή τη περίπτωση το UAV) που έχει σύστημα αξόνων Oxyz.

Η γωνιακή ταχύτητα ενός UAV, στο ίδιο σύστημα αξόνων ορίζεται ως εξής :

$$\Omega = P_i + Q_j + R_k$$

όπου P, Q, R προσδιορίζουν τη στροφή, τη κλίση και την εκτροπή αντίστοιχα.

Οι Κινηματικοί συσχετισμοί μεταξύ του συνόλου των παραμέτρων της γωνιακής ταχύτητας και τις παραγώγους των γωνιών έχουν την ακόλουθη μητρώϊκή μορφή:

$$\begin{bmatrix} \dot{\varphi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} 1 & \sin \varphi \tan \theta & \cos \varphi \tan \theta \\ 0 & \cos \varphi & -\sin \varphi \\ 0 & \sin \varphi \sec \theta & \cos \varphi \sec \theta \end{bmatrix} \begin{bmatrix} P \\ Q \\ R \end{bmatrix}$$

Οι κινηματικοί συσχετισμοί μεταξύ του συνόλου των παραμέτρων της γραμμικής ταχύτητας ($\dot{x}_1, \dot{y}_1, \dot{z}_1$), που μετρούνται στο σύστημα αξόνων $\mathbf{0}_1 x_1 y_1 z_1$ συνδέονται με τη Γ_1 και το σύνολο των παραμέτρων της ταχύτητας (U, V, W) στο σύστημα αξόνων Oxyz που συνδέονται με κινούμενα αντικείμενα έχουν τη παρακάτω μιτρώϊκή μορφή :

$$\begin{bmatrix} \dot{x}_1 \\ \dot{y}_1 \\ \dot{z}_1 \end{bmatrix} = \begin{bmatrix} \cos \psi \cos \theta & \cos \psi \sin \theta \sin \varphi - \cos \varphi \cos \psi & \cos \psi \sin \theta \cos \varphi + \sin \varphi \cos \psi \\ \sin \psi \cos \theta & \sin \psi \sin \theta \sin \varphi - \sin \varphi \cos \psi & \sin \psi \sin \theta \cos \varphi - \sin \varphi \cos \psi \\ -\sin \theta & \sin \varphi \cos \theta & \cos \varphi \cos \theta \end{bmatrix} \begin{bmatrix} U \\ V \\ W \end{bmatrix}$$

Οι γωνίες πρόσπτωσης και πλαγιολίσθησης, α και β αντίστοιχα υπολογίζονται ως εξής

$$\alpha = \arctan \frac{W}{U}$$

$$\beta = \arcsin \frac{V}{V_0}$$

Εφόσον σε ένα UAV έχει ανατεθεί κάποια προγραμματισμένη διαδρομή, αυτό θέτει περιορισμούς στη θέση, τη γραμμική και γωνιακή ταχύτητα του UAV. Έτσι πραγματοποιείται η προγραμματισμένη κίνηση :

$$\widehat{K_0 K_E} = s(x_{1p}, y_{1p}, z_{1p}, \varphi_p, \theta_p, \psi_p)$$

$$V_z(t) = \dot{s}(\dot{x}_{1p}, \dot{y}_{1p}, \dot{z}_{1p}, \dot{\varphi}_p, \dot{\theta}_p, \dot{\psi}_p)$$

όπου $x_{1p}, y_{1p}, z_{1p}, \varphi_p, \theta_p, \psi_p$ είναι οι παράμετροι κίνησης του ελεγχόμενου UAV.

Όταν UAV ανιχνεύσει το στόχο, μπορεί να τον ακολουθήσει ή να του επιτεθεί. Παρακάτω αναφέρονται οι παράμετροι κατά τις οποίες ελέγχεται το UAV. Υποθέτουμε ότι οι γωνιακές συντεταγμένες είναι ίδιες με αυτές του στόχου .

$$\varphi_p = \varphi_t$$

$$\theta_p = \theta_t$$

$$\psi_p = \psi_t$$

Το διάνυσμα της προκαθορισμένης θέσης του UAV στο σύστημα αξόνων Oxyz

$$\mathbf{r}_p = x_{1p} \mathbf{i}_1 + y_{1p} \mathbf{j}_1 + z_{1p} \mathbf{k}_1$$

όπου :

$$x_{1p} = r_{ot} \cos \psi_t \cos \theta_t$$

$$y_{1p} = -r_{ot} \sin \psi_t \cos \theta_t$$

$$z_{1p} = -r_{ot} \sin \theta_t$$

και r_{ot} είναι η απόσταση της πραγματικής θέσης του στόχου του UAV, ενώ $\varphi_t, \theta_t, \psi_t$ είναι οι γωνίες στροφής, κλίσης και την εκτροπής του κινούμενου στόχου (γωνίες Euler).

το διάνυσμα των παραμέτρων της προκαθορισμένης γραμμικής ταχύτητας του UAV περιγράφεται παρακάτω σε μητρωϊκή μορφή.

$$\begin{bmatrix} U_p \\ V_p \\ W_p \end{bmatrix} = \begin{bmatrix} \cos\psi_t \cos\theta_t & \sin\psi_t \cos\theta_t & -\sin\theta_t \\ \sin\varphi_t \cos\psi_t \sin\theta_t - \sin\psi_t \cos\varphi_t & \sin\varphi_t \sin\psi_t \sin\theta_t + \cos\psi_t \cos\varphi_t & \sin\varphi_t \cos\theta_t \\ \cos\psi_t \cos\psi_t \sin\theta_t + \sin\psi_t \sin\varphi_t & \cos\varphi_t \sin\psi_t \sin\theta_t - \cos\psi_t \sin\varphi_t & \cos\varphi_t \cos\theta_t \end{bmatrix} \begin{bmatrix} x\dot{1}_p \\ y\dot{1}_p \\ z\dot{1}_p \end{bmatrix}$$

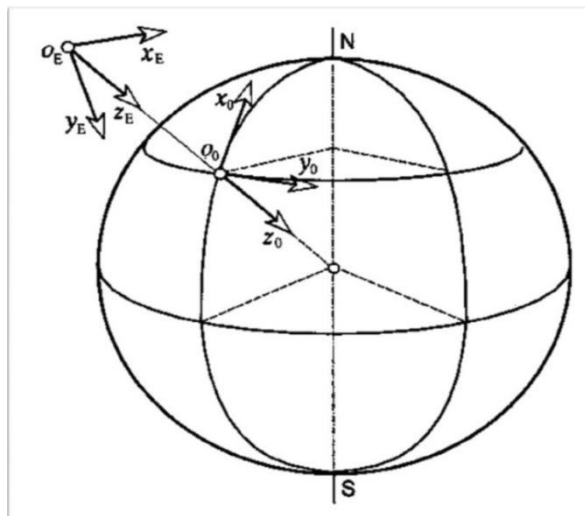
το διάνυσμα των παραμέτρων της προκαθορισμένης γωνιακής ταχύτητας του UAV περιγράφεται παρακάτω σε μητρωϊκή μορφή.

$$\begin{bmatrix} P_p \\ Q_p \\ R_p \end{bmatrix} = \begin{bmatrix} 1 & 0 & -\sin\theta_t \\ 0 & \cos\varphi_t & \sin\varphi_t \cos\theta_t \\ 0 & -\sin\varphi_t & \cos\varphi_t \cos\theta_t \end{bmatrix} \begin{bmatrix} \dot{\varphi}_t \\ \dot{\theta}_t \\ \dot{\psi}_t \end{bmatrix}$$

5.7 Κινηματικό Μοντέλο UAVs

5.7.1 Χωρόδετοι (Αδρανειακοί) Αξονες

Οι άξονες της γης (earth axis) ορίζονται από ένα σημείο αναφοράς O_0 όπου είναι η αρχή των αξόνων βασιζόμενοι σε ένα δεξιόστροφο σύστημα αναφοράς, οι οποίοι είναι οι O_0X_0 , O_0Y_0 , O_0Z_0 , όπως φαίνεται στη παρακάτω εικόνα.



Εικόνα 5.7.1: Οι άξονες της γης

Από τη παραπάνω εικόνα παρατηρούμε ότι ο άξονας O_0X_0 έχει κατεύθυνση προς το βορρά, ο άξονας O_0Y_0 έχει κατεύθυνση προς την ανατολή και ο άξονας O_0Z_0 έχει κατεύθυνση κατακόρυφα προς τα κάτω (το κέντρο της γης), παράλληλη με το βαρυτικό διάνυσμα. Το επίπεδο O_0, X_0, Y_0, Z_0 εφάπτεται στην επιφάνεια της Γης και ορίζει το οριζόντιο επίπεδο.

Άρα με τη προϋπόθεση ότι αναφερόμαστε σε επίπεδη-οριζόντια γη (flat) όπου υπάρχει βαρύτητα, ένα οποιοδήποτε αεροσκάφος το οποίο πετά οπουδήποτε στην ατμόσφαιρα βάσει

του σημείου αναφοράς O_0 μπορεί να περιγράψει η θέση του (συντεταγμένες), με το σύστημα αξόνων που περιεγράφηκε παραπάνω.

Το οριζόντιο αυτό επίπεδο ορίζεται από τους άξονες $OEXEYEZE$ και είναι παράλληλο προς το επίπεδο της επιφάνειας της γης $O_0X_0Y_0Z_0$ όπως φαίνεται από τη παραπάνω εικόνα. Οι άξονες $OEXEYEZE$ ονομάζονται γήινοι άξονες αναφοράς. Η μοναδική διαφορά είναι ότι ο άξονας $OEXE$ έχει κατεύθυνση όχι προς το βορρά αλλά προς μια τυχαία διεύθυνση πτήσης του αεροσκάφους.

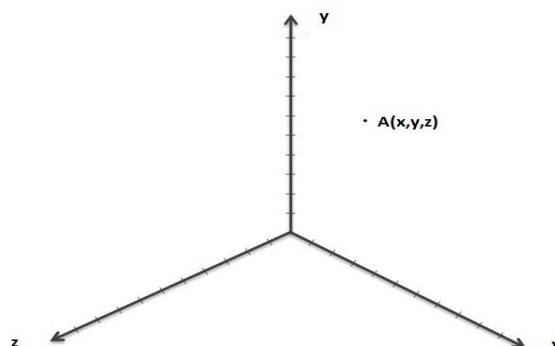
5.7.2 Σωματόδετοι Άξονες αεροσκάφους

Σε αντίθεση με άλλα οχήματα η κίνηση ενός αεροσκάφους περιλαμβάνει 3 άξονες : Τον άξονα x , τον άξονα y και τον άξονα z , ο οποίος προσδιορίζει το ύψος πτήσης του αεροσκάφους.

Συνεπώς, η θέση A ενός αεροσκάφους εξαρτάται από τις μεταβλητές x , y και z , όπως φαίνεται στη παραπάνω εικόνα. Επιπροσθέτως, το αεροσκάφος έχει 6 βαθμούς ελευθερίας (DoF), άρα συμπεριλαμβάνει ακόμα 3 άξονες οι οποίοι περνούν από το κέντρο αυτού.

Ο πρώτος είναι ο άξονας του Yaw (εκτροπή) ο οποίος τέμνεται κάθετα του κέντρου βάρους του αεροσκάφους και προσδιορίζει τη γωνία με την οποία θα κινηθεί το αεροσκάφος, δηλαδή τις μοίρες με τις οποίες το αεροσκάφος στρίβει είτε αριστερά είτε δεξιά.

Ο δεύτερος είναι ο άξονας του Roll (κλίση) ο οποίος τέμνεται οριζόντια του κέντρου βάρους του αεροσκάφους και κάθετα με τον άξονα του Yaw και προσδιορίζει κατά πόσες μοίρες έχει κλίση το αεροσκάφος προς τα δεξιά ή προς τα αριστερά.



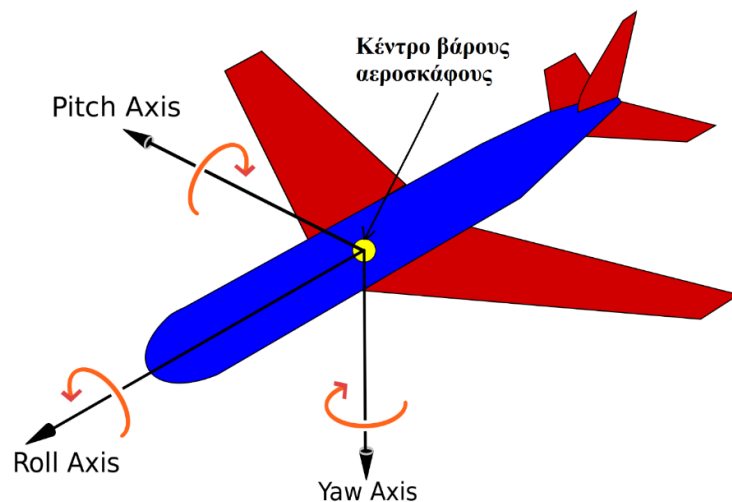
Εικόνα 5.7.2.1: Ένα τυπικό σύστημα αξόνων xyz

Ο τρίτος και τελευταίος άξονας είναι αυτός του Pitch (πρόνευση) , είναι κάθετος των αξόνων Yaw και Roll και τέμνεται οριζόντια του κέντρου βάρους του αεροσκάφους. Σε αντίθεση με το προηγούμενο άξονα (Roll) κινείται κατά πλάτος του αεροσκάφους και όχι κατά μήκος.

Οι δύο πρώτοι είναι υπεύθυνοι για την για την αλλαγή κατεύθυνσης του αεροσκάφους (Yaw, Roll) ενώ ο τελευταίος (Pitch) είναι υπεύθυνος για το καθορισμό ανόδου ή καθόδου του αεροσκάφους.

Με αυτούς τους άξονες το αεροσκάφος μπορεί να επιτελέσει τις εξής λειτουργίες : εμπρόσθια κίνηση, κάτω πλάγια κίνηση και τέλος περιστροφή γύρω και από τους τρεις προαναφερθέντες άξονες .

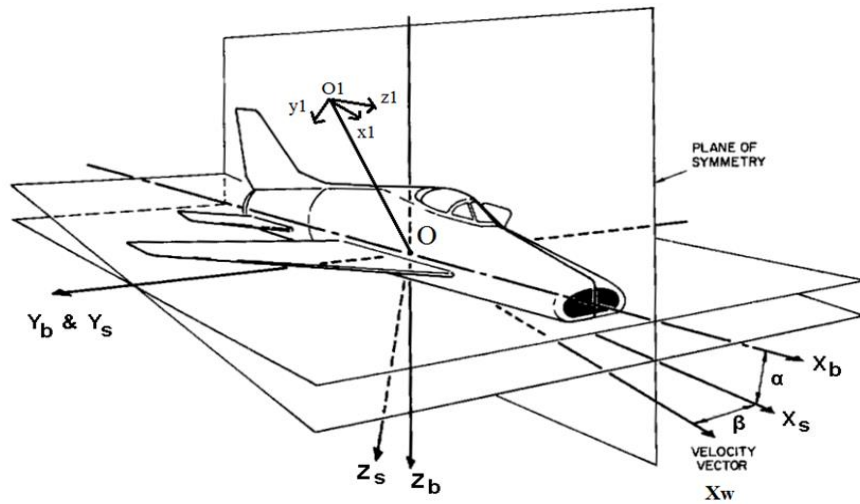
Οι άξονες του αεροσκάφους διακρίνονται στη παρακάτω εικόνα.



Εικόνα 5.7.2.2: Οι άξονες του αεροσκάφους

Το σύστημα των αξόνων αυτών είναι προσδεμένο και κινείται μαζί με το αεροσκάφος. Έτσι εάν συμβεί και μεταβληθούν οι αρχικές συνθήκες πτήσης οι άξονες κινούνται μαζί με το αεροσκάφος.

Αυτό το σύστημα αξόνων ονομάζεται σωματόδετο σύστημα αξόνων (body axis system) και φαίνεται στη παρακάτω εικόνα



Εικόνα 5.7.2.3: Σωματόδετο σύστημα αξόνων

Το επίπεδο OX_bZ_b ορίζει το επίπεδο συμμετρίας του αεροσκάφους ενώ γενικά μας εξυπηρετεί ο άξονας OX_b να είναι παράλληλος με τη γεωμετρική αναφορά της ατράκτου (horizontal fuselage datum), ο άξονας OY_b έχει φορά προς τα δεξιά και ο άξονας OZ_b προς τα κάτω.

Ο άξονας Ox_w είναι σύμφωνα με τη παραπάνω εικόνα παράλληλος με το διάνυσμα ολικής ταχύτητας V του ανέμου, και προσδιορίζεται σε σχέση με τον άξονα OX_b μέσω των γωνιών α και β , όπου α η γωνία πρόσπτωσης και β η γωνία πλαγιολίσθησης. Αυτοί οι άξονες ($OX_wZ_wY_w$) ονομάζονται αεροδυναμικοί.

Σε σταθερή πτήση (steady flight) οι άξονες του ανέμου $OX_wZ_wY_w$ είναι ένας τύπος του σωματόδετου συστήματος αξόνων, το οποίο έχει περιστραφεί γύρω από τον άξονα OY_b .

Σε συνέχεια των παραπάνω υπάρχουν οι άξονες ευστάθειας $OXSZSYS$ όπως διακρίνονται παραπάνω. Η διαφορά με τους αεροδυναμικούς άξονες είναι ότι ο άξονας είναι παράλληλος με τη προβολή της σχετικής ταχύτητας του ανέμου στο επίπεδο Oxz . Κατ' αυτό το τρόπο ορίζεται η γωνία πλαγιολίσθησης β . Με $\beta=0$ οι άξονες ανέμου και ευστάθειας ταυτίζονται μεταξύ τους.

5.8 Επικοινωνίες UAVs

5.8.1 Ασύρματη Επικοινωνία

Ασύρματη επικοινωνία ονομάζεται αυτή μέσω της οποίας η πληροφορία μεταφέρεται κατά μία απόσταση χωρίς τη χρήση καλωδίων. Αυτή η απόσταση μπορεί να είναι είτε μικρή, είτε μεγάλη.

Σήμερα, η ασύρματη επικοινωνία γίνεται όλο και πιο σύνηθες φαινόμενο στη σύγχρονη ζωή. Κινητά τηλέφωνα, Wi-Fi, ακόμα και συστήματα ασφαλείας χρησιμοποιούν τεχνικές ασύρματης επικοινωνίας. Αυτό συμβαίνει για μερικούς από τους εξής λόγους.

- Η ασύρματη επικοινωνία εκτείνεται σε μια απόσταση πέρα από τις δυνατότητες της καλωδιακής επικοινωνίας. Όπως για παράδειγμα η χρήση του διαδικτύου γίνεται πιο δύσκολη όταν χρησιμοποιούμε καλώδιο για να συνδεθούμε, καθώς πιθανώς να υπάρχουν περιορισμοί λόγω του μήκους των καλωδίων. Κατ' αυτό το τρόπο οι άνθρωποι μπορούν να πραγματοποιήσουν πιο άνετα την εργασία τους.
- Μια εφεδρική σύνδεση παρέχεται κατά τις ασύρματες επικοινωνίες σε περιπτώσεις βλάβης.
- Με τις ασύρματες τεχνικές επικοινωνίας υπάρχει η δυνατότητα σύνδεσης φορητών συσκευών ή προσωρινών σταθμών εργασίας. Κατ' αυτό το τρόπο παρέχονται περισσότερες συνδέσεις και επιλογές.

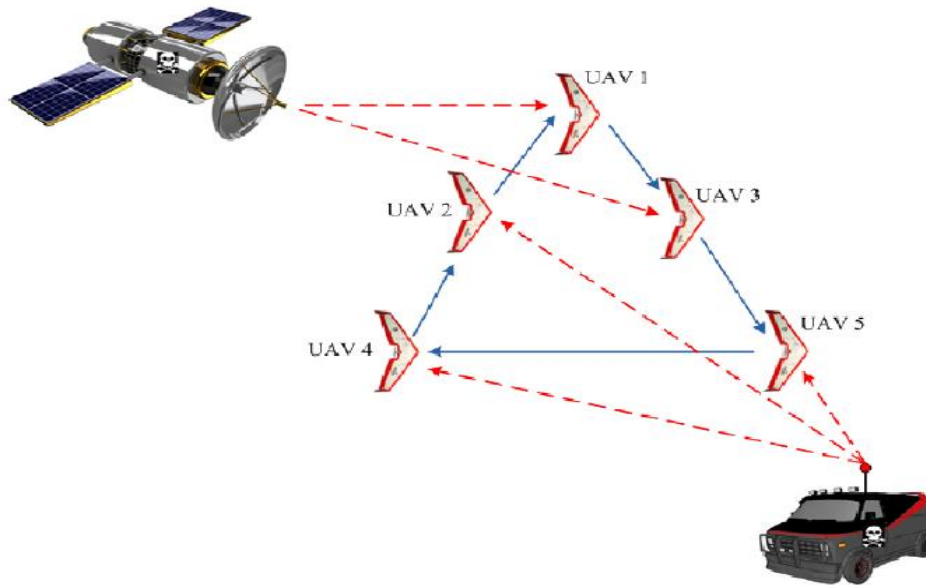
Οι παραπάνω είναι μερικοί από τους παράγοντες των ασύρματων επικοινωνιών. Μερικά παραδείγματα αυτών είναι το γνωστό Wi-Fi ή το τηλεχειριστήριο μιας τηλεόρασης που επιτυγχάνει επικοινωνία μέσω υπέρυθρων και είναι μικρής εμβέλειας.

5.8.2 Ασύρματες Τεχνολογίες Στην Επικοινωνία Των UAVs

Από τη στιγμή που τα UAV μπορούν να κινηθούν ευέλικτα και γρήγορα, προφανώς και είναι αδύνατο να γίνει χρήση καλωδίων, ιδιαίτερα όταν πετούν.

Ακόμα όμως και με τη χρήση ασύρματων τεχνικών, θα πρέπει να ληφθεί υπόψη το ύψος πτήσης και η ταχύτητα του UAV.

Στη παρακάτω εικόνα βλέπουμε πως επιτυγχάνεται η επικοινωνία μεταξύ των UAV



Εικόνα 5.8.2: η επικοινωνία μεταξύ των UAV

Ωστόσο, τα παρακάτω είναι μια σειρά πλεονεκτημάτων χρησιμοποιώντας τεχνολογίες ασύρματης δικτύωσης.

1. Τα UAV μπορούν να παρέχουν υψηλής ποιότητας επικοινωνία λόγω της διάδοσης του σήματος, όπως φαίνεται στη παραπάνω εικόνα όπου διακρίνεται ο τρόπος επικοινωνίας μεταξύ των UAV.
2. Τα UAV μπορούν να βελτιώσουν τη ποιότητα της ασύρματης επικοινωνίας και δικτύωσης προσαρμόζοντας κατάλληλα τη πορεία τους.
3. Τα UAV μέσω των ασύρματων αυτών τεχνολογιών μπορούν να αντιλαμβάνονται και με στόχο την ανάπτυξη της περιοχής ενδιαφέροντος τους να προβούν σε δυναμική συγχώνευση των κόμβων δεδομένων.
4. Τα UAV μπορούν να διαβιβάσουν και να μεταφέρουν τεράστιο όγκο δεδομένων.

5.8.3 Τα Βασικά Ενός Συστήματος Επικοινωνίας

Υπάρχουν διάφορα εξαρτήματα που θα πρέπει να συμπεριληφθούν σε ένα σύστημα επικοινωνίας :

- Το στοιχείο εκπομπής
- Η συσκευή λήψης
- Το περιβάλλον μέσα από το οποίο λαμβάνει χώρα η επικοινωνία
- Κεραίες

Ο ρόλος του πομπού σε ασύρματες επικοινωνίες είναι να τροφοδοτεί ένα σήμα σε μια κεραία για εκπομπή.

Ένας ραδιοφωνικός πομπός κωδικοποιεί τα δεδομένα σε κύματα RF (ραδιοκύματα) για την προβολή του σήματος σε ένα δέκτη.

Ο δέκτης λαμβάνει και αποκωδικοποιεί δεδομένα που έρχονται μέσω της κεραίας λήψης. Ο δέκτης εκτελεί το έργο της αποδοχής και της αποκωδικοποίησης των σημάτων RF ενώ απορρίπτει ανεπιθύμητα ή πλεονάζοντα δεδομένα. Να σημειωθεί εδώ συχνότητα ραδιοκυμάτων για την επικοινωνία των UAV είναι στα 2.4GHz.

Ο χώρος μεταξύ του πομπού και του δέκτη που ονομάζεται " περιβάλλον ". Εμπόδια και κύματα που υπήρχαν στο περιβάλλον μειώνουν την ικανότητα του συστήματος να παρθούν πληροφορίες από το ένα μέρος σε άλλο. Στοιχεία όπως ο αέρας, τα δέντρα και τα κτίρια μπορούν να μειώσουν το φάσμα της επικοινωνίας .

Κεραίες είναι συσκευές που εστιάζουν την ενέργεια σε μια συγκεκριμένη κατεύθυνση. Οι κεραίες μπορούν παρέχουν διαφορετικά σχέδια ακτινοβολίας, ανάλογα με το σχεδιασμό και την εφαρμογή.

5.8.4 Πρωτόκολλα Ασύρματης Επικοινωνίας

Στον τομέα της επικοινωνίας , υπάρχουν μερικά πρότυπα IEEE . Το IEEE 802 είναι μια ομάδα προτύπων που σχετίζονται με αυτό το έργο. Αναφέρεται σε μία οικογένεια προτύπων IEEE που ασχολούνται με τοπικά και μητροπολιτικά δίκτυα .

Οι υπηρεσίες και τα πρωτόκολλα που ορίζονται στο πρότυπο IEEE 802 βρίσκονται στα 2 κατώτερα στρώματα του μοντέλου αναφοράς δικτύωσης OSI, δηλαδή στο φυσικό επίπεδο (Physical) και στο επίπεδο Σύνδεσης Δεδομένων (Data Link).

Υπάρχουν αρκετά πρωτόκολλα που μπορούν να χρησιμοποιηθούν σε ασύρματες επικοινωνίες, μερικά από αυτά είναι :

- IEEE 802.11 a / b / g / n ασύρματου LAN (πιστοποίηση Wi - Fi), όπου η επικοινωνία επιτυγχάνεται σε ζώνες συχνοτήτων των 2.4 , 3.6 και 5GHz
- IEEE 802.15 ασύρματο PAN (bluetooth), όπου η επικοινωνία επιτυγχάνεται σε αποστάσεις το πολύ 100 μέτρων.
- IEEE 802.16 ευρυζωνικής ασύρματης πρόσβασης (πιστοποίηση WiMAX) όπου η επικοινωνία επιτυγχάνεται σε ζώνες συχνοτήτων από 2.3 έως και 2.7 GHz

5.8.5 Παράγοντες Που Επηρεάζουν Την Επικοινωνία Των UAVs

Υπάρχουν ορισμένοι παράγοντες που πρέπει να λαμβάνονται υπόψη κατά την επικοινωνία των UAV Ένας από αυτούς είναι η αποδοτικότητα του φάσματος. Αναφέρεται στο ρυθμό πληροφοριών που μπορούν να μεταδοθούν σε ένα δεδομένο εύρος ζώνης μέσα σε ένα ειδικό σύστημα επικοινωνίας. Όπως είναι γνωστό, η μελέτη για το εύρος ζώνης είναι πολύ σημαντική όταν πρόκειται για τη διαβίβαση των δεδομένων .

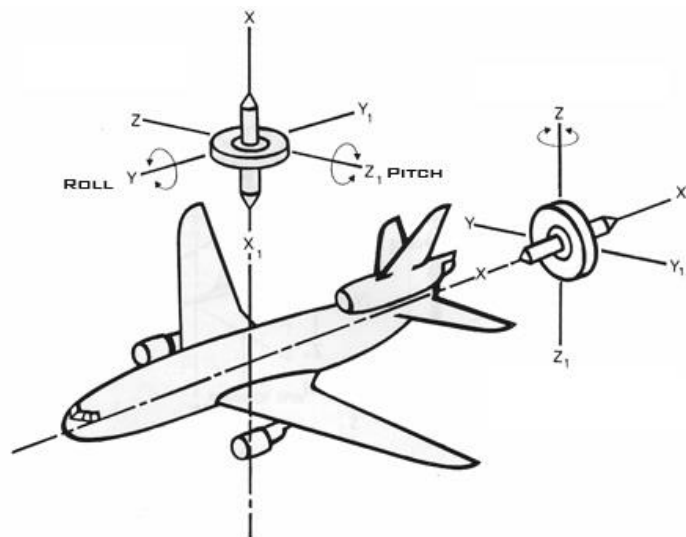
Ένας υψηλότερος ρυθμός μετάδοσης δεδομένων μπορεί να παρέχει μετάδοση δεδομένων σε μικρότερο χρονικό διάστημα. Ένας άλλος σημαντικός παράγοντας είναι η αντοχή. Όταν τα UAV πετούν στον ουρανό, χρειάζονται ενέργεια για να διατηρήσουν την πτήση τους. Η ενέργεια είναι ένας περιορισμένος πόρος , ιδίως για τα μικρά UAV . Το εύρος και ο ρυθμός δεδομένων, κατά τη μετάδοση δεδομένων των UAV εξαρτώνται από τη διαθέσιμη ενέργεια του αεροσκάφους. Στο στρατιωτικό πεδίο εφαρμογής , υπάρχουν επίσης πολλά άλλα θέματα, όπως: η ικανότητα του ρελέ, η ασφάλεια των δεδομένων, μεταξύ άλλων.

5.9 Αισθητήρια Του UAVs

Τα UAV κατά βάση χρησιμοποιούν τα παρακάτω αισθητήρια (sensors)

5.9.1 Γυροσκόπιο

Ένα γυροσκόπιο μετρά το ποσοστό της γωνιακής αλλαγής μέχρι και σε τρεις γωνιακούς άξονες. Οι μονάδες είναι συχνά μοιρών ανά δευτερόλεπτο. Ένα γυροσκόπιο δεν μετρά τις απόλυτες γωνίες άμεσα, αλλά πρέπει να επαναληφθεί η μέτρηση -όπως συμβαίνει και στο επιταχυνσιόμετρο- για να πραγματοποιήσουμε την επιθυμητή γωνία μετατόπισης. Η έξοδος του πραγματικού γυροσκόπιου τείνει να είναι αναλογική αλλά στις περισσότερες περιπτώσεις αυτό γίνεται από τον κώδικα του ελεγκτή πτήσης. Το γυροσκόπιο θα πρέπει να τοποθετηθεί έτσι ώστε άξονες περιστροφής του είναι ευθυγραμμισμένο με τους άξονες του UAV. Στη παρακάτω εικόνα διακρίνουμε το γυροσκόπιο σε συνάρτηση με τους άξονες του UAV.



Εικόνα 5.9.1: Ένα γυροσκόπιο

5.9.2 Επιταχυνσιόμετρο

Όπως υποδηλώνει το όνομα, επιταχυνσιόμετρα μέτρηση γραμμική επιτάχυνση μέχρι και σε τρεις άξονες. Η έξοδος ενός επιταχυνσιόμετρου μπορεί να ενσωματωθεί δύο φορές για να δώσει μια θέση αν και λόγω απωλειών στην έξοδο, υπόκειται σε « μετατόπιση ». Ένα πολύ σημαντικό χαρακτηριστικό των τριών αξόνων του επιταχυνσιόμετρου είναι ότι ανιχνεύει βαρύτητα και μπορεί να γνωρίζει ποια κατεύθυνση είναι «κάτω». Αυτό παίζει σημαντικό ρόλο επιτρέποντας αεροσκάφη να παραμένουν σταθερά. Το επιταχυνσιόμετρο θα πρέπει να

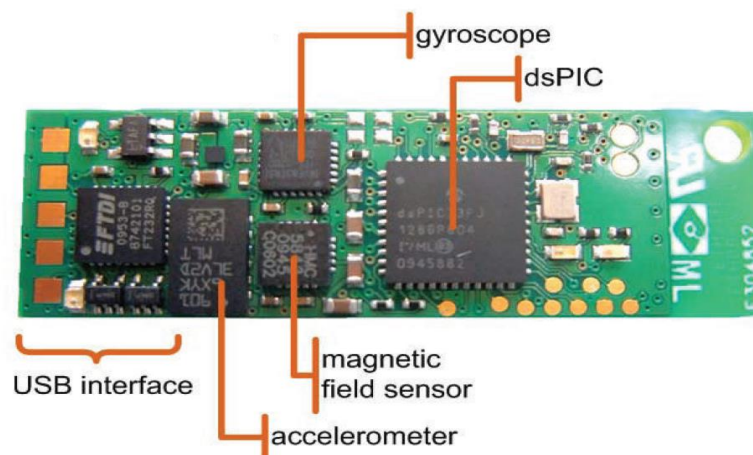
τοποθετηθεί στον ελεγκτή πτήσης, έτσι ώστε οι γραμμικοί άξονες να ευθυγραμμίζονται με τους κύριους άξονες του UAV.



Εικόνα 5.9.2: Ένα επιταχυνσιόμετρο

5.9.3 Μαγνητόμετρο

Το μαγνητόμετρο είναι μια ηλεκτρονική μαγνητική πυξίδα είναι σε θέση να μετρήσει το μαγνητικό πεδίο της γης και να το χρησιμοποιήσει για να καθορίσει την κατεύθυνση της πυξίδας του μη επανδρωμένου αεροσκάφους του (σε σχέση με το μαγνητικό βορρά) . Τα μαγνητόμετρα χρησιμοποιούνται για συλλογή πληροφοριών που έχουν να κάνουν με τη κατεύθυνση του UAV σε τρεις διαστάσεις, και εφόσον είναι συνδεδεμένα με επιταχυνσιόμετρα και γυροσκόπια τότε δημιουργούν τη Μονάδα Αδρανειακής Μέτρησης, παρέχοντας συνολικά 9 άξονες μέτρησης (Inertial Measurement Unit - IMU).

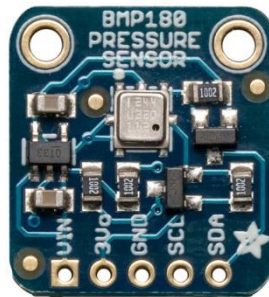


Εικόνα 5.9.3: Ένα Μαγνητόμετρο

5.9.4 Βαρόμετρο/Αισθητήρας Πίεσης

Από τη στιγμή που η ατμοσφαιρική πίεση αλλάζει όσο πιο μακριά είναι από το επίπεδο της θάλασσας ένας αισθητήρας πίεσης μπορεί να χρησιμοποιηθεί για να μας δώσει μια αρκετά ακριβή μέτρηση για το ύψος του UAV. Οι περισσότεροι ελεγκτές πτήσης λαμβάνουν

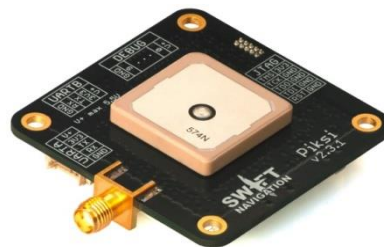
είσοδο τόσο από τον αισθητήρα πίεσης και το υψόμετρο του GPS για να υπολογίσει με μεγαλύτερη ακρίβεια το ύψος του UAV πάνω από την επιφάνεια της θάλασσας.



Εικόνα 5.9.4: Ένα Βαρόμετρο/Αισθητήρας Πίεσης

5.9.5 GPS

Μια συσκευή GPS (Global Position System) χρησιμοποιεί τα σήματα που αποστέλλονται από έναν αριθμό δορυφόρων σε τροχιά γύρω από τη γη, προκειμένου να καθοριστεί η ειδική γεωγραφική του θέση. Ένας ελεγκτής πτήσης μπορεί να έχουν επί του σκάφους GPS είτε να είναι συνδεδεμένο σε αυτό μέσω ενός καλωδίου. Η κεραία GPS δεν πρέπει να συγχέεται με το ίδιο το τσιπ GPS. Για να πάρουμε μια ακριβή θέση του GPS, το τσιπ του GPS θα πρέπει να λαμβάνει δεδομένα από πολλούς δορυφόρους, όσο περισσότεροι δορυφόροι τόσο το καλύτερο.



Εικόνα 5.9.5: Ένα GPS

5.10 Ελεγκτής Πτήσης

Ένας ελεγκτής πτήσης είναι συνήθως ένα ολοκληρωμένο κύκλωμα, το οποίο αποτελείται από μικροεπεξεργαστή, αισθητήρες, και ακροδέκτες εισόδου-εξόδου. Από μόνος του ο ελεγκτής πτήσης δε γνωρίζει το τύπο του UAV ή τις ρυθμίσεις που θέλουμε να του ορίσουμε. Αυτές οι ρυθμίσεις πραγματοποιούνται μέσω προγράμματος λογισμικού, και μόλις ολοκληρώσουμε τις παραμέτρους που θέλουμε, αυτές αποστέλλονται στον ελεγκτή πτήσης.



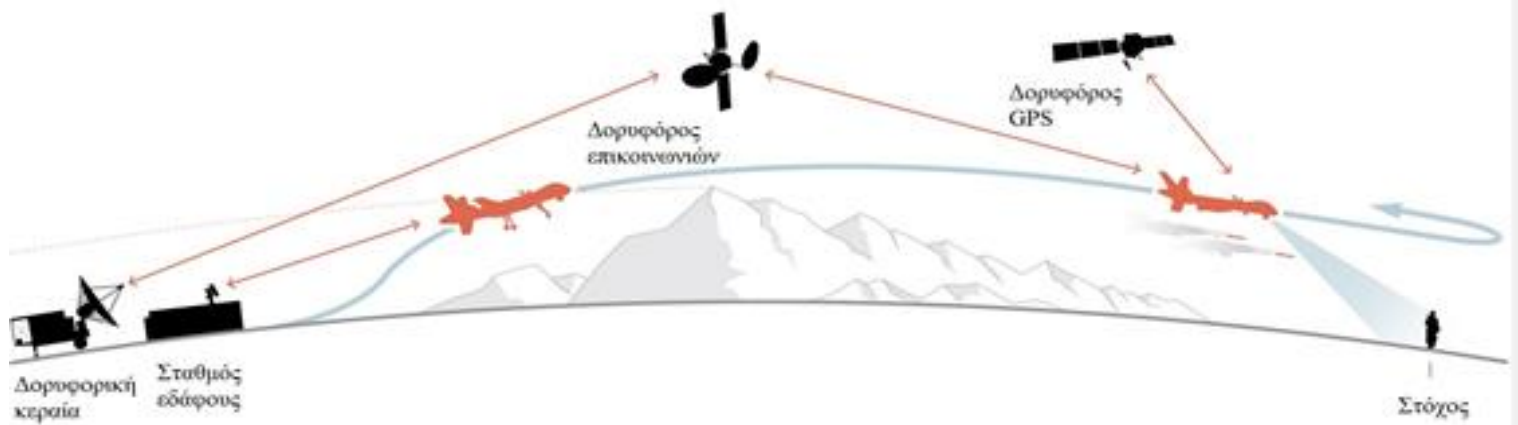
Εικόνα 5.10: Ένα ελεγκτής πτήσης

Ο ελεγκτής πτήσης χρησιμοποιεί ένα μικροελεγκτή (έναν από τους AVR, PIC, ARM - οι δύο πρώτοι είναι των 8 και των 16 bit, ενώ ο τελευταίος των 16 και 32 bit αντίστοιχα), μια CPU (από 8 έως 64 bit) η οποία λειτουργεί σε μία συγκεκριμένη συχνότητα σε Hz, τη μνήμη του προγράμματος (Program Memory ή αλλιώς Flash), όπου αποθηκεύεται το πρόγραμμα, μία στατική μνήμη RAM (SRAM- Static Random Access Memory) για να πραγματοποιεί υπολογισμούς, μια μνήμη EEPROM για αποθήκευση πληροφοριών που δεν επηρεάζουν τη πτήση, έναν ADC (Analog to Digital Converter), ο οποίος αναλογικές μετρήσεις σε ψηφιακή μορφή δεδομένων, και διάφορες ακροδέκτες εισόδου-εξόδου.

5.11 Πως Λειτουργεί Το UAV

Για να πραγματοποιήσει πτήση το UAV απαιτείται ένας ελεγκτής πτήσης, έτσι ώστε να απογειωθεί, να προσγειωθεί και να πλοηγηθεί. Ένας τέτοιος ελεγκτής μπορεί να πάρει πολλές μορφές, από χειριστήριο τύπου gamepad, μέχρι smartphone ή tablet. Ανεξάρτητα όμως από τη μορφή του, ο ελεγκτής επικοινωνεί συνήθως με ραδιοκύματα με συχνότητα 2.4GHz, είτε μέσω WiFi επίσης στα 2.4GHz, κάτι το οποίο τα smartphones και τα tablets μπορούν να αξιοποιήσουν χωρίς περεταίρω αξεσουάρ.

Τα μη επανδρωμένα αεροσκάφη έχουν υιοθετήσει την εν πλω τεχνολογία που αναπτύχθηκε λόγω της επανάστασης των smartphones. Ένα από αυτά είναι το GPS που παρουσιάστηκε παραπάνω, το οποίο αναμεταδίδει τη θέση του στον ελεγκτή πτήσης. Καταγράφει το σημείο απογείωσης του αεροσκάφους, σε περίπτωση που χρειάζεται να επιστρέψει πίσω χωρίς βοήθεια. Αυτό περιγράφεται και στη παρακάτω εικόνα. Για να μπορεί να πετάξει χρησιμοποιεί όλους τους αισθητήρες που αναφέρθηκαν στο παραπάνω κεφάλαιο.



Εικόνα 5.11: Συνεργασία UAVs με GPS, το οποίο αναμεταδίδει τη θέση του στον ελεγκτή πτήσης

Από τη στιγμή που το UAV απογειωθεί και για όσο υπάρχει οπτική επαφή, ελέγχεται από το σταθμό εδάφους (GCS), όταν η οπτική επαφή μεταξύ Σταθμού εδάφους και UAV χαθεί, τότε ο σταθμός μεταβαίνει σε έλεγχο μέσω δορυφορικής σύνδεσης. Αυτό πραγματοποιείται με το GPS με το οποίο είναι εξοπλισμένο το UAV, έτσι ώστε να μεταδίδει τη θέση του. Εάν η ζεύξη επικοινωνίας χαθεί, το UAV είναι προγραμματισμένο να πετά αυτόνομα σε κύκλους ή να επιστρέψει στη βάση, μέχρι να γίνει επανασύνδεση της σύνδεσης.

5.12 Σμήνος UAVs

Ένα σμήνος από UAV μπορεί να παρουσιαστεί με ένα σμήνος πτηνών. Προφανώς το σμήνος αυτό παρέχει περισσότερες δυνατότητες έναντι ενός μόνο αεροσκάφους. Αυτό συμβαίνει γιατί με την αύξηση των αεροσκαφών μοιράζονται οι αρμοδιότητες του καθενός καθώς και ο χρόνος εκτέλεσης της εκάστοτε εργασίας μειώνεται κατά πολύ. Όσο μεγαλύτερος ο αριθμός του σμήνους των αεροσκαφών, τόσο το καλύτερο για το χρόνο εκτέλεσης της αποστολής που έχουν να διεκπεραιώσουν τα αεροσκάφη. Ακόμα και αν κάποιο από τα αεροσκάφη παρουσιάσουν δυσλειτουργία, και χρειαστεί να υποχωρήσει, τα υπόλοιπα αεροσκάφη θα αναπληρώσουν το έργο του και θα φέρουν σε πέρας την αποστολή που τους ανατέθηκε και σίγουρα με μεγαλύτερη επιτυχία και αποδοτικότητα απ' ότι θα έκανε ένα μεμονωμένο αεροσκάφος. Γι' αυτό σε τέτοιες περιπτώσεις επιβάλλεται η χρήση σμήνους μη επανδρωμένων αεροσκαφών.

Η αυτόνομη λειτουργία του σμήνους των μη επανδρωμένων αεροσκαφών προκύπτει από τη χρησιμοποίηση λογισμικού. Για τον έλεγχο ενός μόνο αεροσκάφους, θα ήταν εύκολη η υλοποίηση, αλλά με τη χρήση σμήνους τα πράγματα γίνονται λίγο πιο περίπλοκα, καθώς αυτό απαιτεί και συλλογή και επεξεργασία των δεδομένων πτήσης όλου του σμήνους. Αυτό σημαίνει ότι το ένα UAV πρέπει να επικοινωνεί με το άλλο ανταλλάσσοντας πληροφορίες σχετικά με τη πτήση τους και το ποσοστό επιτυχίας της αποστολής τους.

ΚΕΦΑΛΑΙΟ 6

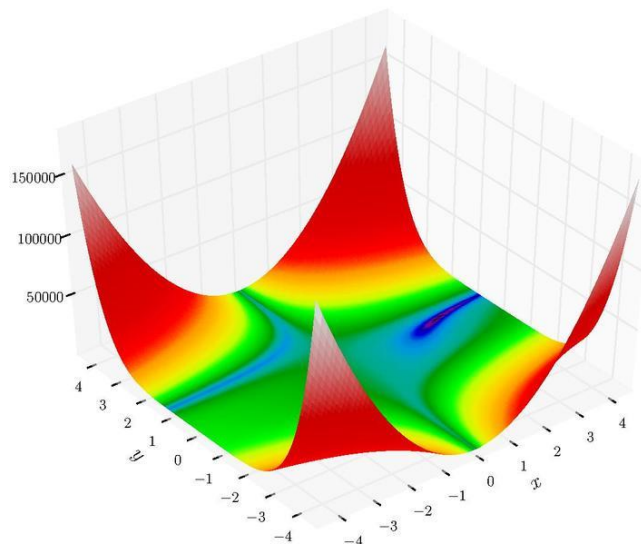
Εφαρμογή και Υλοποίηση Αλγορίθμου

6.1 Test Functions

Οι συναρτήσεις που αναφέρονται παρακάτω είναι μερικές από τις κοινές συναρτήσεις και τα σύνολα δεδομένων που χρησιμοποιούνται για τον έλεγχο και την αξιολόγηση των αλγορίθμων βελτιστοποίησης όπως:

- Ταχύτητα σύγκλισης.
- Ακρίβεια.
- Ευρωστία.
- Γενική απόδοση.

6.1.1 Beale Function

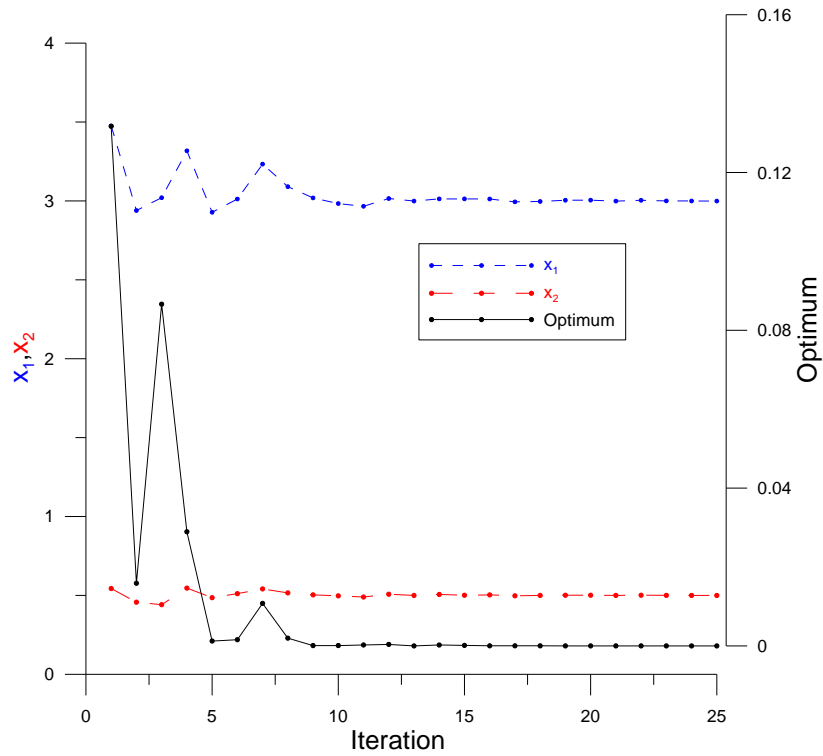


Εικόνα 6.1.1.1: Η συνάρτηση beale είναι πολυτροπική, με κοφτερές κορυφές στις γωνίες του πεδίου εισόδου.

$$f(x, y) = (1.5 - x + xy)^2 + (2.25 - x + xy^2)^2 + (2.625 - x + xy^3)^2$$

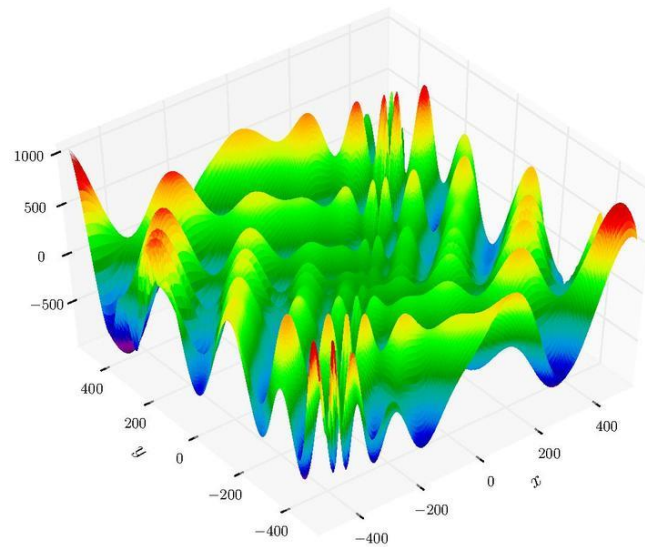
$$\text{minimum: } f(x, y) = f(3, 0.5) = 0$$

$$\text{search domain: } -4.5 \leq x \leq 4.5 \quad -4.5 \leq y \leq 4.5$$



Εικόνα 6.1.1.2: Η εξέλιξη της μεταβολής της συνάρτησης

6.1.2 Eggholder Function



Εικόνα 6.1.2.1: Η συνάρτηση Eggholder είναι μια δύσκολη συνάρτηση για τη βελτιστοποίηση, λόγω του μεγάλου αριθμού των τοπικών ελαχίστων.

$$f(x, y) = -(y + 47) \sin\left(\sqrt{\left|y + \frac{x}{2} + 47\right|}\right) - x \sin\left(\sqrt{|x - (y + 47)|}\right)$$

$$\text{minimum: } f(x, y) = f(512, 404.2319) = -959.6407$$

$$\text{search domain: } -512 \leq x \leq 512 \quad -512 \leq y \leq 512$$

Trial with :

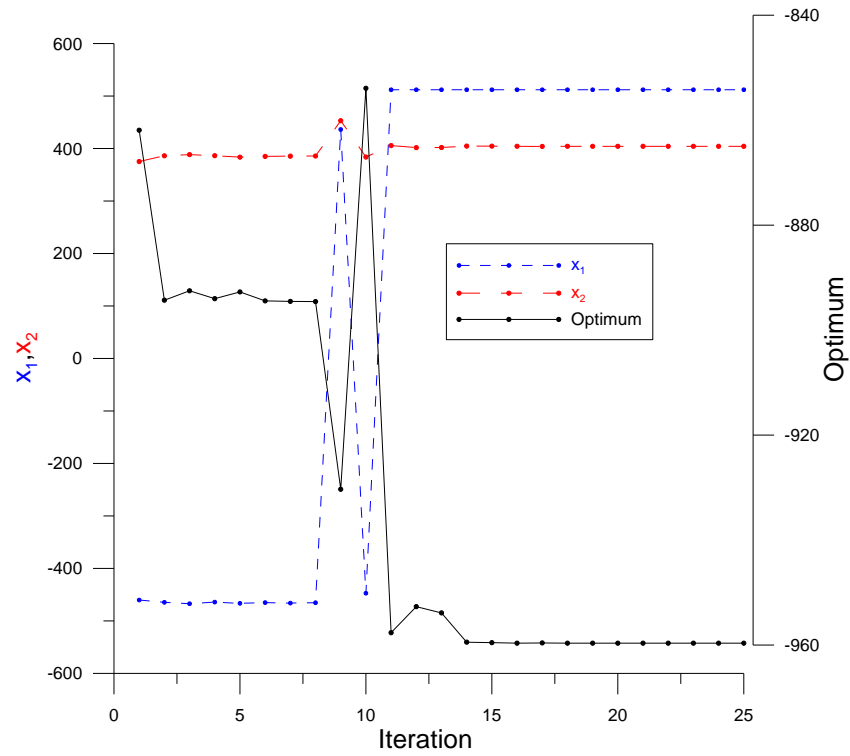
$$\alpha = (1,1)$$

number of particles : 400

$$\beta = (1,1)$$

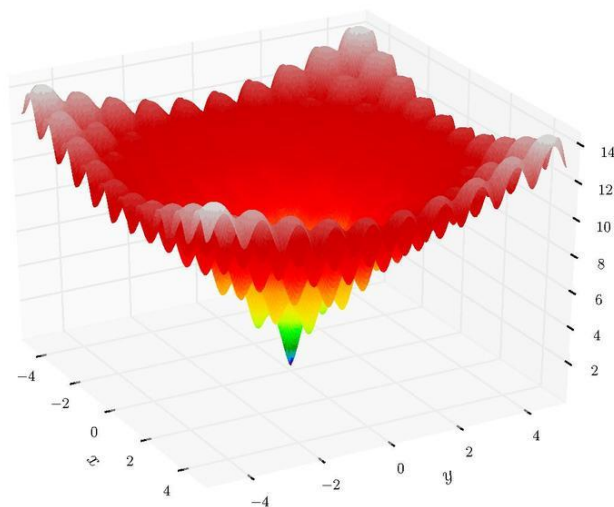
number of iterations : 25

$$\theta = 0.5$$



Εικόνα 6.1.2.2: Η εξέλιξη της μεταβολής της συνάρτησης

6.1.3 Ackley Function



Εικόνα 6.1.3.1: Η συνάρτηση Ackley χρησιμοποιείται ευρέως για τον έλεγχο των αλγορίθμων βελτιστοποίησης. Στη δισδιάστατη μορφή, όπως φαίνεται στην παραπάνω γραφική χαρακτηρίζεται από μια σχεδόν επίπεδη

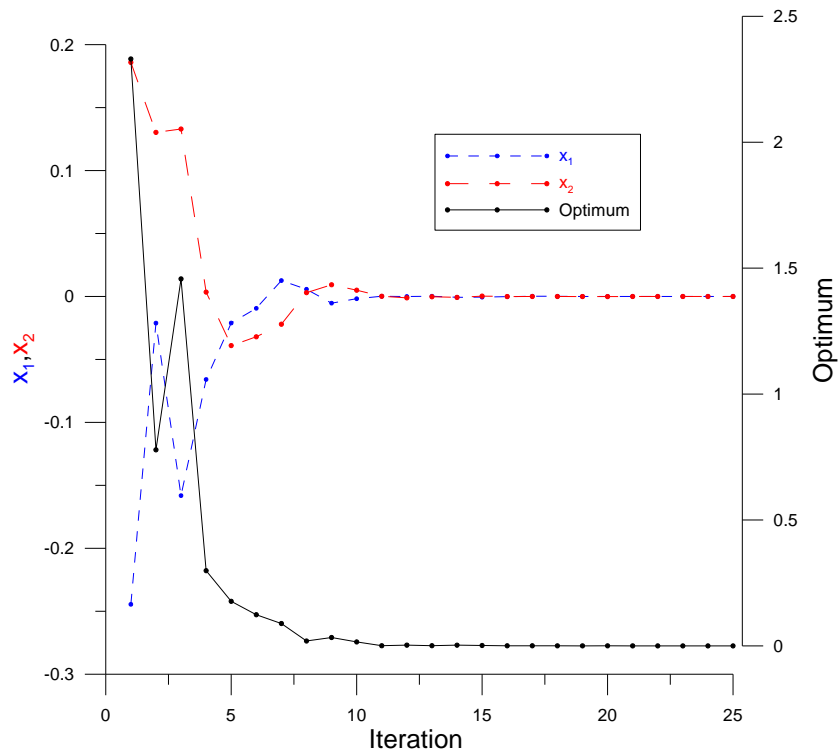
εξωτερική περιφέρεια, καθώς και μια μεγάλη τρύπα στο κέντρο. Η συνάρτηση ενέχει κίνδυνο για τους αλγόριθμους βελτιστοποίησης, ιδιαίτερα για τους hillclimbing αλγόριθμους, να παγιδευτούν σε ένα από τα πολλά τοπικά ελάχιστα της.

$$f(x, y) = -20 \exp(-0.2 \sqrt{0.5(x^2 + y^2)}) - \exp(0.5 \cos 2\pi x + 0.5 \cos 2\pi y) + 20 + e$$

minimum: $f(x, y) = f(0, 0) = 0$
 search domain: $-5 \leq x \leq 5 \quad -5 \leq y \leq 5$

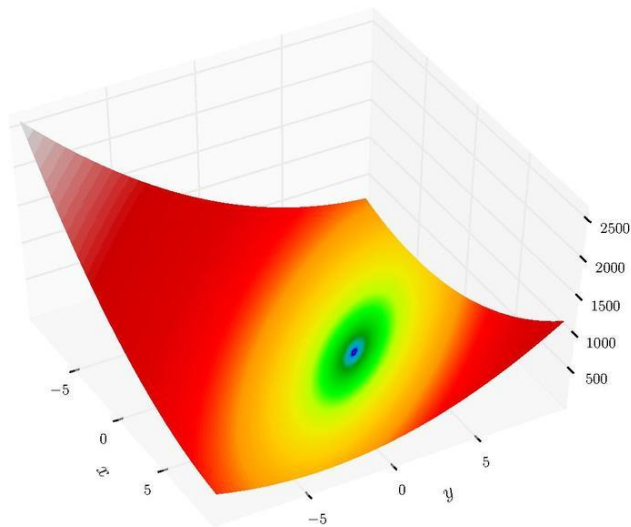
Trial with :

$\alpha = (1, 1)$ number of particles : 30
 $\beta = (1, 1)$ number of iterations : 25
 $\theta = 0.33$



Εικόνα 6.1.3.2: Η εξέλιξη της μεταβολής της συνάρτησης

6.1.4 Booth Function



Εικόνα 6.1.4.1: Η συνάρτηση Booth Function

$$f(x, y) = (x + 2y - 7)^2 + (2x + y - 5)$$

minimum: $f(x, y) = f(1, 3) = 0$

search domain: $-10 \leq x \leq 10 \quad -10 \leq y \leq 10$

Trial with :

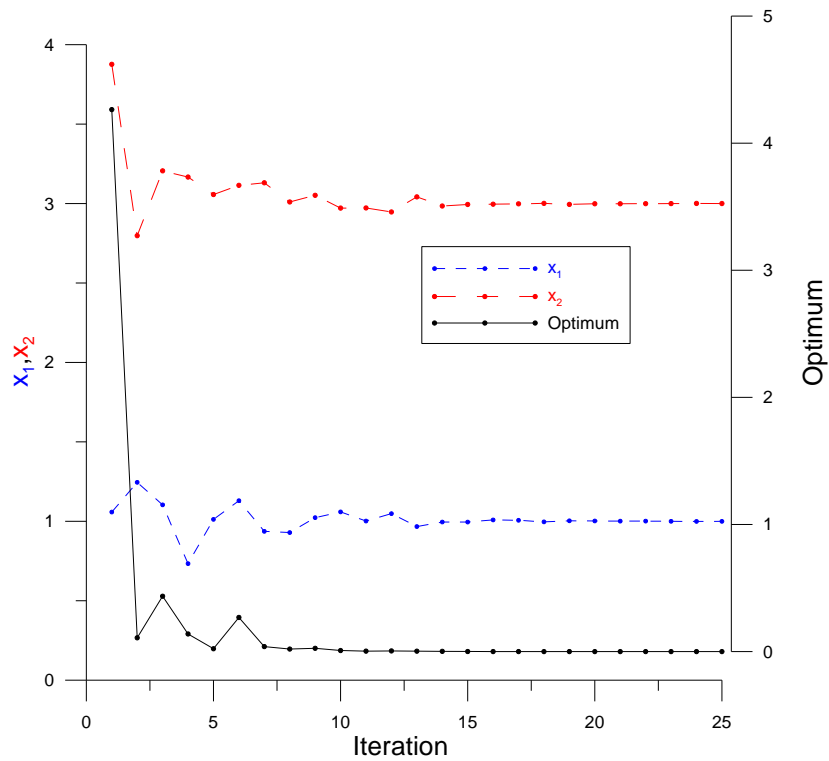
$$\alpha = (1, 1)$$

number of particles : 30

$$\beta = (1, 1)$$

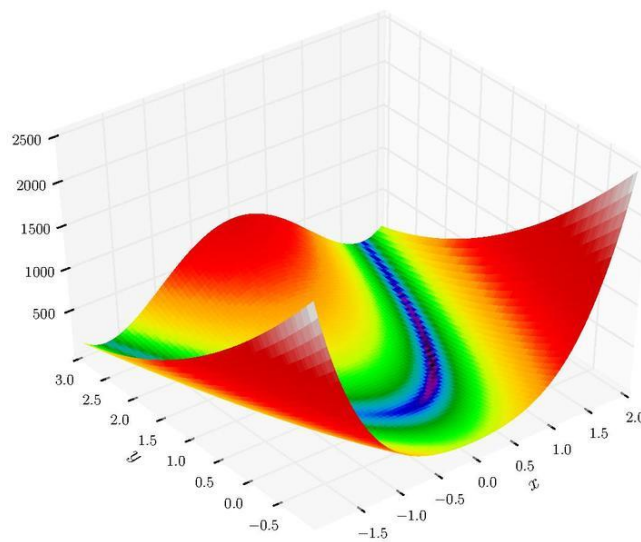
number of iterations : 25

$$\theta = 0.5$$



Εικόνα 6.1.4.2: Η εξέλιξη της μεταβολής της συνάρτησης

6.1.5 Rosenbrock Function



Εικόνα 6.1.5.1: Η συνάρτηση Rosenbrock, που αναφέρεται επίσης ως συνάρτηση κοιλάδα ή μπανάνα, είναι ένα δημοφιλές παράδειγμα λειτουργίας για αλγορίθμους βελτιστοποίησης που βασίζονται σε κλίση. Φαίνεται και από τη γραφική πάνω στη δισδιάστατη μορφή. Η συνάρτηση είναι μονοτροπική και το σφαιρικό ελάχιστο έγκειται σε μια στενή, παραβολική κοιλάδα. Ωστόσο, ακόμα κι αν η κοιλάδα είναι εύκολο να βρεθεί, η σύγκλιση στο ελάχιστο είναι δύσκολο (Picheny et al., 2012).

$$f(x, y) = 100(y - x)^2 + (x - 1)^2$$

$$\text{minimum: } f(x, y) = f(1, 1) = 0$$

$$\text{search domain: } -10 \leq x \leq 10 \quad -10 \leq y \leq 10$$

Trial with :

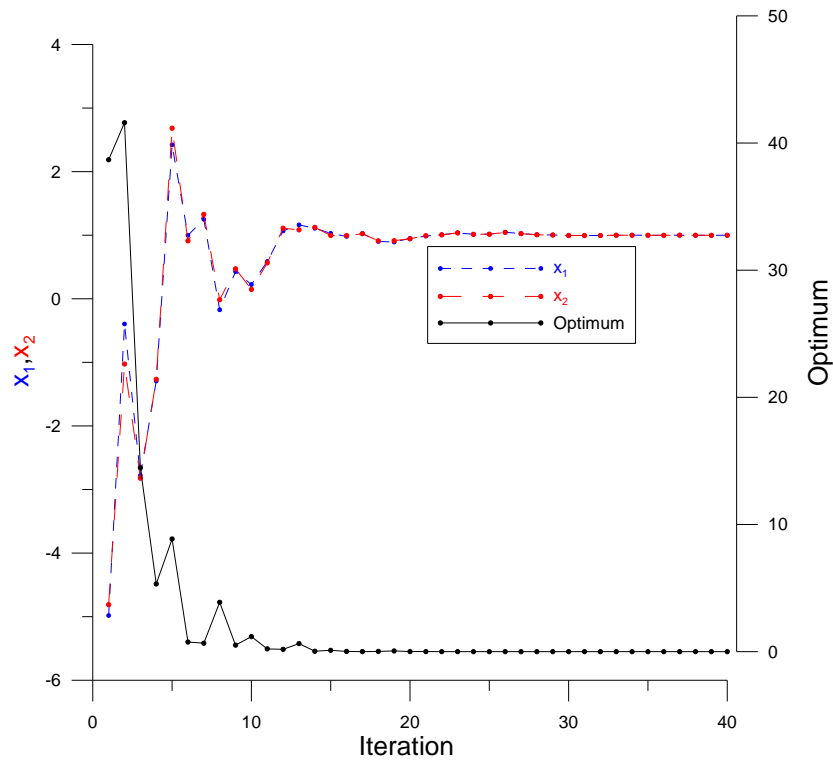
$$\alpha = (1, 1)$$

number of particles : 30

$$\beta = (1, 1)$$

number of iterations : 40

$$\theta = 0.5$$

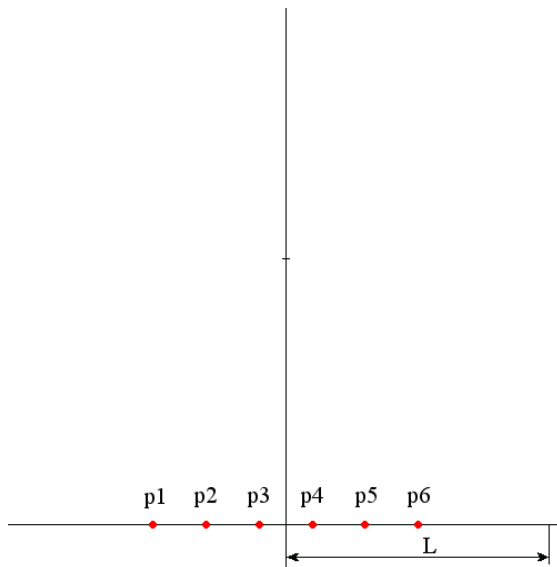


Εικόνα 6.1.5.2: Η εξέλιξη της μεταβολής της συνάρτησης

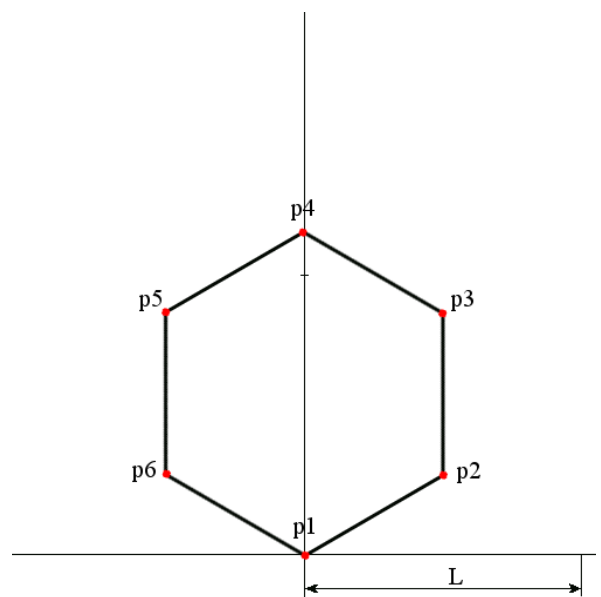
6.2 Εφαρμογή Και Υλοποίηση Αλγορίθμου

Στον παρακάτω κώδικα παρουσιάζεται η μετάβαση από το αρχικό σχηματισμό στο τελικό επιθυμητό σχηματισμό και ο έλεγχος τους επιτυγχάνεται χρησιμοποιώντας βελτιστοποίηση σμήνους σωματιδίων. Ο αλγόριθμος ΒΣΣ έχει ως βάση ένα σμήνος (swarm) από σωματίδια (particles), όπου κάθε σωματίδιο αναπαριστά μια υποψήφια λύση (candidate solution) για το πρόβλημα βελτιστοποίησης. Τα σωματίδια αυτά κινούνται σε ένα πολυδιάστατο χώρο αναζήτησης, όπου η θέση του καθενός κάθε χρονική στιγμή – για την ακρίβεια σε κάθε βήμα της επαναληπτικής διαδικασίας – καθορίζεται τόσο από την ίδια του την εμπειρία όσο και από αυτή του υπόλοιπου σμήνους. Σε αναλογία με τους εξελικτικούς αλγορίθμους (EA, evolutionary algorithms), το σμήνος της ΒΣΣ αντιστοιχεί στον πληθυσμό (population) των EA, ενώ τα σωματίδια αντιστοιχούν στα άτομα (individuals). Ωστόσο, σε αντίθεση με τους EA, η ΒΣΣ δε χρησιμοποιεί εξελικτικούς τελεστές, όπως η διασταύρωση (crossover) και η μετάλλαξη (mutation).

6.2.1 Σχηματισμός Από Ευθεία Σε Εξάγωνο



Εικόνα 6.2.1.1: Αρχικός Σχηματισμός



Εικόνα 6.2.1.2: Τελικός Σχηματισμός

func

```
function f = func(x,init,final)
%Evaluates the objective function to be minimized

%x is the independent variables, that is the displacements of the UAVs
%x(1:6) are the displacements in the x axis
%x(7:12) are the displacements in the y axis
%init(1:6) are the x coordinates of the initial positions
%init(7:12) are the y coordinates of the initial positions
%final(1:6) are the x coordinates of the final positions
%final(7:12) are the y coordinates of the final positions

inter=zeros(1,12); %inter are the coordinates of the UAVs after
                    %they are moved (to their test positions)
%inter(1:6) are the x coordinates of the test positions
%inter(7:12) are the y coordinates of the test positions

for i=1:12
    inter(i)=init(i)+x(i);
end

dist=zeros(6); %dist(j,i) is the distance of UAV j from final position i
                %after it has moved according to the displacements x and y
                %and have acquired the test positions

%   | position of UAV1 from final position 1 | position of UAV1 from final position 2 | position of
%   |                                       |                                       | UAV1 from final position 3 ...
%   |                                       |                                       |
%   | position of UAV2 from final position 1 | position of UAV2 from final position 2 | position of
%   |                                       |                                       | UAV2 from final position 3 ...
%   |                                       |                                       |
%   | position of UAV3 from final position 1 | position of UAV3 from final position 2 | position of
%   |                                       |                                       | UAV3 from final position 3 ...

for i=1:6
    for j=1:6
        dist(j,i)=sqrt((inter(j)-final(i))^2+(inter(j+6)-final(i+6))^2);
    end
end

d=min(dist');
%d is a 6 element vector that holds the minimum distance of each UAV at its
%test position from all of the final desired positions

obj=0;

for i=1:6
    obj=obj+d(i);
end
%The part of the objective function up to this point has to be zero in an
%optimum solution, in order to assure that all of the UAVs acquire a final
%position that is at one of the prescribed final positions
```

```

limit=1000*eye(6);
%Array limit holds the distances of each final position from the rest final
%positions, except the diagonal which is set to a large value (1000)

for i=1:6
    for j=1:6
        if(i~=j)
            limit(i,j)=sqrt((fina(i)-fina(j))^2+(fina(i+6)-fina(j+6))^2);
        end
    end
end

min_dist=min(min(limit)); %min_dist is the minimum distance amongst all of the
                        %final positions from the rest

check=1000*eye(6);
%Array check holds the distances of each test position from the rest test
%positions, except the diagonal which is set to a large value (1000)

for i=1:6
    for j=1:6
        if(i~=j)
            check(i,j)=sqrt((inter(i)-inter(j))^2+(inter(i+6)-inter(j+6))^2);
        end
    end
end

if(min(min(check))<(0.90*min_dist))
    obj=100;
end
%If in a candidate solution, two (or more) UAVs are closer than (0.90*min_dist)
%this solution is penalized

found=zeros(1,12);

for i=1:12
    found(i)=init(i)+ inter(i);
end
%found is a vector that holds the test positions
%found(1:6) are the x coordinates
%found(7:12) are the y coordinates
%it is not used in for the evaluation of the objective function

distances=zeros(6,1);
%distances is a a vector that hold the total displacement for each UAV

for i=1:6
    distances(i)=sqrt((x(i))^2+(x(i+6))^2);
end

obj = obj + 0.1*sum(distances);
%at this point the objective function is augmented by the sum of the total
%displacements for each of the UAVs, that has to be minimized, multiplied by a
%weighting factor of 0.1

```

```
f=obj;  
  
end
```

```
function [best]=psopso(nop,nov,noit,b_lb,b_ub,alphas,betas,theta,init,fina)  
  
[ xi ] = initialize_positions(nop,nov,b_lb,b_ub);  
%xi : particle positions  
  
u=zeros(nop,nov);  
  
[ fi ] = evaluate_obj_fun(xi,init,fina);  
%fi : objective function values over xi  
  
[ ~ ,g, ~ ] = find_current_overall_best(fi,xi);  
%g : current overall best  
  
xi_star=xi;  
% xi_star : current best of each particle  
  
fi_star=fi;  
% fi_star : current best objective function of each particle  
  
best=zeros(noit,nov+2);  
  
for it=1:noit  
  
    for ip=1:nop  
  
        a=alphas.*rand(1,nov);  
        b=betas.*rand(1,nov);  
        first_mul=g-xi(ip,1:nov);  
        second_mul=xi_star(ip,1:nov)-xi(ip,1:nov);  
        u(ip,1:nov)=theta*u(ip,1:nov)+a.*first_mul+b.*second_mul;  
    end  
  
    for ip=1:nop  
        xi(ip,1:nov)=xi(ip,1:nov)+u(ip,1:nov);  
        xi(ip,1:nov)=check_postions(nov,xi(ip,1:nov),b_lb,b_ub);  
    end  
  
    [ fi ] = evaluate_obj_fun(xi,init,fina);  
    [xi_star,fi_star]=update_xi_star(xi,xi_star,fi,fi_star);  
    [ g_value ,g, ~ ] = find_current_overall_best(fi,xi);  
    best(it,1)=it;  
    best(it,2)=g_value;  
    best(it,3:(nov+2))=g;
```

```
end
```

initialize_positions

```
function [ initial_positions ] = initialize_positions(nop,nov,lb,ub)
%Initialize positions of particles

%%%%%%%%%INPUTS
% nop : number of particles
% nov : number of independent variables
% lb : lower limit of independent variables
% ub : uper limit of independent variables

%%%%%%%%%OUTUTS
% initial_positions : array containing the initial positions
% initial_positions size will be equal to nop*nov

x=zeros(nop,nov);

for i=1:nop
    for j=1:nov
        x(i,j)=lb(j)+rand*(ub(j)-lb(j));
    end
end

initial_positions=x;
```

makeinitial

```
function init = makeinitial(L,a)

init=zeros(1,12);

% Shimatismos 0

init(1)=-5*L/10;
init(1+6)=0;

init(2)=-3*L/10;
init(2+6)=0;

init(3)=-1*L/10;
init(3+6)=0;

init(4)=1*L/10;
init(4+6)=0;

init(5)=3*L/10;
init(5+6)=0;
```

```
init(6)=5*L/10;  
init(6+6)=0;
```

makefinal

```
function fina = makefinal(a)  
fina=zeros(1,12);  
  
% Shimatismos 3  
  
fina(1)=0;  
fina(1+6)=0;  
  
fina(2)=a*cos(30*pi/180);  
fina(2+6)=a*0.5;  
  
fina(3)=a*cos(30*pi/180);  
fina(3+6)=a*1.5;  
  
fina(4)=0;  
fina(4+6)=2*a;  
  
fina(5)=-a*cos(30*pi/180);  
fina(5+6)=1.5*a;  
  
fina(6)=-a*cos(30*pi/180);  
fina(6+6)=0.5*a;
```

evaluate_obj_fun

```
function [ obj_fun_values ] = evaluate_obj_fun(particle_positions,init,fina)  
% Evaluate the objective function over a set of particle positions  
% The evaluation must be provided through the function func  
nop=size(particle_positions,1);  
  
obj_fun_values=zeros(nop,1);  
  
for i=1:nop  
    x=particle_positions(i,:);  
    obj_fun_values(i,1)=func( x,init,fina );  
end
```


final_eval

```
function e=final_eval(init,x)

e=init+x;
e=[e(1:6);e(7:12)];

end
```

find_current_overall_best

```
function [ current_overall_best_value ,current_overall_best_particle,
current_overall_best_particle_index ] = find_current_overall_best(obj_fun_values,particle_positions)
%Finds the best current value of objective function
%and the corresponding particle and its index

[current_overall_best_value, current_overall_best_particle_index ]=min(obj_fun_values);
```

update_individual_particle_current_best_positions

```
function
individual_particle_current_best_positions=update_individual_particle_current_best_positions(partic
e_positions,obj_fun_values,obj_fun_values_before,individual_particle_current_best_positions)

%nop=size(particle_positions,1);
nop=30;

for i=1:nop
    if(obj_fun_values(i)<obj_fun_values_before(i))
        individual_particle_current_best_positions(i,:)=particle_positions(i,:);
    else
        end
end
end
snapnow;
```

check_postions

```
function xi=check_postions(nov,xi,b_lb,b_ub)

for iv=1:nov

    if(xi(iv)<b_lb(iv))
        xi(iv)=b_lb(iv);
```

```

end

if(xi(iv)>b_ub(iv))
    xi(iv)=b_ub(iv);
end

end

```

update_xi_star

```

function [xi_star,fi_star]=update_xi_star(xi,xi_star,fi,fi_star)

nop=size(xi,1);
nov=size(xi,2);

for ip=1:nop
    if(fi(ip)<fi_star(ip))
        xi_star(ip,1:nov)=xi(ip,1:nov);
        fi_star(ip)=fi(ip);
    end
end

end

```

mul_runs_pso

```

function [MUL_best,OVERALL_best,ind,time]=
mul_runs_pso(nor,nop,nov,noit,b_lb,b_ub,alphas,betas,theta,init,final)

tic

MUL_best=zeros(noit,nov+2,nor);
for i=1:nor
    best=pso(nop,nov,noit,b_lb,b_ub,alphas,betas,theta,init,final);
    MUL_best(:,i)=best;

    disp('RUN MADE')
    disp(i)
end

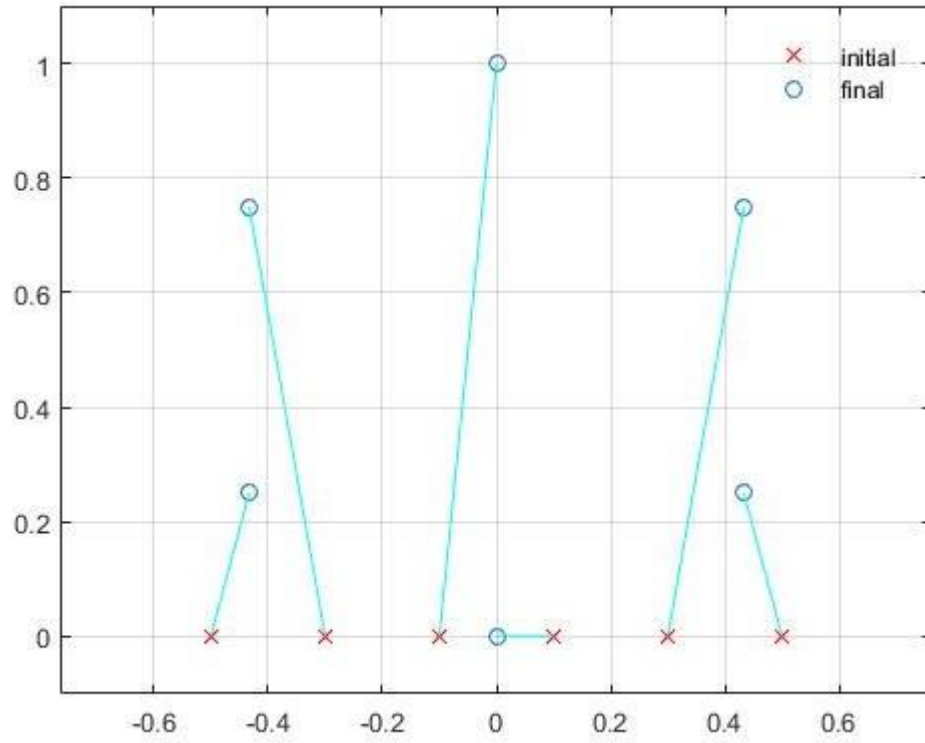
s=MUL_best(noit,2,:);
s=squeeze(s);
[~,ind]=min(s);
OVERALL_best=MUL_best(:,ind);

time=toc;

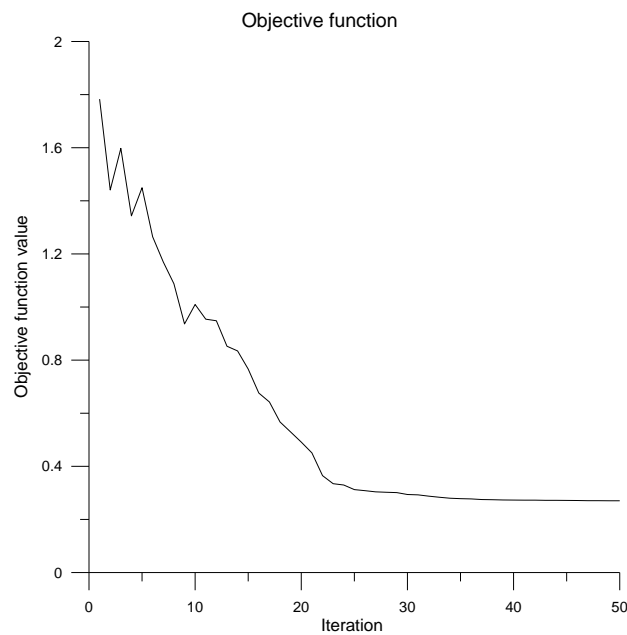
```

| Particle | X initial | Y initial | X final | Y final |
|----------|-----------|-----------|-----------|----------|
| 1 | -0.500000 | 0 | 0 | 0 |
| 2 | -0.300000 | 0 | 0.433012 | 0.250000 |
| 3 | -0.100000 | 0 | 0.433012 | 0.750000 |
| 4 | 0.100000 | 0 | 0 | 1 |
| 5 | 0.300000 | 0 | -0.433012 | 0.750000 |
| 6 | 0.500000 | 0 | -0.433012 | 0.250000 |

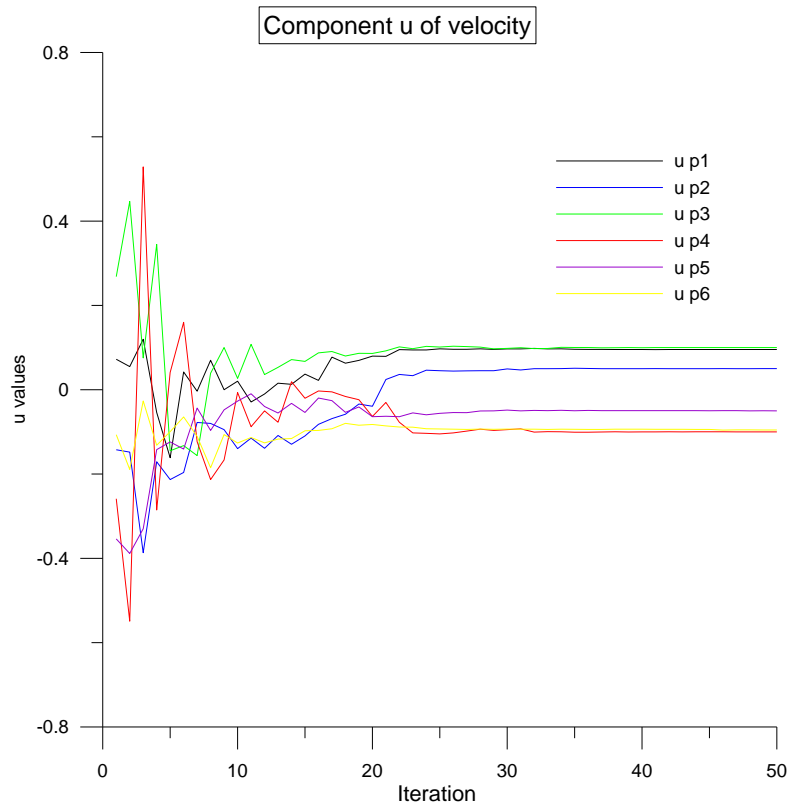
Πίνακας 6.2.1: Οι συντεταγμένες των αρχικών και τελικών σημείων



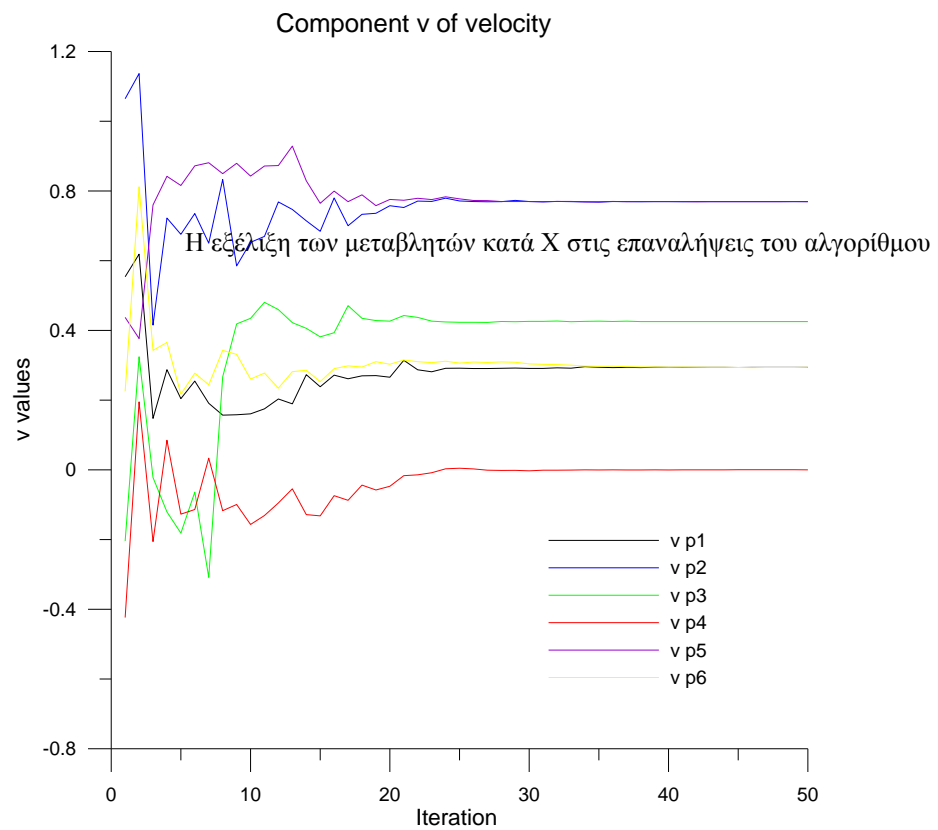
Εικόνα 6.2.1.3: Η γραφική παράσταση ως αποτέλεσμα του κώδικα



Εικόνα 6.2.1.4: Η εξέλιξη της μεταβολής της αντικειμενικής συνάρτησης

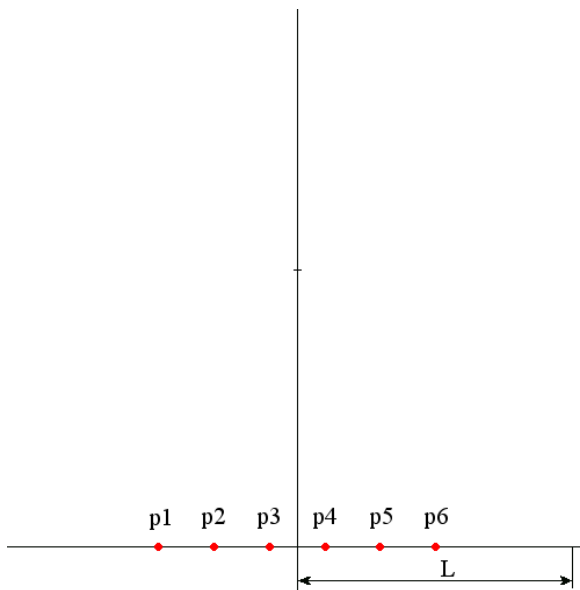


Εικόνα 6.2.1.5: Η εξέλιξη των μεταβλητών κατά X στις επαναλήψεις του αλγορίθμου

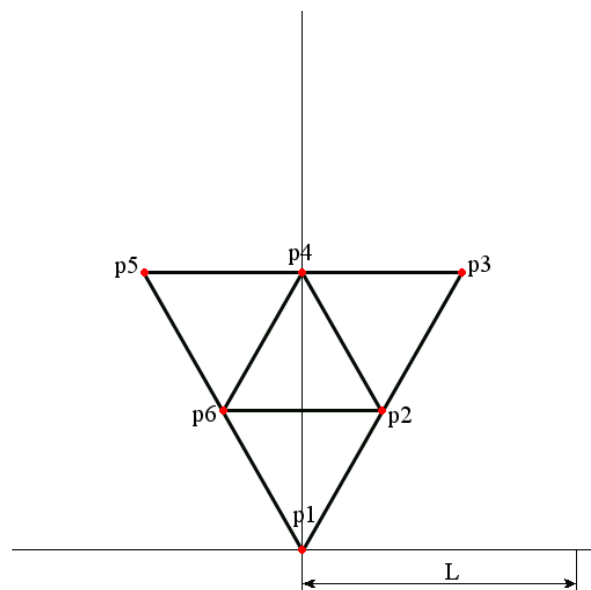


Εικόνα 6.2.1.6: Η εξέλιξη των μεταβλητών κατά Y στις επαναλήψεις του αλγορίθμου

6.2.2 Σχηματισμός Από Ευθεία Σε Τρίγωνο



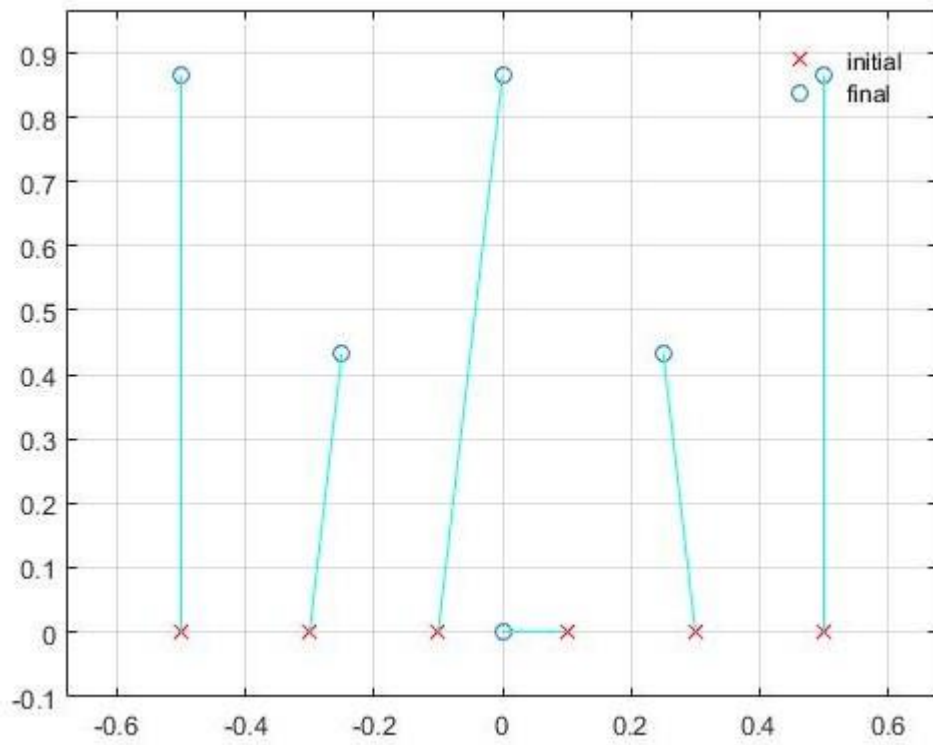
Εικόνα 6.2.2.1: Αρχικός σχηματισμός



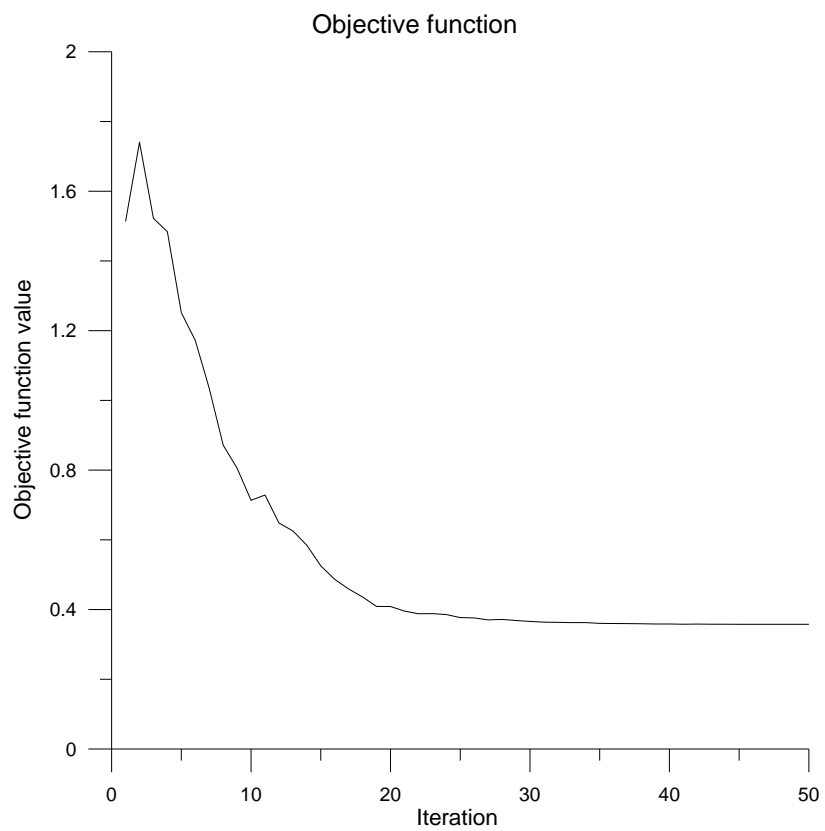
Εικόνα 6.2.2.2: Τελικός σχηματισμός

| Particle | X initial | Y initial | X final | Y final |
|----------|-----------|-----------|-----------|----------|
| 1 | -0.500000 | 0 | 0 | 0 |
| 2 | -0.300000 | 0 | 0.250000 | 0.433012 |
| 3 | -0.100000 | 0 | 0.500000 | 0.866025 |
| 4 | 0.100000 | 0 | 0 | 0.866025 |
| 5 | 0.300000 | 0 | -0.500000 | 0.866025 |
| 6 | 0.500000 | 0 | -0.250000 | 0.433012 |

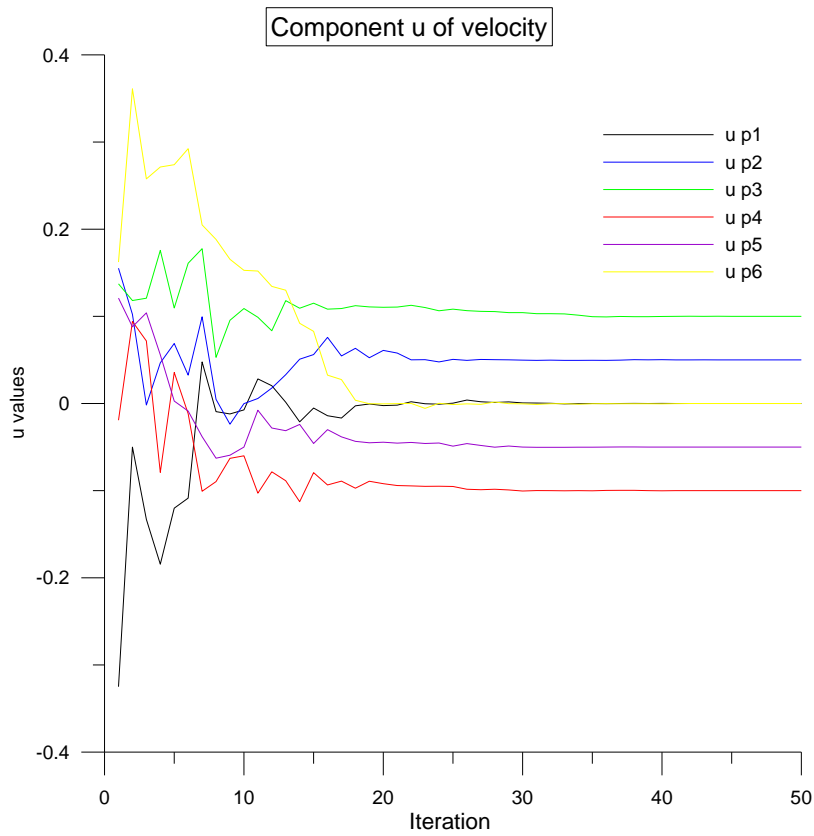
Πίνακας 6.2.2: Οι συντεταγμένες των αρχικών και τελικών σημείων



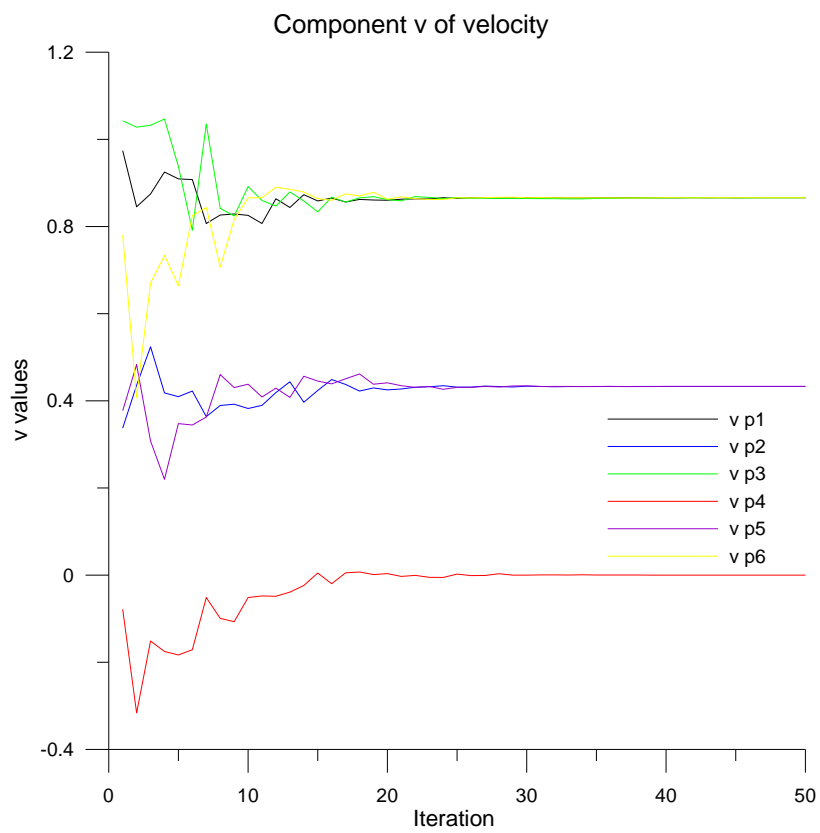
Εικόνα 6.2.2.3: Η γραφική παράσταση ως αποτέλεσμα του κώδικα



Εικόνα 6.2.2.4: Η εξέλιξη της μεταβολής της αντικειμενικής συνάρτησης

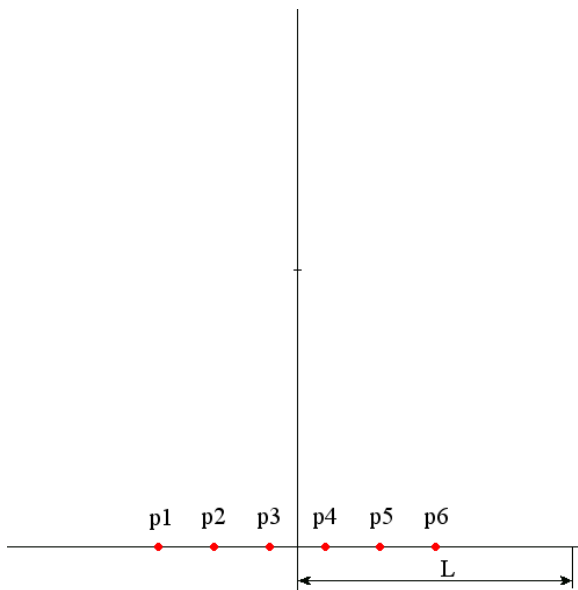


Εικόνα 6.2.2.5: Η εξέλιξη των μεταβλητών κατά X στις επαναλήψεις του αλγορίθμου

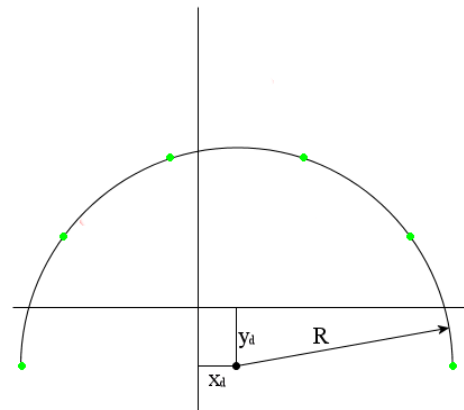


Εικόνα 6.2.2.6: Η εξέλιξη των μεταβλητών κατά Y στις επαναλήψεις του αλγορίθμου

6.2.3 Σχηματισμός Από Ευθεία Σε Ημικύκλιο



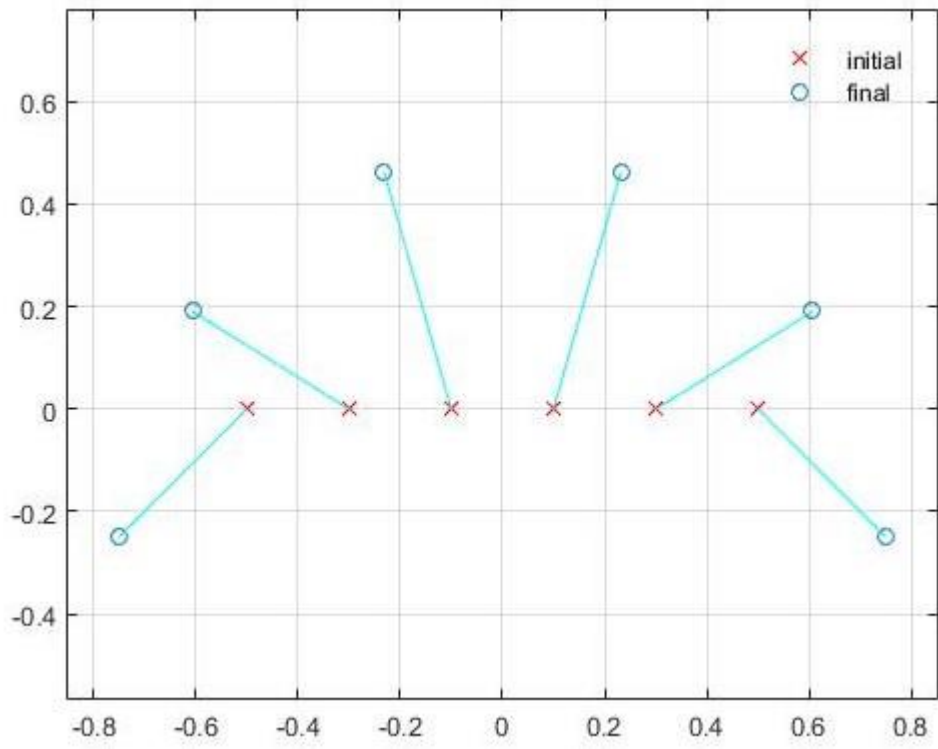
Εικόνα 6.2.3.1: Αρχικός σχηματισμός



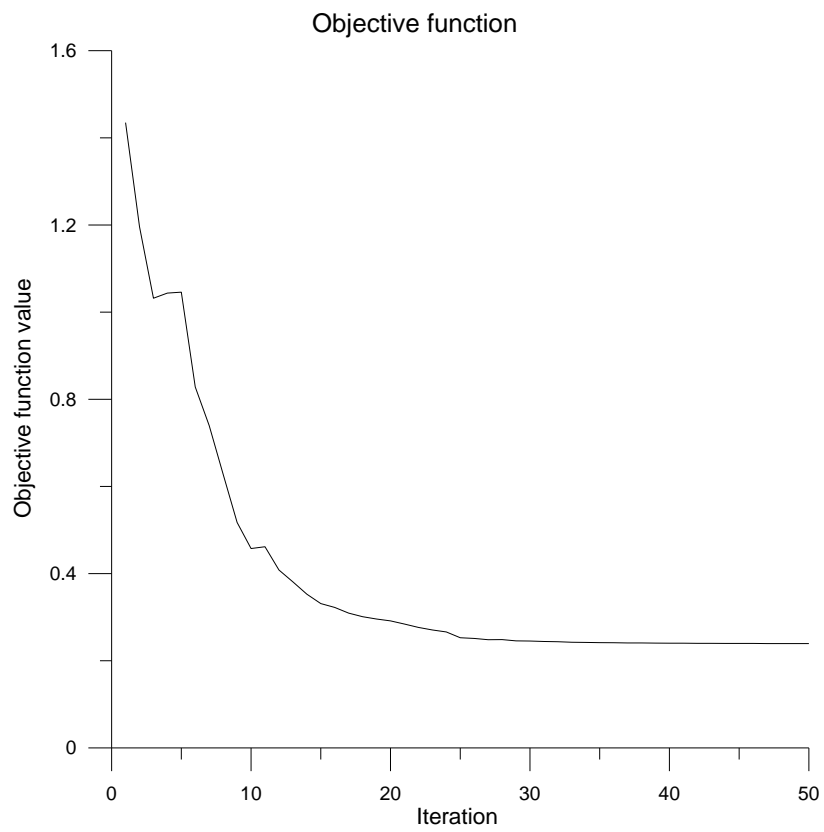
Εικόνα 6.2.3.2: Τελικός σχηματισμός

| Particle | X initial | Y initial | X final | Y final |
|----------|-----------|-----------|-----------|-----------|
| 1 | -0.500000 | 0 | -0.750000 | -0.250000 |
| 2 | -0.300000 | 0 | -0.606762 | 0.190838 |
| 3 | -0.100000 | 0 | -0.231762 | 0.463292 |
| 4 | 0.100000 | 0 | 0.231762 | 0.463292 |
| 5 | 0.300000 | 0 | 0.606762 | 0.190838 |
| 6 | 0.500000 | 0 | 0.750000 | -0.250000 |

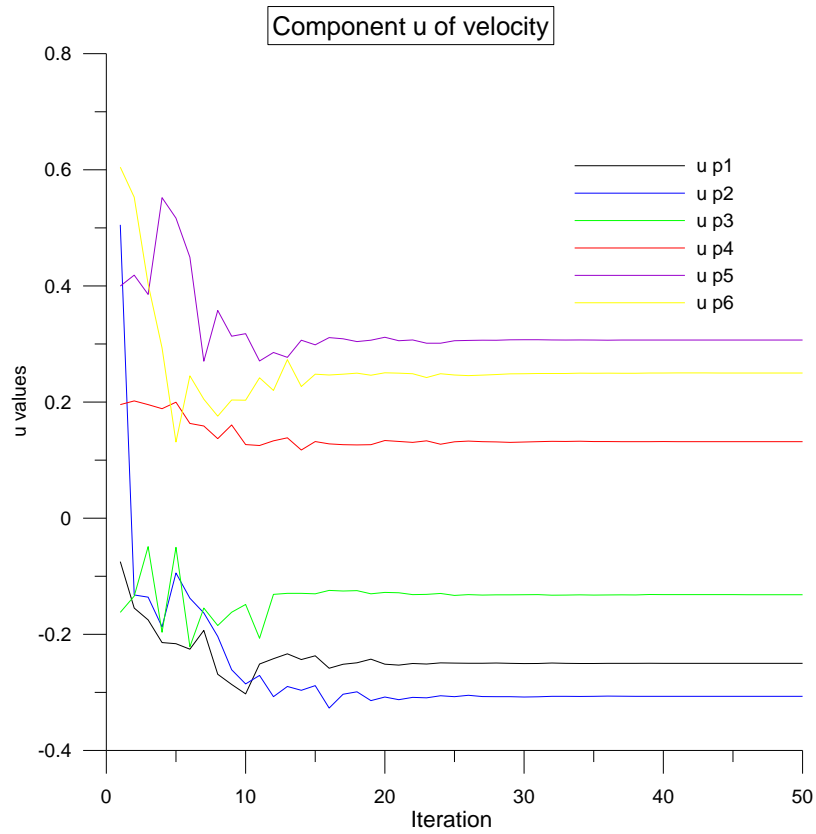
Πίνακας 6.2.3: Οι συντεταγμένες των αρχικών και τελικών σημείων



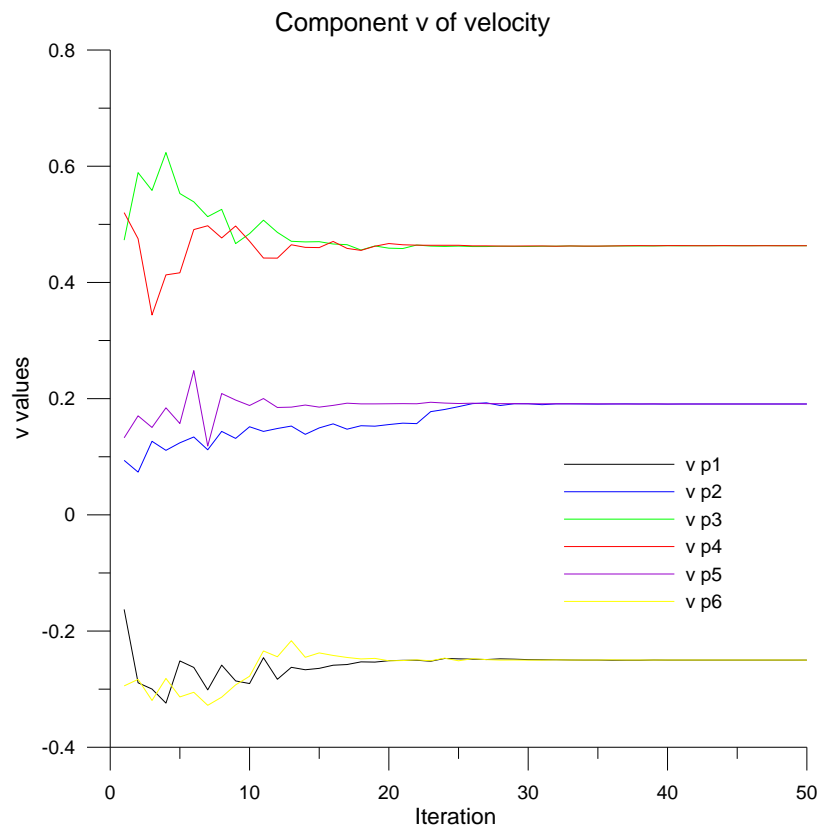
Εικόνα 6.2.3.3: Η γραφική παράσταση ως αποτέλεσμα του κώδικα



Εικόνα 6.2.3.4: Η εξέλιξη της μεταβολής της αντικειμενικής συνάρτησης

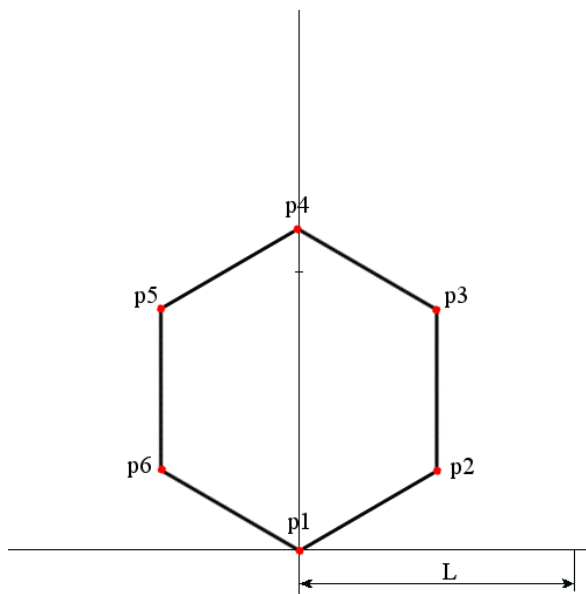


Εικόνα 6.2.3.5: Η εξέλιξη των μεταβλητών κατά X στις επαναλήψεις του αλγορίθμου

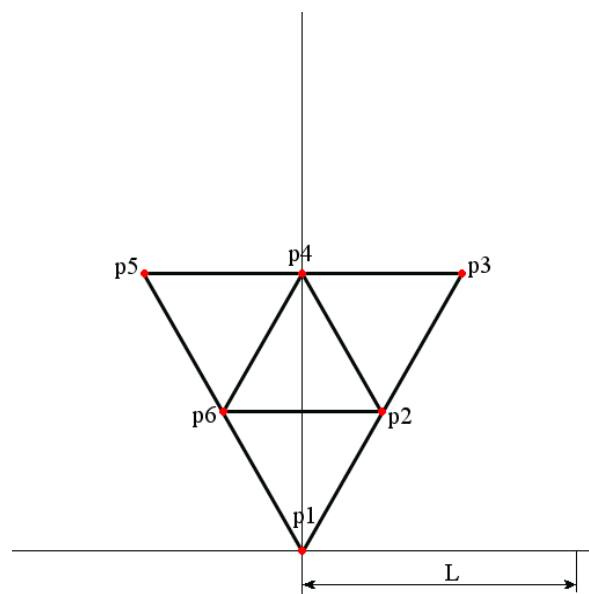


Εικόνα 6.2.3.6: Η εξέλιξη των μεταβλητών κατά Y στις επαναλήψεις του αλγορίθμου

6.2.4 Σχηματισμός Από Εξάγωνο Σε Τρίγωνο



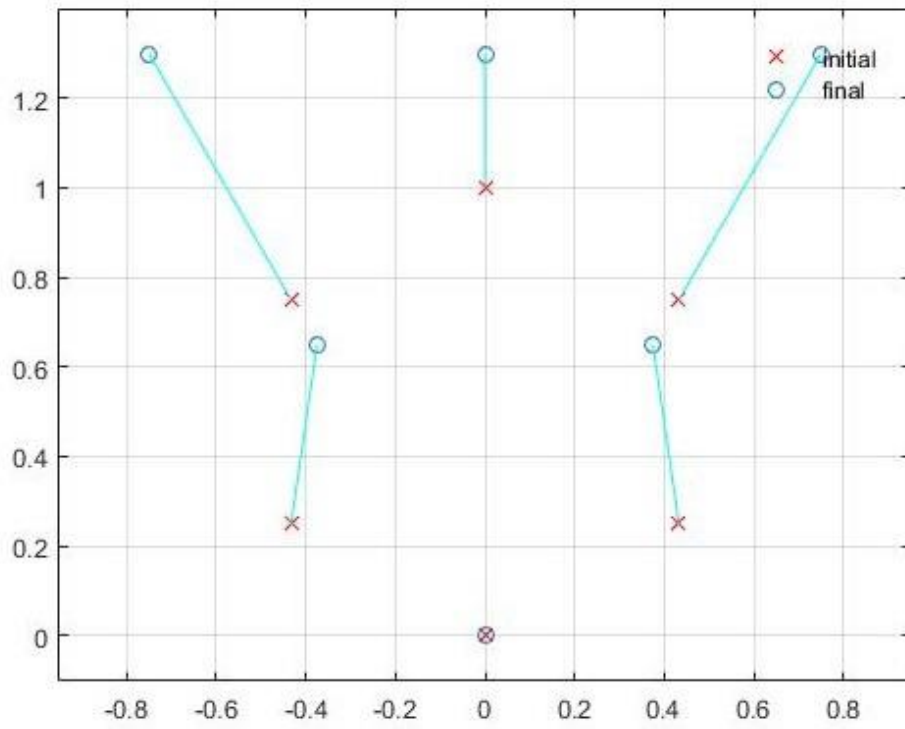
Εικόνα 6.2.4.1: Αρχικός σχηματισμός



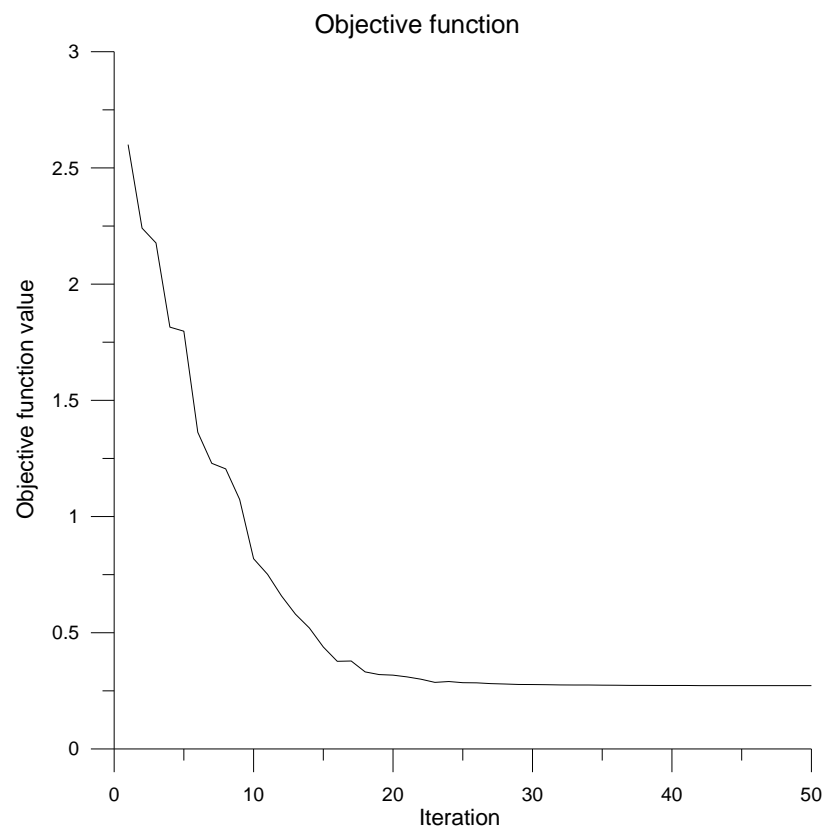
Εικόνα 6.2.4.2: Τελικός σχηματισμός

| Particle | X initial | Y initial | X final | Y final |
|----------|-----------|-----------|-----------|----------|
| 1 | 0 | 0 | 0 | 0 |
| 2 | 0.376721 | 0.217500 | 0.375000 | 0.649519 |
| 3 | 0.376721 | 0.652500 | 0.750000 | 1.299038 |
| 4 | 0 | 0.870000 | 0 | 1.299038 |
| 5 | -0.376721 | 0.652500 | -0.750000 | 1.299038 |
| 6 | -0.376721 | 0.217500 | -0.375000 | 0.649519 |

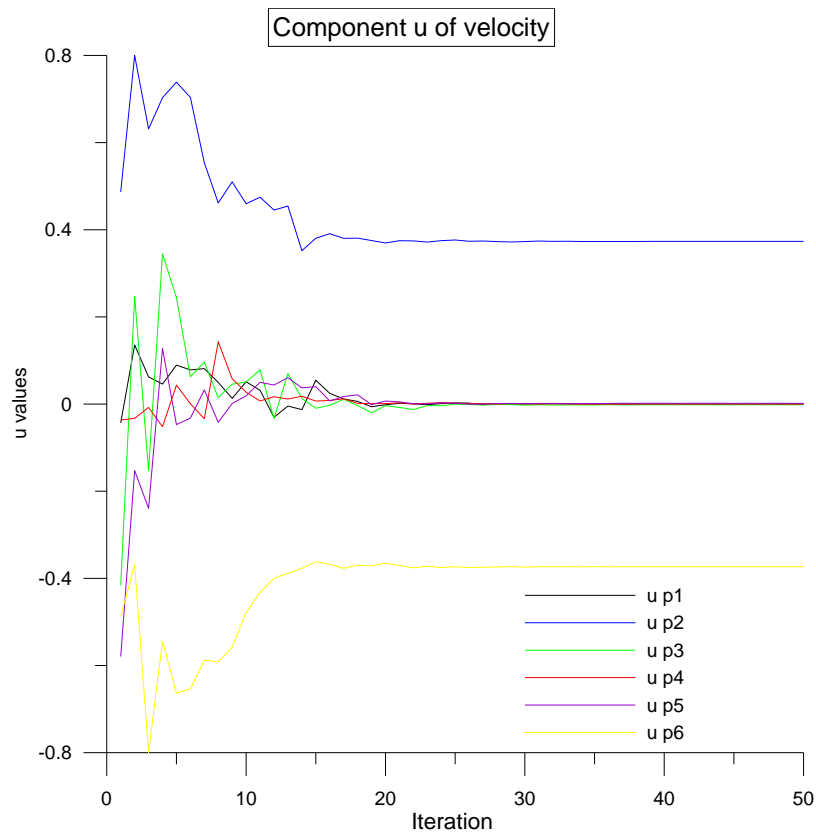
Πίνακας 6.2.4: Οι συντεταγμένες των αρχικών και τελικών σημείων



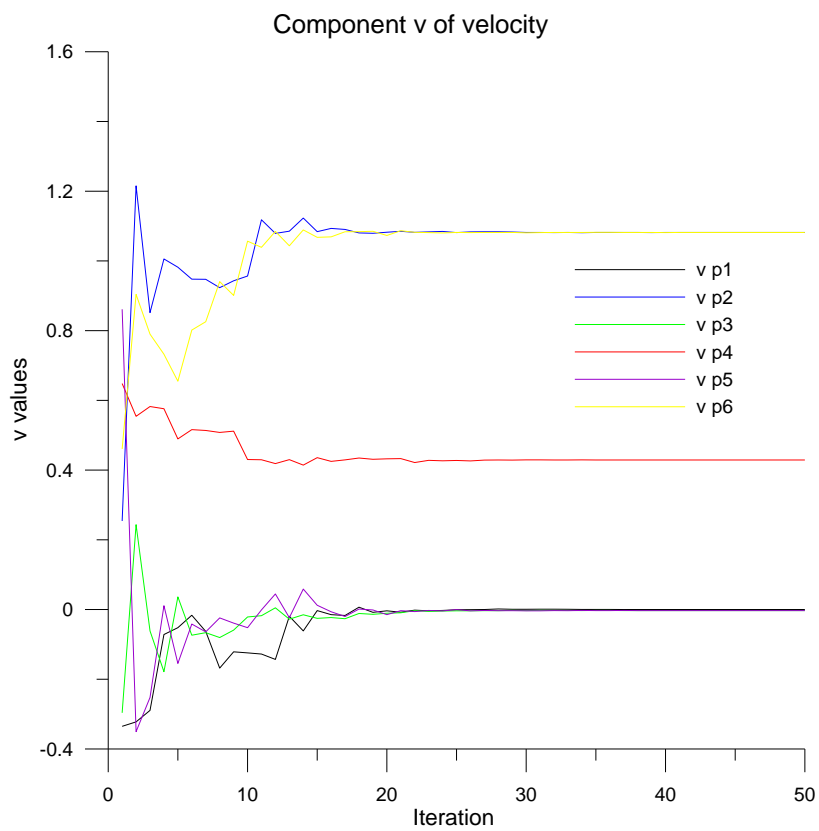
Εικόνα 6.2.4.3: Η γραφική παράσταση ως αποτέλεσμα του κώδικα



Εικόνα 6.2.4.4: Η εξέλιξη της μεταβολής της αντικειμενικής συνάρτησης

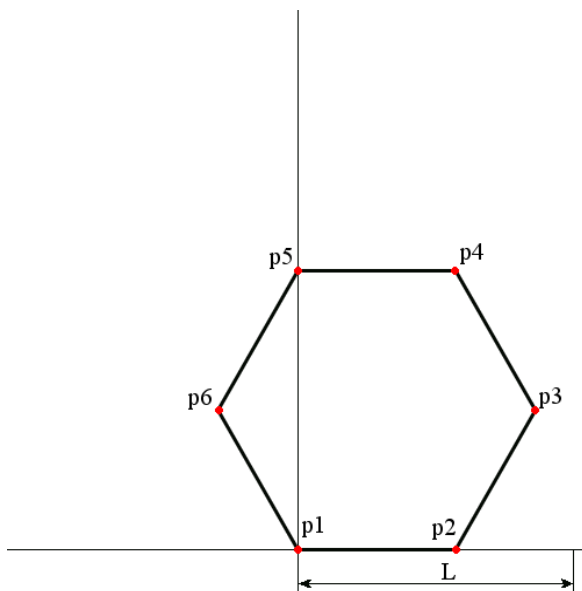


Εικόνα 6.2.4.5: Η εξέλιξη των μεταβλητών κατά X στις επαναλήψεις του αλγορίθμου

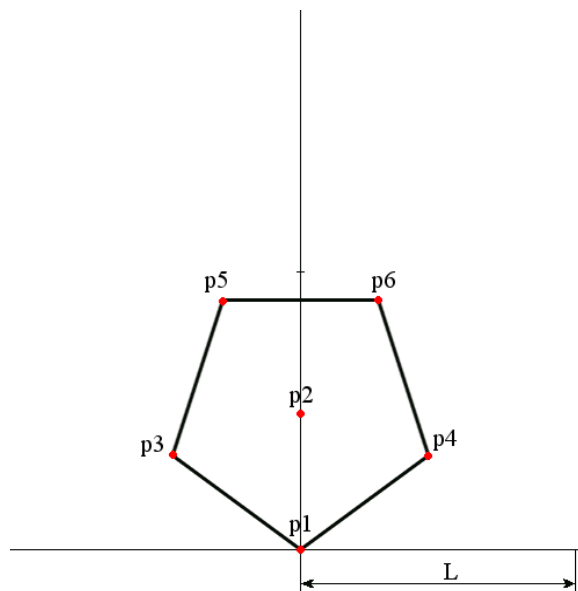


Εικόνα 6.2.4.6: Η εξέλιξη των μεταβλητών κατά Y στις επαναλήψεις του αλγορίθμου

6.2.5 Σχηματισμός Από Μετατοπισμένο Εξάγωνο Σε Πεντάγωνο



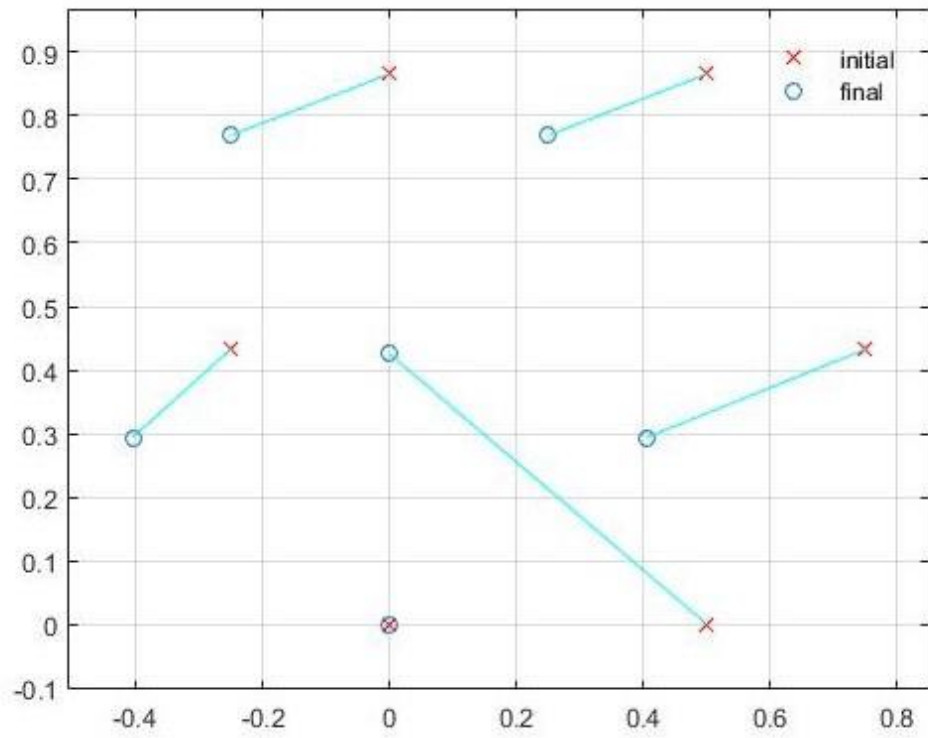
Εικόνα 6.2.5.1: Αρχικός σχηματισμός



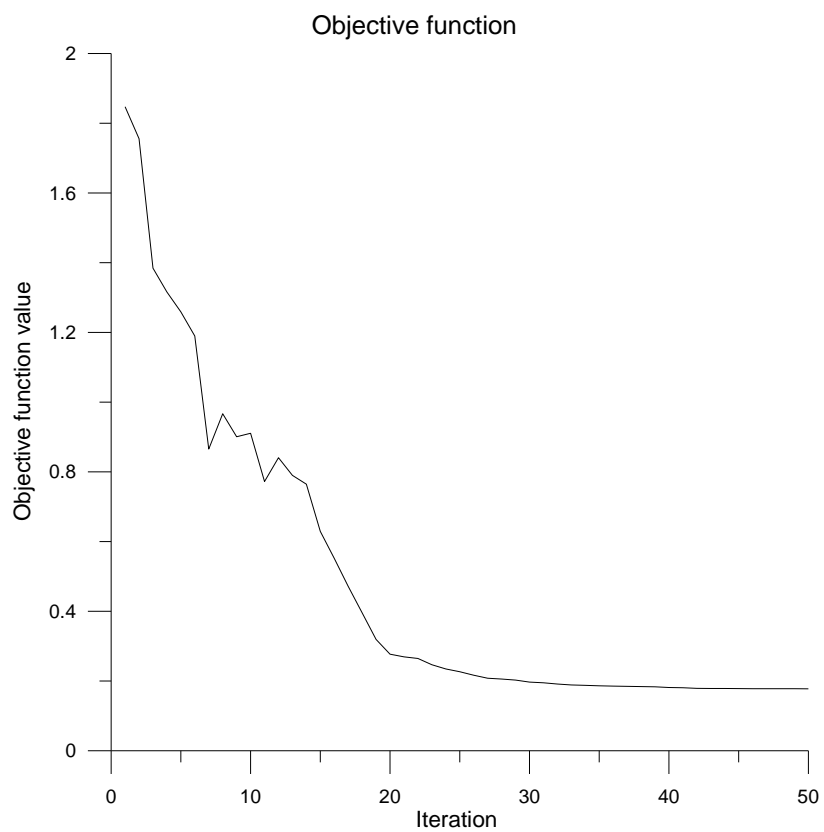
Εικόνα 6.2.5.2: Τελικός σχηματισμός

| Particle | X initial | Y initial | X final | Y final |
|----------|-----------|-----------|-----------|----------|
| 1 | 0 | 0 | 0 | 0 |
| 2 | 0.500000 | 0 | 0 | 0.425325 |
| 3 | 0.750000 | 0.433012 | -0.404508 | 0.293892 |
| 4 | 0.500000 | 0.866025 | 0.404508 | 0.293892 |
| 5 | 0 | 0.866025 | -0.250000 | 0.769420 |
| 6 | -0.250000 | 0.433012 | 0.250000 | 0.769420 |

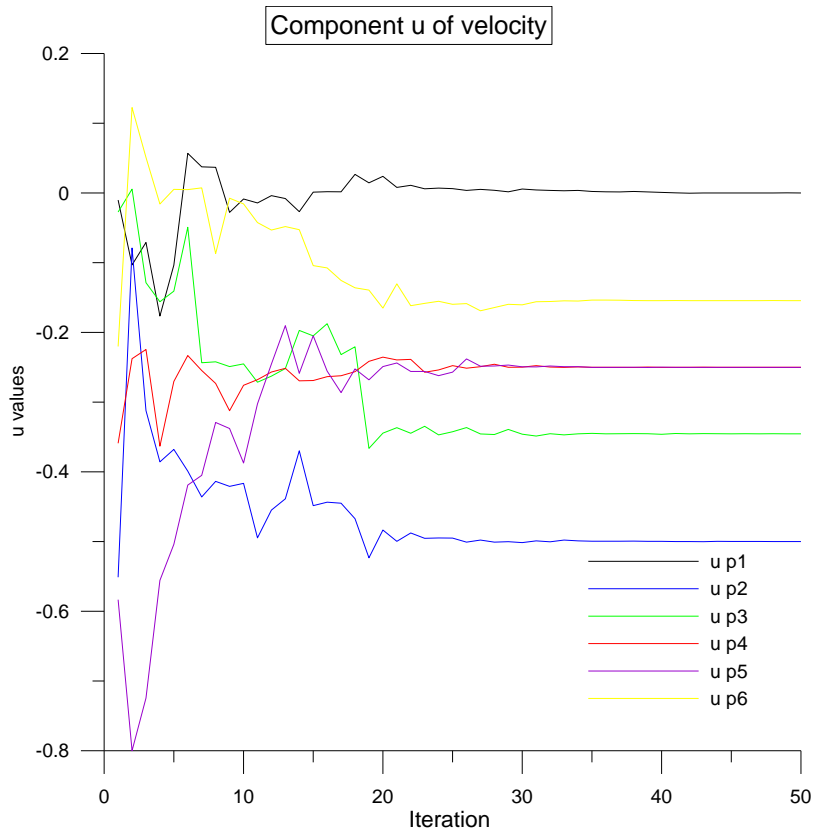
Πίνακας 6.2.5: Οι συντεταγμένες των αρχικών και τελικών σημείων



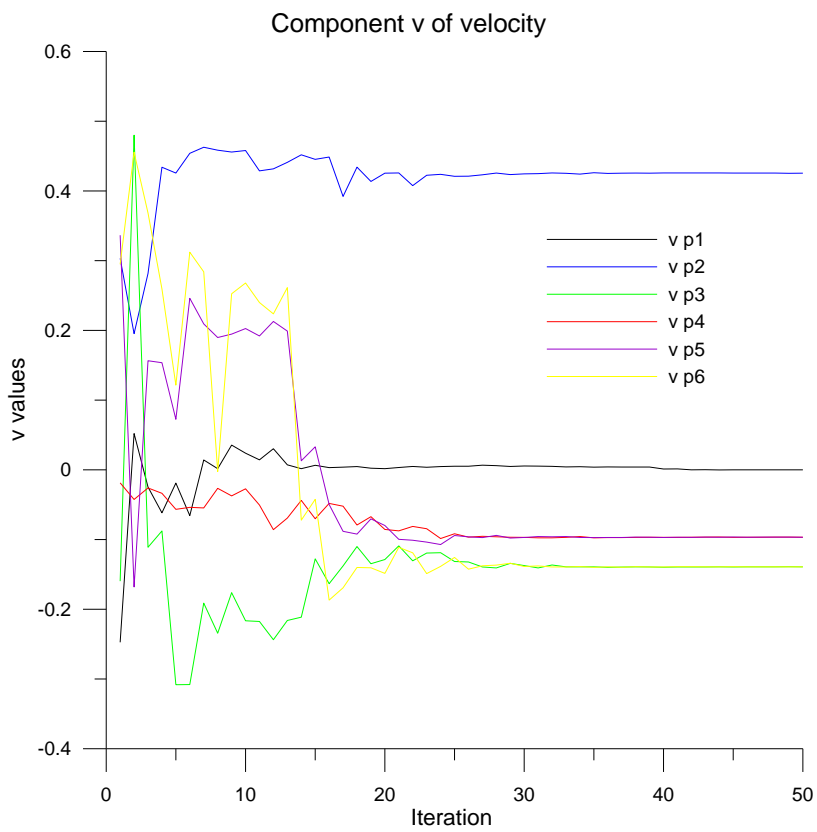
Εικόνα 6.2.5.3: Η γραφική παράσταση ως αποτέλεσμα του κώδικα



Εικόνα 6.2.5.4: Η εξέλιξη της μεταβολής της αντικειμενικής συνάρτησης

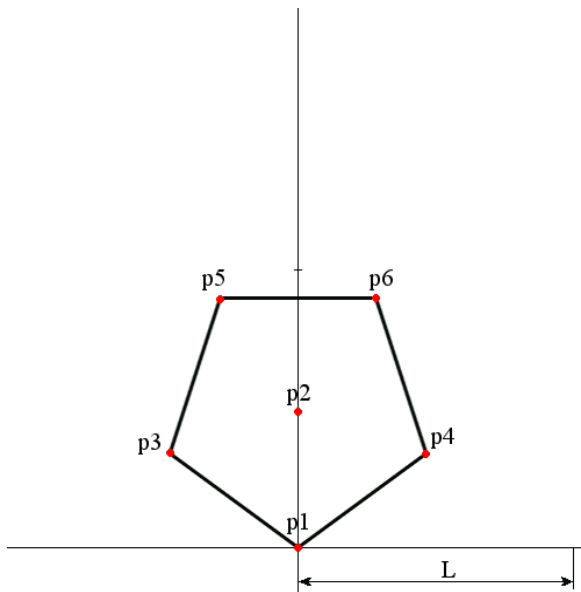


Εικόνα 6.2.5.5: Η εξέλιξη των μεταβλητών κατά X στις επαναλήψεις του αλγορίθμου

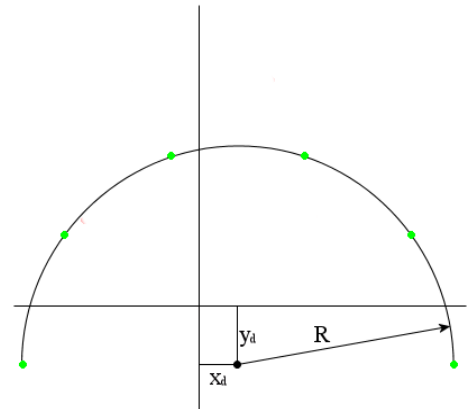


Εικόνα 6.2.5.6: Η εξέλιξη των μεταβλητών κατά Y στις επαναλήψεις του αλγορίθμου

6.2.6 Σχηματισμός Από Πεντάγωνο Σε Ημικόκλιο



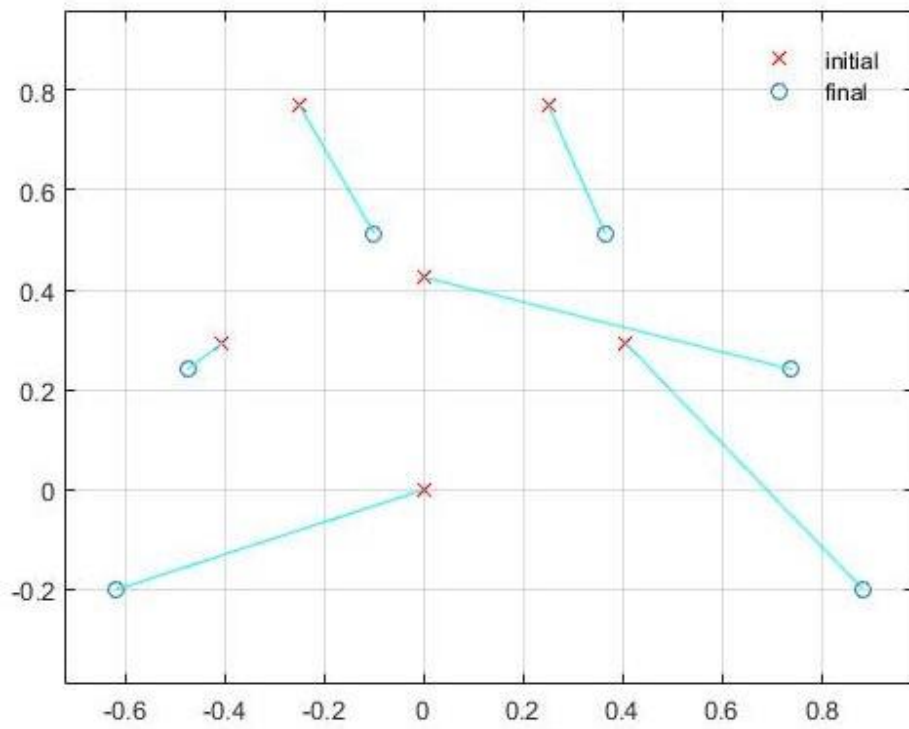
Εικόνα 6.2.6.1: Αρχικός σχηματισμός



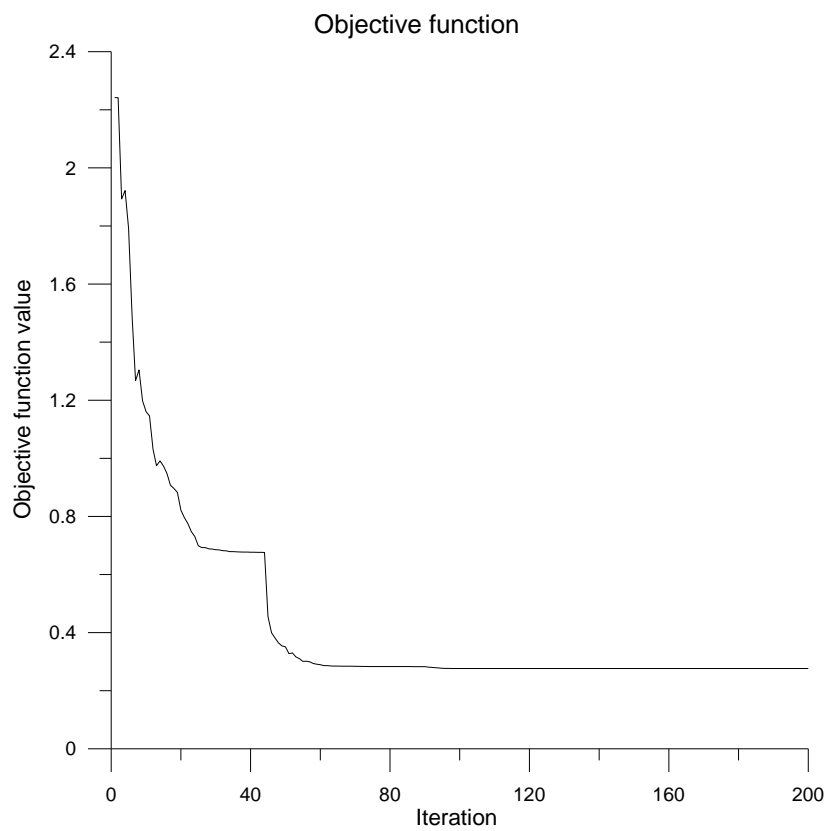
Εικόνα 6.2.6.2: Τελικός σχηματισμός

| Particle | X initial | Y initial | X final | Y final |
|----------|-----------|-----------|----------|-----------|
| 1 | 0 | 0 | -0.618 | -0.200000 |
| 2 | 0 | 0.425325 | -0.47476 | 0.240839 |
| 3 | -0.40451 | 0.293893 | -0.09976 | 0.513292 |
| 4 | 0.404508 | 0.293893 | 0.363763 | 0.513292 |
| 5 | -0.250000 | 0.769421 | 0.738763 | 0.240839 |
| 6 | 0.250000 | 0.769421 | 0.882 | -0.200000 |

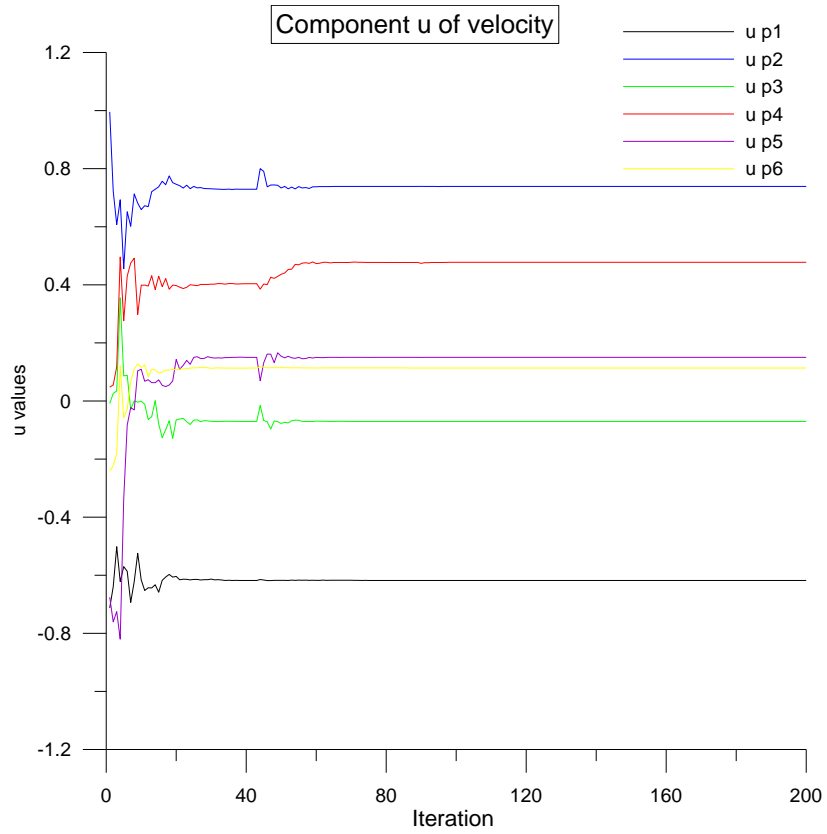
Πίνακας 6.2.6: Οι συντεταγμένες των αρχικών και τελικών σημείων



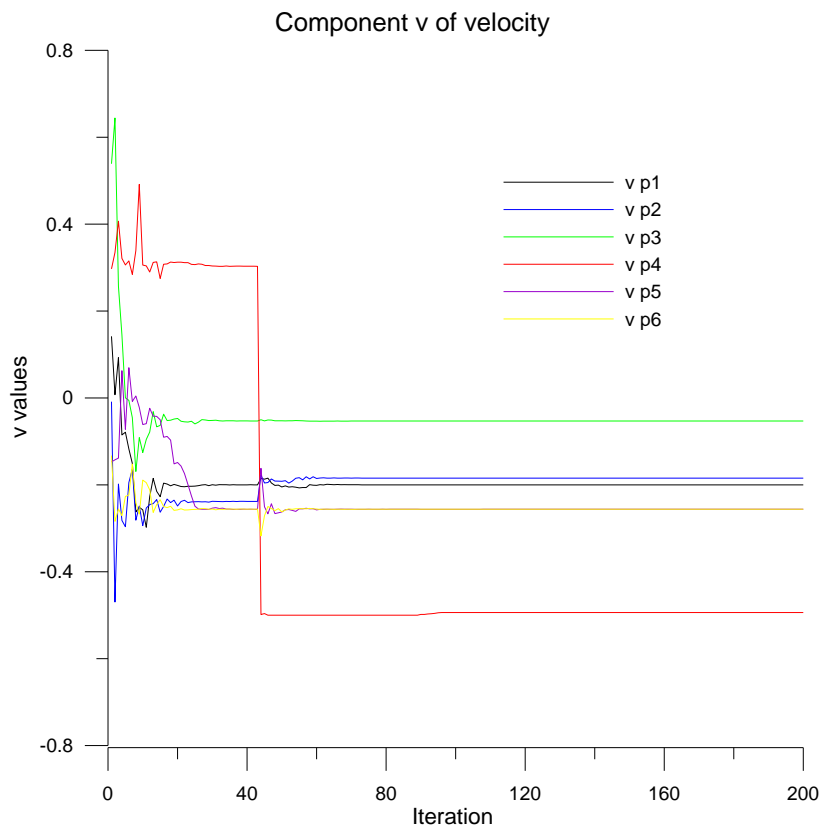
Εικόνα 6.2.6.3: Η γραφική παράσταση ως αποτέλεσμα του κώδικα



Εικόνα 6.2.6.4: Η εξέλιξη της μεταβολής της αντικειμενικής συνάρτησης



Εικόνα 6.2.6.5: Η εξέλιξη των μεταβλητών κατά Y στις επαναλήψεις του αλγορίθμου



Εικόνα 6.2.6.6: Η εξέλιξη των μεταβλητών κατά Y στις επαναλήψεις του αλγορίθμου

Λίστα Εικόνων

| | |
|--|----|
| Εικόνα 1.1: Ένα μη επανδρωμένο αεροσκάφος | 3 |
| Εικόνα 2.2.3: Η βασική λειτουργική δομή ενός τυπικού ΓΑ | 15 |
| Εικόνα 3.1: Ένα σμήνος πουλιών | 17 |
| Εικόνα 4.2: 1) Το πρώτο μυρμήγκι επιλέγει μια διαδρομή προς την τροφή (F) και επιστρέφει στη φωλιά (N) αφήνοντας ίχνη φερομόνης 2) τα άλλα μυρμήγκια ακολουθούν κάποια από τις τέσσερις διαδρομές 3) η διαδρομή με την μεγαλύτερη εναπόθεση φερομόνης , που είναι και η συντομότερη, γίνεται προτιμητέα..... | 21 |
| Εικόνα 5.2.1: Sperry & Hewitt Areal Torpedo 1916..... | 31 |
| Εικόνα 5.2.2: IAI Scout 1978 | 32 |
| Εικόνα 5.2.3: Το UAV Predator | 32 |
| Εικόνα 5.2.4: NASA Helios | 33 |
| Εικόνα 5.2.5: NASA Altum | 33 |
| Εικόνα 5.3.1: RQ-16 T Hawk..... | 34 |
| Εικόνα 5.3.2: Pioneer | 34 |
| Εικόνα 5.3.3: AeroVironment | 35 |
| Εικόνα 5.3.4: Aeronautics Defense Dominator | 35 |
| Εικόνα 5.3.5: Black Hawk..... | 35 |
| Εικόνα 5.7.1: Οι άξονες της γης..... | 39 |
| Εικόνα 5.7.2.1: Ένα τυπικό σύστημα αξόνων xyz | 40 |
| Εικόνα 5.7.2.2: Οι άξονες του αεροσκάφους | 41 |
| Εικόνα 5.7.2.3: σωματόδετο σύστημα αξόνων | 42 |
| Εικόνα 5.8.2: η επικοινωνία μεταξύ των UAV | 44 |
| Εικόνα 5.9.1: Ένα γυροσκόπιο | 47 |
| Εικόνα 5.9.2: Ένα επιταχυνσιόμετρο | 48 |
| Εικόνα 5.9.3: Ένα Μαγνητόμετρο | 48 |
| Εικόνα 5.9.4: Ένα Βαρόμετρο/Αισθητήρας Πίεσης | 49 |
| Εικόνα 5.9.5: Ένα GPS..... | 49 |
| Εικόνα 5.10: Ένα ελεγκτής πτήσης | 50 |
| Εικόνα 5.11: Συνεργασία UAVs με GPS, το οποίο αναμεταδίδει τη θέση του στον ελεγκτή πτήσης..... | 51 |

| | |
|--|----|
| Εικόνα 6.1.1.1: Η συνάρτηση beale είναι πολυτροπική, με κοφτερές κορυφές στις γωνίες του πεδίου εισόδου..... | 52 |
| Εικόνα 6.1.1.2: Η εξέλιξη της μεταβολής της συνάρτησης | 53 |
| Εικόνα 6.1.2.1: Η συνάρτηση Eggholder είναι μια δύσκολη συνάρτηση για τη βελτιστοποίηση, λόγω του μεγάλου αριθμού των τοπικών ελαχίστων..... | 53 |
| Εικόνα 6.1.2.2: Η εξέλιξη της μεταβολής της συνάρτησης | 54 |
| Εικόνα 6.1.3.1: Η συνάρτηση Ackley χρησιμοποιείται ευρέως για τον έλεγχο των αλγορίθμων βελτιστοποίησης. Στη δισδιάστατη μορφή, όπως φαίνεται στην παραπάνω γραφική χαρακτηρίζεται από μια σχεδόν επίπεδη εξωτερική περιφέρεια, καθώς και μια μεγάλη τρύπα στο κέντρο. Η συνάρτηση ενέχει κίνδυνο για τους αλγόριθμους βελτιστοποίησης, ιδιαίτερα για τους hillclimbing αλγόριθμους, να παγιδευτούν σε ένα από τα πολλά τοπικά ελάχιστα της. | 54 |
| Εικόνα 6.1.3.2: Η εξέλιξη της μεταβολής της συνάρτησης | 55 |
| Εικόνα 6.1.4.1: Η συνάρτηση Booth Function..... | 56 |
| Εικόνα 6.1.4.2: Η εξέλιξη της μεταβολής της συνάρτησης | 57 |
| Εικόνα 6.1.5.1: Η συνάρτηση Rosenbrock, που αναφέρεται επίσης ως συνάρτηση κοιλάδα ή μπανάνα, είναι ένα δημοφιλές παράδειγμα λειτουργίας για αλγορίθμους βελτιστοποίησης που βασίζονται σε κλίση. Φαίνεται και από τη γραφική πάνω στη δισδιάστατη μορφή. Η συνάρτηση είναι μονοτροπική και το σφαιρικό ελάχιστο έγκειται σε μια στενή, παραβολική κοιλάδα. Ωστόσο, ακόμα κι αν η κοιλάδα είναι εύκολο να βρεθεί, η σύγκλιση στο ελάχιστο είναι δύσκολο (Picheny et al., 2012). | 57 |
| Εικόνα 6.1.5.2: Η εξέλιξη της μεταβολής της συνάρτησης | 58 |
| Εικόνα 6.2.1.1: Αρχικός Σχηματισμός | 59 |
| Εικόνα 6.2.1.2: Τελικός Σχηματισμός | 59 |
| Εικόνα 6.2.1.3: Η γραφική παράσταση ως αποτέλεσμα του κώδικα | 67 |
| Εικόνα 6.2.1.4: Η εξέλιξη της μεταβολής της αντικειμενικής συνάρτησης | 67 |
| Εικόνα 6.2.1.5: Η εξέλιξη των μεταβλητών κατά X στις επαναλήψεις του αλγορίθμου | 68 |
| Εικόνα 6.2.1.6: Η εξέλιξη των μεταβλητών κατά Y στις επαναλήψεις του αλγορίθμου..... | 68 |
| Εικόνα 6.2.2.1: Αρχικός Σχηματισμός | 69 |
| Εικόνα 6.2.2.2: Τελικός Σχηματισμός | 69 |

| | |
|---|----|
| Εικόνα 6.2.2.3: Η γραφική παράσταση ως αποτέλεσμα του κώδικα | 70 |
| Εικόνα 6.2.2.4: Η εξέλιξη της μεταβολής της αντικειμενικής συνάρτησης | 70 |
| Εικόνα 6.2.2.5: Η εξέλιξη των μεταβλητών κατά X στις επαναλήψεις του αλγορίθμου | 71 |
| Εικόνα 6.2.2.6: Η εξέλιξη των μεταβλητών κατά Y στις επαναλήψεις του αλγορίθμου..... | 71 |
| Εικόνα 6.2.3.1: Αρχικός σχηματισμός | 72 |
| Εικόνα 6.2.3.2: Τελικός σχηματισμός | 72 |
| Εικόνα 6.2.3.3: Η γραφική παράσταση ως αποτέλεσμα του κώδικα | 73 |
| Εικόνα 6.2.3.4: Η εξέλιξη της μεταβολής της αντικειμενικής συνάρτησης | 73 |
| Εικόνα 6.2.3.5: Η εξέλιξη των μεταβλητών κατά X στις επαναλήψεις του αλγορίθμου..... | 74 |
| Εικόνα 6.2.3.6: Η εξέλιξη των μεταβλητών κατά Y στις επαναλήψεις του αλγορίθμου..... | 74 |
| Εικόνα 6.2.4.1: Αρχικός σχηματισμός | 75 |
| Εικόνα 6.2.4.2: Τελικός σχηματισμός | 75 |
| Εικόνα 6.2.4.3: Η γραφική παράσταση ως αποτέλεσμα του κώδικα | 76 |
| Εικόνα 6.2.4.4: Η εξέλιξη της μεταβολής της αντικειμενικής συνάρτησης..... | 76 |
| Εικόνα 6.2.4.5: Η εξέλιξη των μεταβλητών κατά X στις επαναλήψεις του αλγορίθμου..... | 77 |
| Εικόνα 6.2.4.6: Η εξέλιξη των μεταβλητών κατά Y στις επαναλήψεις του αλγορίθμου..... | 77 |
| Εικόνα 6.2.5.1: Αρχικός σχηματισμός | 78 |
| Εικόνα 6.2.5.2: Τελικός σχηματισμός | 78 |
| Εικόνα 6.2.5.3: Η γραφική παράσταση ως αποτέλεσμα του κώδικα | 79 |
| Εικόνα 6.2.5.4: Η εξέλιξη της μεταβολής της αντικειμενικής συνάρτησης..... | 79 |
| Εικόνα 6.2.5.5: Η εξέλιξη των μεταβλητών κατά X στις επαναλήψεις του αλγορίθμου..... | 80 |
| Εικόνα 6.2.5.6: Η εξέλιξη των μεταβλητών κατά Y στις επαναλήψεις του αλγορίθμου..... | 80 |
| Εικόνα 6.2.6.1: Αρχικός σχηματισμός | 81 |
| Εικόνα 6.2.6.2: Τελικός σχηματισμός | 81 |

| | |
|--|----|
| Εικόνα 6.2.6.3: Η γραφική παράσταση ως αποτέλεσμα του κώδικα | 82 |
| Εικόνα 6.2.6.4: Η εξέλιξη της μεταβολής της αντικειμενικής συνάρτησης..... | 82 |
| Εικόνα 6.2.6.5: Η εξέλιξη των μεταβλητών κατά X στις επαναλήψεις του αλγορίθμου..... | 83 |
| Εικόνα 6.2.6.6: Η εξέλιξη των μεταβλητών κατά Y στις επαναλήψεις του αλγορίθμου..... | 83 |

Λίστα Πινάκων

| | |
|---|----|
| Πίνακας 6.2.1: Οι συντεταγμένες των αρχικών και τελικών σημείων | 67 |
| Πίνακας 6.2.2: Οι συντεταγμένες των αρχικών και τελικών σημείων | 69 |
| Πίνακας 6.2.3: Οι συντεταγμένες των αρχικών και τελικών σημείων | 72 |
| Πίνακας 6.2.4: Οι συντεταγμένες των αρχικών και τελικών σημείων | 75 |
| Πίνακας 6.2.5: Οι συντεταγμένες των αρχικών και τελικών σημείων | 78 |
| Πίνακας 6.2.6: Οι συντεταγμένες των αρχικών και τελικών σημείων | 81 |

BIBΛΙΟΓΡΑΦΙΑ

- Aerial Navigation and Navigating Instruments H.N. Heaton Bureau of Standards
- Engelbrecht, A. P. (2006). Fundamentals of Computational Swarm Intelligence, John Wiley & Sons.
- Dorigo M. & Stützle T. (2004). Ant Colony Optimization, MIT Press.
- Dorigo, M., & Stützle, T. (2003). The ant colony optimization metaheuristic: Algorithms, applications, and advances. In: Handbook of Metaheuristics (pp. 250-285). Springer US.
- Yang, X. S. (2010). Engineering Optimization.
- Davidovic, T., Teodorovic, D., & Selmic, M. (2015). Bee colony optimization Part I: The algorithm overview, Yugoslav Journal of Operations Research
- Mitchell, M.: An Introduction to Genetic Algorithms. MIT Press, (1998)
- Altringham, J. D.: Bats: Biology and Behaviour, Oxford University Press, (1996)
- J. Kennedy, and R. C. Eberhart. (1995). Particle swarm optimization. In Proceedings of the 1995 IEEE International Conference on Neural Networks, Vol. 4, pp. 1942-1948
- Parunak, V., M. Purcell, and R. O'Connell. "Digital Pheromones for Autonomous Coordination of Swarming UAVs." American Institute of Aeronautics and Aerospace, 2002
- Parsopoulos, K. E., & Vrahatis, M. N. (2010). Particle Swarm Optimization and Intelligence: Advances and Applications, IGI Global
- Αντωνιάδης Ι. (2015), Δυναμική και έλεγχος πτήσης
- G. Wang, Z. Chen, Z. Yu C. Dong, "A Method of Self-Adaptive Inertia Weight for PSO", International Conference on Computer Science and Software Engineering, vol. 1, pp. 1195 - 1198, 2008
- M. Abramowitz and I. A. Stegun, Handbook of Mathematical Functions, Dover Publication, 1965.

- A. Afshar, O. B. Haddad, M. A. Marino, B. J. Adams, "Honey-bee mating optimization (HBMO) algorithm for optimal reservoir operation", J. Franklin Institute, 344, 452-462 (2007).
- M. A. El-Beltagy, Keane A. J., "A comparison of various optimization algorithms on a multilevel problem", Engin. Appl. Art. Intell., 12, 639-654 (1999).
- P. E. Gill, W. Murray, and M. H. Wright, Practical optimization, Academic Press Inc, 1981.
- Owen, M., R. Beard, and T. McLain: Implementing dubins airplane paths on fixed-wing uavs. In Valavanis, K. P. and G. J. Vachtsevanos (editors): Handbook of Unmanned Aerial Vehicles, pages 1677–1701. Springer Netherlands, 2014, ISBN 978-90-481-9706-4
- A. J. Wood, B. F. Woolenber, Power Generation, Operation and Control. Wiley, 1996.
- Houck, C., Joines, J., and Kay M., 1995, A Genetic Algorithm for Function Optimization: A Matlab Implementation, ACM Transactions on Mathematical Software, Submitted 1996
- Kennedy, J., Eberhart, R., Shi, Y., 2001, Swarm Intelligence, Academic Press, Inc.
- Gerardo Beni, "From Swarm Intelligence to Swarm Robotics," Swarm Robotics LNCS 3342, Springer-Verlag, Berlin, pp. 1-9, 2005.
- Venu G. Gudise and Ganesh K. Venayagamoorthy, "Comparison of Particle Swarm Optimization and Backpropagation as Training Algorithms for Neural Networks," Proceedings of 2003 IEEE Swarm Intelligence Symposium, pp. 110117, April 2003
- Bouabdallah, S., Noth, A. and Siegwart, R. (2004) PID vs LQ Control Techniques Applied to an Indoor Micro Quadrotor. Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2004), Vol. 3, 28 September-2 October 2004, 2451-2456.
- ZHANG Jinhua. Modified adaptive PSO algorithm based on cloud theory [J]. Computer Engineering and Applications, 2012, 48(5):29-31

Ηλεκτρονικές Πηγές

- https://en.wikipedia.org/wiki/Unmanned_aerial_vehicle
- <https://en.wikipedia.org/wiki/Gyroscope>
- <https://en.wikipedia.org/wiki/Accelerometer>
- <https://en.wikipedia.org/wiki/Magnetometer>
- <https://en.wikipedia.org/wiki/Barometer>
- https://en.wikipedia.org/wiki/Global_Positioning_System
- https://en.wikipedia.org/wiki/Test_functions_for_optimization
- <http://www.ee.washington.edu/research/pstca/>
- <http://www.cs.ndsu.nodak.edu/~nygard/research/munitions.pdf>
- <http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=6289431>
- http://read.pudn.com/downloads63/sourcecode/windows/other/222276/%E7%B2%92%E5%AD%90%E7%BE%A4%E4%BC%98%E5%8C%96%E7%AE%97%E6%B3%95%EF%BC%88PSO%EF%BC%89%E5%BA%94%E7%94%A8%E4%BA%8E%E7%A5%9E%E7%BB%8F%E7%BD%91%E7%BB%9C%E4%BC%98%E5%8C%96%E7%A8%8B%E5%BA%8F/DemoTrainPSO.m_.htm
- <http://www.sfu.ca/~ssurjano/optimization.html>
- <http://www.wikiwand.com/el/%CE%9D%CE%BF%CE%B7%CE%BC%CE%BF%CF%83%CF%8D%CE%BD%CE%B7%CF%83%CE%BC%CE%AE%CE%BD%CE%BF%CF%85%CF%82>
- <http://www.nrl.navy.mil/research/nrl-review/2004/simulation-computing-modeling/oh/>
- http://www-optima.amp.i.kyoto-u.ac.jp/member/student/hedar/Hedar_files/TestGO_files/Page364.htm
- <http://www.swarmintelligence.org/>
- <http://www.theuav.com/>

