



ΣΤΡΑΤΙΩΤΙΚΗ ΣΧΟΛΗ ΕΥΕΛΠΙΔΩΝ

Τμήμα Στρατιωτικών Επιστημών

ΕΛΛΗΝΙΚΗ ΔΗΜΟΚΡΑΤΙΑ

ΔΙΔΡΥΜΑΤΙΚΟ ΔΙΑΤΜΗΜΑΤΙΚΟ

ΠΡΟΓΡΑΜΜΑ ΜΕΤΑΠΤΥΧΙΑΚΩΝ ΣΠΟΥΔΩΝ

ΣΧΕΔΙΑΣΗ & ΕΠΕΞΕΡΓΑΣΙΑ ΣΥΣΤΗΜΑΤΩΝ
(SYSTEMS ENGINEERING)

(ΠΔ 96 /2015/ΦΕΚ 163Α'/20.08.2014)



ΠΟΛΥΤΕΧΝΕΙΟ ΚΡΗΤΗΣ

Σχολή Μηχανικών Παραγωγής & Διοίκησης

ΜΕΤΑΠΤΥΧΙΑΚΗ ΔΙΑΤΡΙΒΗ

Σχεδιασμός και βελτιστοποίηση τροχιάς μη στελεχωμένων
εναέριων συστημάτων μέσω εξελικτικών αλγορίθμων για
πραγματικά χαρτογραφικά δεδομένα

Κωνσταντίνος Πλατανίτης

A.M. 2017018015

Φεβρουάριος 2023

Η Μεταπτυχιακή Διατριβή του κ. Πλατανίτη Κωνσταντίνου εγκρίνεται:

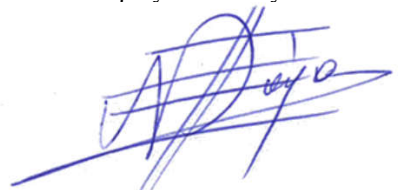
ΤΡΙΜΕΛΗΣ ΕΞΕΤΑΣΤΙΚΗ ΕΠΙΤΡΟΠΗ

Καθηγητής ΟΝΟΜΑΤΕΠΩΝΥΜΟ (Επιβλέπων) Τσουρβελούδης Νικόλαος

Nikolaos Tsourveloudis Nikolaos Tsourveloudis
16.02.2023 00:59

Καθηγητής ΟΝΟΜΑΤΕΠΩΝΥΜΟ

Δάρας Νικόλαος



Καθηγητής ΟΝΟΜΑΤΕΠΩΝΥΜΟ

Δοϊτσίδης Ελευθέριος

Eleftherios Doitsidis Eleftherios Doitsidis
16.02.2023 08:31

Πίνακας περιεχομένων

| | |
|--|----|
| Πρόλογος | 6 |
| Κεφάλαιο 1 - Εισαγωγή | 8 |
| 1.1 Σχεδιασμός τροχιάς | 8 |
| 1.2 Κατηγοριοποίηση τρόπων επίλυσης | 9 |
| 1.3 Ανασκόπηση μεθόδων επίλυσης..... | 10 |
| 1.3.1 Γεωμετρική προσέγγιση - Τροχιές Dubins | 10 |
| 1.3.2 Μέθοδος με B-Splines..... | 13 |
| 1.3.3 Μέθοδος πεδίων δυναμικού | 16 |
| Κεφάλαιο 2 - Προαπαιτούμενα | 18 |
| 2.1 Εισαγωγή..... | 18 |
| 2.2 Αναπαράσταση ανάγλυφου | 18 |
| 2.2.1 Δεδομένα | 18 |
| 2.2.2 Τριγωνοποίηση | 19 |
| 2.2.3 Παρεμβολή..... | 22 |
| 2.3 Μετατροπή συστημάτων συντεταγμένων..... | 24 |
| 2.4 Αναπαράσταση τροχιάς | 25 |
| 2.5 Αριθμητική επίλυση Διαφορικών Εξισώσεων (ΔΕ)..... | 28 |
| 2.6 Γενετικοί αλγόριθμοι | 30 |
| 2.6.1 Ονοματολογία..... | 31 |
| 2.7 Σύνοψη..... | 33 |
| Κεφάλαιο 3 ^ο – Μηχανισμός δημιουργίας τροχιάς..... | 34 |
| 3.1 Εισαγωγή..... | 34 |
| 3.2 Περιγραφή της διαδικασίας προσομοίωσης..... | 34 |
| 3.3 Σύνοψη..... | 41 |
| Κεφάλαιο 4 ^ο – Μεθοδολογία με γενετικούς αλγόριθμους..... | 42 |
| 4.1 Εισαγωγή..... | 42 |
| 4.2 Ο γενετικός αλγόριθμος..... | 42 |
| 4.3 Συνάρτηση καταλληλότητας..... | 45 |
| 4.4 Σύνοψη..... | 51 |
| Κεφάλαιο 5 ^ο – Εφαρμογή με πραγματικά χαρτογραφικά δεδομένα | 52 |

| | |
|---|----|
| 5.1 Εισαγωγή..... | 52 |
| 5.2 Χαρτογραφικά δεδομένα..... | 52 |
| 5.3 Συντομότερη δυνατή τροχιά..... | 53 |
| 5.4 Τροχιά με δύο κριτήρια | 55 |
| 5.5 Μια γενική λύση | 57 |
| 5.6 Σύνοψη..... | 58 |
| Κεφάλαιο 6 – Επέκταση σε πολλαπλά οχήματα (AIROUS)..... | 59 |
| 6.1 Εισαγωγή..... | 59 |
| 6.2 Πολλαπλά οχήματα..... | 59 |
| 6.3 Επιπλέον χωρικοί περιορισμοί | 61 |
| 6.4 Λοιπές μεταβολές | 61 |
| 6.5 Ενδεικτικά αποτελέσματα | 63 |
| 6.5.1 Ad-hoc εμπόδια | 63 |
| 6.5.2 Σχεδιασμός αποστολής με πολλαπλά οχήματα | 66 |
| 6.6 Σύνοψη..... | 69 |
| Κεφάλαιο 7ο - Επίλογος | 70 |
| 7.1 Σύνοψη αποτελεσμάτων..... | 70 |
| 7.2 Μελλοντικές εργασίες | 71 |
| Βιβλιογραφία | 72 |

Πρόλογος

Η εργασία αυτή αποτελεί μεταπτυχιακή εργασία στο πλαίσιο του ΔΔΠΜΣ «Σχεδίαση & Επεξεργασία Συστημάτων» της Σχολής Μηχανικών Παραγωγής και Διοίκησης του Πολυτεχνείου Κρήτης και της Στρατιωτικής Σχολής Ευελπίδων.

Αντικείμενο μελέτης της παρούσας εργασίας είναι μία καινούρια μεθοδολογία για τον σχεδιασμό τροχιάς ενός αυτόνομου, μη επανδρωμένου/στελεχωμένου οχήματος η οποία βασίζεται στη βελτιστοποίηση καμπύλων που αναπαριστούν τροχιά μέσω γενετικών αλγορίθμων, ιδέα που αρχικά δημοσιεύτηκε το 2003 [18]. Η μέθοδος που έχει αναπτυχθεί στη παρούσα εργασία αποσκοπεί στην επίλυση προβλημάτων σχεδιασμού τροχιάς με πραγματικά χαρτογραφικά δεδομένα. Βασικό σχεδιαστικό χαρακτηριστικό της μεθόδου είναι η επεκτασιμότητά της.

Η αρχική πραγματοποίηση της μεθόδου έχει παρουσιαστεί στο συνέδριο ICUAS¹ 2020 [1] και στο πλαίσιο του project AIROUS² επεκτάθηκε κατάλληλα για να μπορεί να υποστηρίξει την ταυτόχρονη επίλυση του προβλήματος για πολλά εναέρια οχήματα. Οι τροποποιήσεις και τα αντίστοιχα αποτελέσματα παρουσιάστηκαν στις [2] [3].

Το project AIROUS χρηματοδοτήθηκε στα πλαίσια του Horizon 2020³, και συγκεκριμένα από το UFO project (Innovation boosted by Small Flying Objects)⁴, ενώ μέρος της εργασίας αυτής έχει υποστηριχθεί από τον ΕΛΚΕ του Πολυτεχνείου Κρήτης, μέσω ερευνητικών έργων.

¹ <https://www.uasconferences.com>

² <https://airous.eu>

³ <https://ec.europa.eu/programmes/horizon2020/>

⁴ <https://www.ufoproject.eu/>

Στην εργασία θα παρατεθούν και θα σχολιαστούν σημεία όπως τα παρακάτω:

- Τι είναι ο σχεδιασμός τροχιάς, και γιατί είναι απαραίτητος σε μια αποστολή μη επανδρωμένου οχήματος
- Ανασκόπηση επιλεγμένων μεθόδων επίλυσης του προβλήματος σχεδιασμού τροχιάς
- Εισαγωγή στα προαπαιτούμενα για την ανάπτυξη της μεθοδολογίας
- Περιγραφή και θεωρητική ανάπτυξη της προτεινόμενης μεθοδολογίας
- Υλοποίηση προγραμματιστικών εργαλείων για την υποστήριξη της μεθοδολογίας
- Δοκιμαστικά σενάρια - αξιολόγηση αποτελεσμάτων
- Project AIROUS & επέκταση της μεθοδολογίας σε σμήνος οχημάτων
- Γενική σύνοψη και αξιολόγηση

Κεφάλαιο 1 - Εισαγωγή

1.1 Σχεδιασμός τροχιάς

Ο σχεδιασμός τροχιάς (path planning) είναι ένα πρόβλημα που εμφανίζεται πάντα κατά την χρήση ενός αυτόνομου οχήματος, ενώ φυσικά μπορεί να χρησιμοποιηθεί και για τον καθορισμό μιας τροχιάς και την προετοιμασία μιας αποστολής για ένα όχημα το οποίο δεν θα λειτουργεί αυτόνομα. Στην περίπτωση της αυτόνομης λειτουργίας η τροχιά θα πρέπει να ακολουθηθεί από κάποιο κατάλληλο σύστημα ελέγχου, ενώ εάν πρόκειται για μη αυτόνομο όχημα ο χειριστής είναι υπεύθυνος για τον έλεγχο του οχήματος με βάση την προσχεδιασμένη τροχιά.

Γενικός στόχος του σχεδιασμού τροχιάς για ένα όχημα είναι η εύρεση μιας διαδρομής η οποία συνδέει δύο σημεία, χωρίς να αποκλείεται ωστόσο και ο προσδιορισμός επιπλέον ενδιάμεσων σημείων μεταξύ της αρχής και του τέλους από τα οποία το όχημα θα πρέπει να περάσει. Σε περίπτωση που κάποια μέθοδος υλοποίησης δεν επιτρέπει αυτή την λειτουργία, ένα πρόβλημα σχεδιασμού τροχιάς με N ενδιάμεσα σταθερά σημεία μπορεί να μετατραπεί σε $N + 1$ ξεχωριστά προβλήματα, και να βρεθεί και πάλι μια απάντηση που θα ικανοποιεί την αρχική απαίτηση.

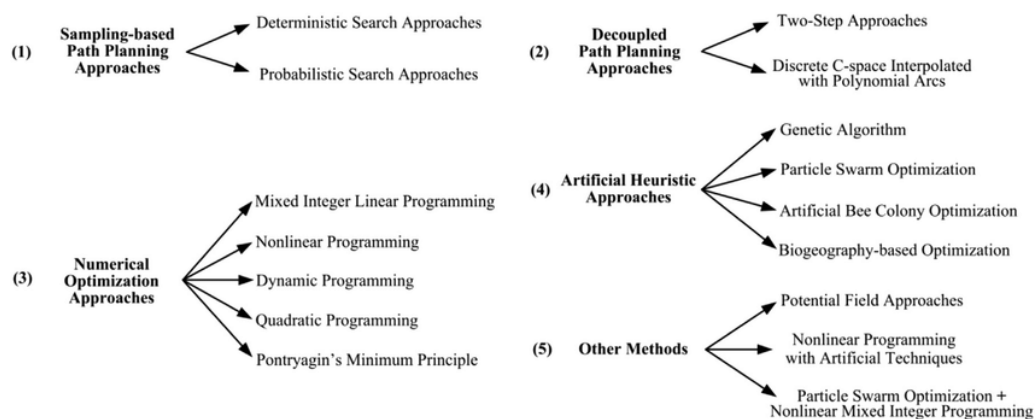
Οι γενικοί στόχοι όταν υπολογίζεται μια τροχιά για κάποιο όχημα συνήθως περιλαμβάνουν τα εξής σημεία:

- Δημιουργία μιας ασφαλούς τροχιάς, τόσο ως προς το όχημα και τις δυνατότητες παρακολούθησης αυτής (πχ φάκελος πτήσης στην περίπτωση αεροσκαφών), όσο και ως προς το περιβάλλον μέσα στο οποίο θα ενεργεί το όχημα (πχ αποφυγή συγκρούσεων)
- Ελαχιστοποίηση των ενεργειακών πόρων που χρειάζονται για την πραγματοποίηση της αποστολής από το όχημα μέσω κατάλληλης διαμόρφωσης της τροχιάς
- Επίτευξη mission-specific στόχων (πχ μεγιστοποίηση περιοχής κάλυψης για εφαρμογές έρευνας και διάσωσης, προσέγγιση συγκεκριμένων σημείων ενδιαφέροντος για εφαρμογές περιπολίας, ...)

Επομένως, για την επίλυση του προβλήματος του σχεδιασμού τροχιάς, θα πρέπει να λυθεί το βασικό πρόβλημα της αποφυγής εμποδίων [4] [5] [6] και στην συνέχεια υπάρχουν διάφορες προσεγγίσεις που μπορούν να εφαρμοσθούν για την βελτιστοποίηση των λύσεων ως προς διάφορους παράγοντες με κοινό παρονομαστή την ενέργεια [7] [8] [9] [10].

1.2 Κατηγοριοποίηση τρόπων επίλυσης

Οι διαφορετικές μεθοδολογίες που υπάρχουν εκμεταλλεύονται διαφορετικά εργαλεία και τεχνικές, και συνήθως αποσκοπούν στην εύρεση μιας τροχιάς η οποία ικανοποιεί κάποιους περιορισμούς ή εμφανίζει κάποια επιθυμητά χαρακτηριστικά, ή ακόμη και αποφεύγει κάποια ανεπιθύμητα [11]. Μια τυπική κατηγοριοποίηση των πιθανών τρόπων υλοποίησης του σχεδιασμού τροχιάς με βάση τα εργαλεία που χρησιμοποιούνται για την βελτιστοποίηση των αποτελεσμάτων παρουσιάζεται στην Εικόνα 1:



Εικόνα 1 - Κατηγοριοποίηση μεθόδων path planning [12]

Η μεθοδολογία που θα παρατεθεί σε αυτή την εργασία, με βάση την παραπάνω κατηγοριοποίηση, εμπεριέχεται στο (4) «Artificial Heuristic Approaches» και πιο συγκεκριμένα στην πρώτη υποκατηγορία «Genetic Algorithm», λόγω της χρήσης γενετικών αλγορίθμων για την βελτιστοποίηση των παραγόμενων λύσεων.

Εκτός της κατηγοριοποίησης με βάση το είδος των εργαλείων βελτιστοποίησης που χρησιμοποιούνται, ένας άλλος τρόπος διαχωρισμού των τρόπων επίλυσης εξετάζει την δυνατότητα χρήσης κατά τη διάρκεια μιας αποστολής (online), ή την αποκλειστική χρήση κατά την προετοιμασία της αποστολής (offline). Η μεθοδολογία που θα παρουσιαστεί εδώ συγκαταλέγεται κατά κύριο λόγο στην δεύτερη κατηγορία (offline) καθώς έχει μεγάλες απαιτήσεις σε υπολογιστικό χρόνο και επεξεργαστική ισχύ, και επομένως δεν είναι εύκολο να τρέξει από onboard υπολογιστικές μονάδες οι οποίες έχουν περιορισμένη απόδοση και αποθέματα ενέργειας. Επίσης, η μεθοδολογία δεν έχει σχεδιαστεί ώστε να υποδέχεται ενημερώσεις σε πραγματικό χρόνο των εμπλεκόμενων παραμέτρων, επομένως δεν είναι εύκολη η εκτέλεση της σε πραγματικό χρόνο.

1.3 Ανασκόπηση μεθόδων επίλυσης

Για την επίλυση του προβλήματος σχεδιασμού τροχιάς υπάρχουν πολλές μεθοδολογίες που έχουν ήδη προταθεί, από κλασσικές προσεγγίσεις καθαρά γεωμετρικές (πχ τροχιές Dubins) μέχρι προσεγγίσεις δημιουργίας τροχιών με χρήση εξελικτικών αλγορίθμων [13] [14] και νευρωνικών δικτύων [15]. Ειδικότερα, θα παρατεθεί μια ανασκόπηση της μεθόδου Dubins καθώς αποτελεί την πεπατημένη, και μεθόδων με εξελικτικούς αλγορίθμους καθώς έχουν συνάφεια με την μεθοδολογία που θα αναπτυχθεί στην εργασία αυτή.

1.3.1 Γεωμετρική προσέγγιση - Τροχιές Dubins

Οι τροχιές Dubins είναι μια μεθοδολογία για την δημιουργία τροχιάς σε 2 διαστάσεις, η οποία δημιουργεί την παράγει μια τροχιά χρησιμοποιώντας ευθύγραμμα και κυκλικά τμήματα με συγκεκριμένους περιορισμούς στην καμπυλότητα. Έχει προταθεί και αποδειχθεί ότι εάν χρησιμοποιηθούν κυκλικά τμήματα μέγιστης καμπυλότητας (όριο που εξαρτάται από το όχημα) τότε η τροχιά Dubins μεταξύ δύο σημείων είναι η συντομότερη πιθανή τροχιά που αποτελείται από τμήματα σταθερής καμπυλότητας και που ικανοποιεί και τον περιορισμό να δίνει συγκεκριμένη γωνία heading (εφαπτομένη συγκεκριμένου προσανατολισμού) στα δύο ακραία σημεία [16].

Η χρήση των τροχιών Dubins περιορίζεται κυρίως σε 2 διαστάσεις, και μπορεί να εύκολα να χρησιμοποιηθεί στην περίπτωση που το κινηματικό μοντέλο του οχήματος εμπεριέχει μια σταθερή γραμμική ταχύτητα V και αντίστοιχα η κατεύθυνση του μπορεί να μεταβληθεί με κάποια σταθερή γωνιακή ταχύτητα ω , ένα μοντέλο που μπορεί να αντιπροσωπεύσει ένα όχημα ή ένα αεροσκάφος που κινείται σε δύο διαστάσεις.

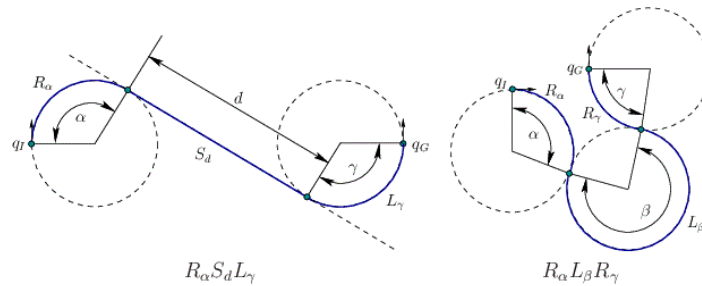
Το παρακάτω κινηματικό μοντέλο είναι γνωστό και ως “Dubins car”, και δείχνει ακριβώς αυτά τα χαρακτηριστικά:

$$\dot{x} = V \cos \theta \quad (1a)$$

$$\dot{y} = V \sin \theta \quad (1b)$$

$$\dot{\theta} = \omega \quad (1c)$$

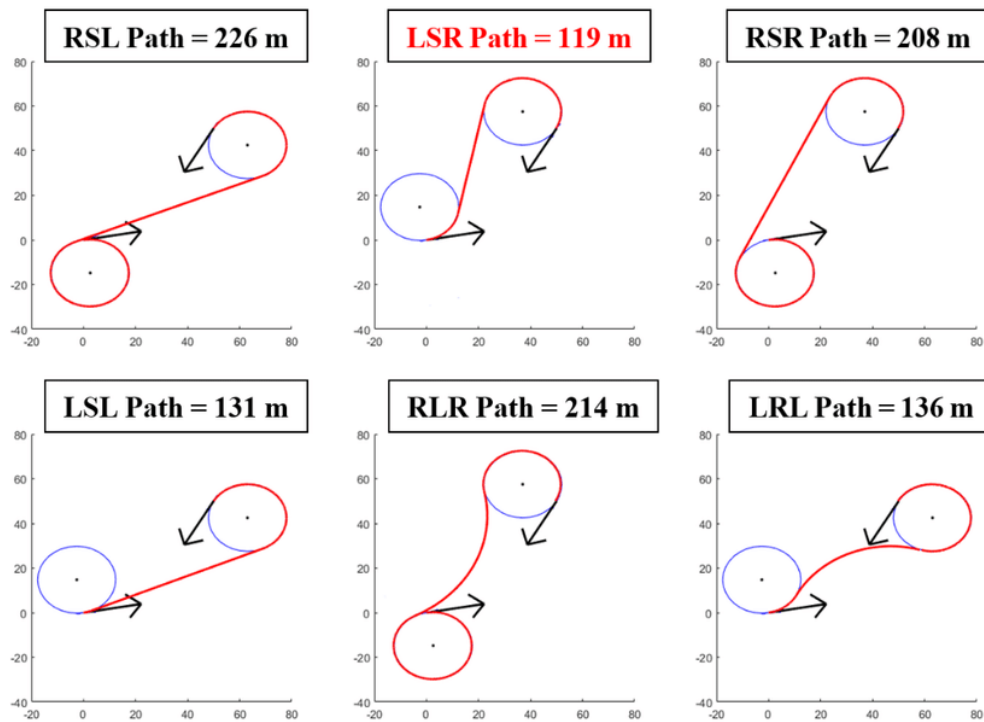
Η τροχιά Dubins μεταξύ δύο σημείων μπορεί να δημιουργηθεί με καθαρά γεωμετρικό τρόπο, σχηματίζοντας κάποιους βοηθητικούς κύκλους οι οποίοι είναι εφαπτόμενοι στην αρχική και την τελική θέση σε σχέση με τους επιθυμητούς προσανατολισμούς, καθώς και κάποιες βοηθητικές ευθείες εφαπτόμενες στους κύκλους αυτούς όπου αυτές χρειάζονται (Εικόνα 2):



Εικόνα 2 - Δύο τροχιές Dubins, μαζί με την κατασκευαστική τους γεωμετρία

Παρότι αυτού του είδους οι τροχιές αποτελούν μια εύκολη λύση στο μεμονωμένο πρόβλημα αλλαγής κατάστασης στο όχημα από (x_0, y_0, θ_0) σε (x_1, y_1, θ_1) και εγγυώνται την εύρεση της συντομότερης πιθανής τροχιάς μεταξύ των δύο σημείων δεδομένων των περιορισμών, προκειμένου να αποφευχθούν πολλαπλά εμπόδια και να βρεθεί βέλτιστη διαδρομή μεταξύ δύο σημείων στην πραγματικότητα θα χρειάζονταν πολλαπλές εφαρμογές της μεθόδου με πολλά άγνωστα ενδιάμεσα σημεία.

Επίσης, για ένα μεμονωμένο πρόβλημα υπάρχουν γενικά 6 διαφορετικοί συνδυασμοί κυκλικών και ευθύγραμμων τμημάτων οι οποίοι ικανοποιούν το αρχικό και το τελικό σημείο, καθώς και τον προσανατολισμό, όπως παρουσιάζεται στην Εικόνα 3:



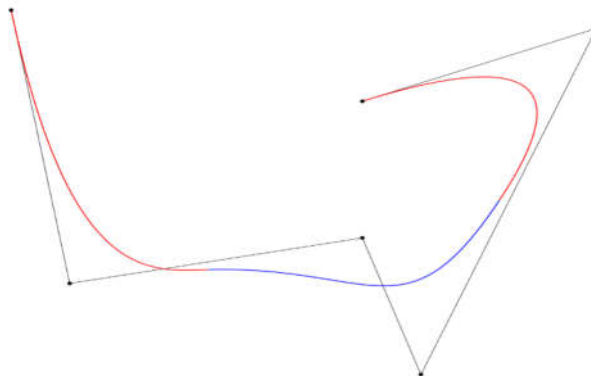
Εικόνα 3 - Οι 6 πιθανές τροχιές Dubins μεταξύ δύο σημείων

Στην Εικόνα 3 φαίνεται πως για κάθε μεμονωμένο πρόβλημα χρειάζεται να υπολογιστούν 6 τροχιές με τα αντίστοιχα 6 μήκη και τελικά να επιλεγεί το συντομότερο, κάτι το οποίο συνεισφέρει επίσης στον υπολογιστικό χρόνο που απαιτείται για την λύση του προβλήματος με αυτή τη μεθοδολογία εάν αναλογιστούμε ότι θα πρέπει και κάθε μια τροχιά να ελεγχθεί ως προς άλλες παραμέτρους (π.χ. αποφυγή εμποδίων). Ωστόσο, στην περίπτωση που μας ενδιαφέρει η συντομότερη δυνατή τροχιά (δεδομένων των περιορισμών χρήσης ευθύγραμμων και κυκλικών τμημάτων) η μέθοδος αυτή εγγυάται την εύρεση της βέλτιστης λύσης ως προς αυτά τα κριτήρια [16].

1.3.2 Μέθοδος με B-Splines

Οι καμπύλες Basis Splines (B-Splines) είναι μαθηματικές καμπύλες που μπορούν να αναπαραστήσουν την διαδρομή ενός οχήματος τόσο σε 2-D περιβάλλον όσο και σε 3 διαστάσεις. Κάθε τέτοια καμπύλη ορίζεται με βάση ένα σύνολο σημείων c_i τα οποία ονομάζονται σημεία ελέγχου (control points), καθώς και ένα σύνολο $t_i \in [0,1]$ που ονομάζεται διάνυσμα κόμβων (knot vector) και καθορίζει τα σημεία στα οποία τα επιμέρους τμήματα της καμπύλης θα τέμνονται, τα οποία ονομάζονται σημεία κόμβων. Γενικά, μια καμπύλη b-spline n – οστής τάξης είναι μια συνάρτηση που αποτελείται από ξεχωριστά πολυωνυμικά τμήματα βαθμού $n - 1$, και άρα εξ ορισμού είναι συνεχής και ομαλή τόσο στα ενδιάμεσα σημεία όσο και στα ακραία σημεία των τμημάτων.

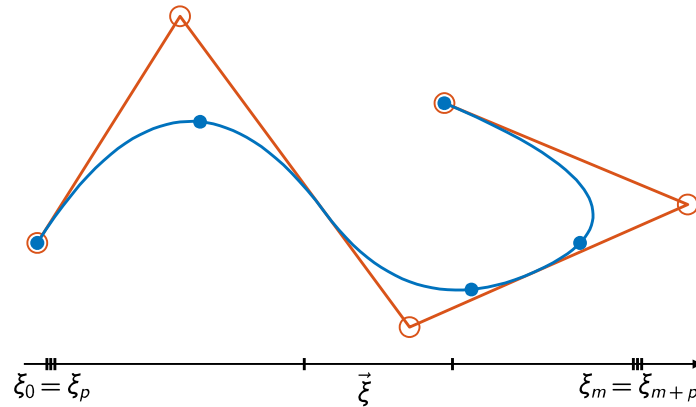
Μια τέτοια καμπύλη σε 2 διαστάσεις φαίνεται στην Εικόνα 4, όπου τονίζονται και τα σημεία ελέγχου καθώς και η τεθλασμένη γραμμή που τα ενώνει και ονομάζεται πολύγωνο ελέγχου:



Εικόνα 4 - Καμπύλη B Spline

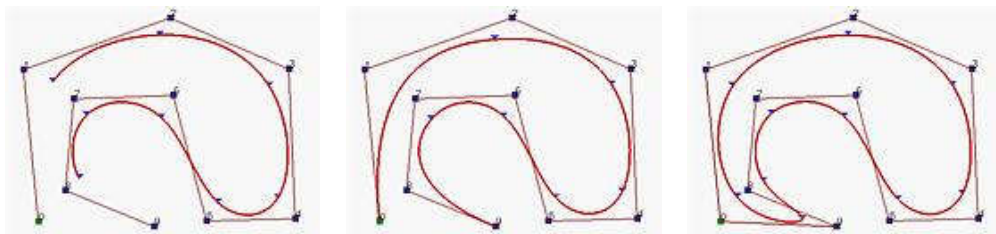
Με διαφορετικό χρώμα στην Εικόνα 4 φαίνονται τα τρία διαφορετικά πολυωνυμικά τμήματα από τα οποία αποτελείται η καμπύλη, ενώ στα σημεία που τέμνονται δύο διαφορετικά χρώματα βρίσκονται τα σημεία των κόμβων, όπως έχουν οριστεί από το σύνολο t_i .

Στην Εικόνα 5 τα σημεία των κόμβων τονίζονται επάνω στην καμπύλη:



Εικόνα 5 - Καμπύλη B Spline

Για δεδομένο σύνολο σημείων ελέγχου, ανάλογα με την επιλογή των σημείων των κόμβων προκύπτουν καμπύλες που είναι ανοικτές, κλειστές, ή καμπύλες που περνάνε από το πρώτο και το τελευταίο σημείο ελέγχου:



Εικόνα 6 - Καμπύλες b-spline

Στην Εικόνα 6 φαίνονται 3 διαφορετικές καμπύλες οι οποίες είναι όλες βαθμού $n = 3$ και δημιουργούνται από το ίδιο σύνολο 10 σημείων ελέγχου. Η διαφορά στην μορφή τους προέρχεται από την διαφορετική επιλογή συνόλου κόμβων η οποία στην πρώτη περίπτωση δημιουργεί μια ανοικτή καμπύλη, στην δεύτερη περίπτωση δημιουργεί πάλι μια ανοικτή καμπύλη η οποία όμως περνάει από το αρχικό και τελικό σημείο ελέγχου ενώ στην τελευταία περίπτωση δημιουργεί μια κλειστή καμπύλη.

Για να δημιουργηθεί μια καμπύλη η οποία περνάει από τα ακραία σημεία ελέγχου, η μόνη προϋπόθεση είναι στο σύνολο κόμβων ο πρώτος και ο τελευταίος κόμβος να εμφανίζονται $n + 1$ φορές, ενώ δεν έχει σημασία το υπόλοιπο του συνόλου. Για την μεσαία καμπύλη στην Εικόνα 4, έχει χρησιμοποιηθεί το εξής σύνολο κόμβων, όπου φαίνεται η πολλαπλότητα των $t_i = 0$ και $t_i = 1$:

$$t_i = \{0, 0, 0, 0, 0.14, 0.28, 0.42, 0.57, 0.71, 0.85, 1, 1, 1, 1\}$$

Η μαθηματική έκφραση μιας τέτοιας καμπύλης έχει την εξής μορφή (μορφή de Boor):

$$S(x) = \sum_i c_i B_{i,p}(x) \quad (2a)$$

Όπου c_i είναι τα σημεία ελέγχου της καμπύλης, $p = n - 1$ είναι ο βαθμός των πολυωνύμων από τα οποία αποτελείται η καμπύλη, και για τις συναρτήσεις $B_{i,p}(x)$ ισχύει ο αναδρομικός τύπος:

$$B_{i,0}(x) = \begin{cases} 1, & t_i \leq x < t_{i+1} \\ 0, & \text{σε άλλη περίπτωση} \end{cases} \quad (2b)$$

$$B_{i,p}(x) = \frac{x - t_i}{t_{i+p} - t_i} B_{i,p-1}(x) + \frac{t_{i+p+1} - x}{t_{i+p+1} - t_{i+1}} B_{i+1,p-1}(x) \quad (2c)$$

Στην περίπτωση της χρήσης γενετικών αλγορίθμων, μια τέτοιου είδους καμπύλη μπορεί να είναι ιδιαίτερα εύχρηστη καθώς τα σημεία ελέγχου (συγκεκριμένα, οι συντεταγμένες τους) μπορεί να αποτελέσουν γονίδια των χρωμοσωμάτων, και τελικά ο γενετικός αλγόριθμος να δημιουργεί ολόκληρες καμπύλες οι οποίες θα περιγράφουν μια τροχιά μεταβάλλοντας μόνο κάποια διακριτά σημεία. Στην καμπύλη που θα προκύψει μπορεί μετά να γίνει δειγματοληψία σε όσα σημεία και με όποια πυκνότητα χρειαστεί με βάση τις παραπάνω εξισώσεις (2), ενώ η γνώση κλειστών αλγεβρικών μορφών για τις εξισώσεις αυτές μπορεί να εξυπηρετήσει και στον υπολογισμό παραγώγων. Ο υπολογισμός των παραγώγων είναι χρήσιμος, καθώς κάνοντας χρήση παραγώγων είναι εφικτό να υπολογιστεί η επιτάχυνση που θα πρέπει να έχει ένα όχημα που θα εκτελέσει την τροχιά, και να συγκριθούν οι απαιτήσεις παρακολούθησης με τις δυνατότητες του οχήματος ως δείκτης αξιολόγησης του αποτελέσματος.

Οι καμπύλες B-Spline έχουν χρησιμοποιηθεί ήδη στην βιβλιογραφία σε συνδυασμό με γενετικούς αλγορίθμους, για την λύση του προβλήματος του σχεδιασμού τροχιάς τόσο σε offline [17] όσο και σε online [18] υλοποιήσεις. Ωστόσο, ένα μειονέκτημα της μεθόδου αυτής είναι ότι λόγω της συνεχόμενης αλλαγής στην καμπυλότητα και την απουσία τμημάτων με σταθερή γεωμετρία (πχ ευθύγραμμο ή κυκλικά τμήματα) θα πρέπει να γίνει ενδελεχής έλεγχος κάθε καμπύλης σε πολλαπλά σημεία προκειμένου να επιβεβαιωθούν οι απαιτούμενες προϋποθέσεις (πχ ότι δεν υπάρχουν συγκρούσεις με το περιβάλλον). Αυτή η διαδικασία αυξάνει τον υπολογιστικό χρόνο που θα απαιτείται για την επιβεβαίωση μιας λύσης και φυσικά είναι ανάλογη με το πλήθος των σημείων που θα ελεγχθούν και το συνολικό μήκος της τροχιάς.

Ένα επιπλέον μειονέκτημα μιας τέτοιας λύσης είναι ότι θα πρέπει είτε να είναι περιορισμένος και ορισμένος εκ των προτέρων ο αριθμός των σημείων ελέγχου (το οποίο θα οδηγήσει σε καμπύλη με λιγότερες λεπτομέρειες), είτε θα πρέπει να υποστηρίζεται από τον αλγόριθμο μεταβλητό πλήθος σημείων ελέγχου με κάποια αντίστοιχη στρατηγική για την επιλογή του αριθμού τους [19] [20].

1.3.3 Μέθοδος πεδίων δυναμικού

Η μέθοδος με την χρήση πεδίων δυναμικού στηρίζεται πάνω σε αρχές κινηματικής και μηχανικής, και είναι μια προσέγγιση εμπνευσμένη από την φύση. Το πρόβλημα της αναζήτησης τροχιάς μετατρέπεται σε ένα πρόβλημα κινηματικής, θεωρώντας ότι το όχημα για το οποίο θέλουμε να δημιουργήσουμε την τροχιά βρίσκεται μέσα σε ένα δυναμικό πεδίο το οποίο αφενός απωθεί το όχημα από τα εμπόδια ενώ αντίθετα το όχημα έλκεται από το τελικό επιθυμητό σημείο.

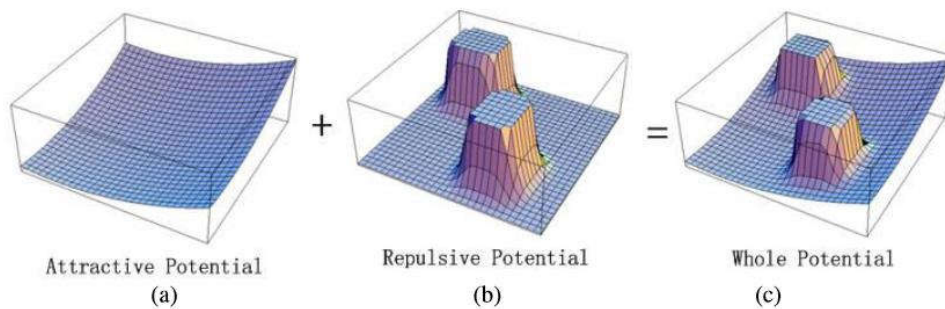
Γενικά, η ακριβής μορφή του δυναμικού ορίζεται από την εκάστοτε υλοποίηση και υπάρχουν πολλές διαφορετικές προσεγγίσεις για την δημιουργία του, αλλά σε κάθε περίπτωση μπορούμε να υποθέσουμε ότι υπάρχει ένα δυναμικό με την μορφή [21]

$$U(r) = U_{rep}(r) + U_{att}(r) \quad (3a)$$

όπου οι επιμέρους όροι U_{rep} και U_{att} αποτελούν αντίστοιχα το απωστικό και το ελκτικό τμήμα του πεδίου, οδηγώντας τελικά σε μια δύναμη που επιδρά πάνω στο όχημα και το οδηγεί προς τον στόχο με την εξής μορφή:

$$F(r) = -\nabla U(r) \quad (3b)$$

Ένα τέτοιο δυναμικό πεδίο, το οποίο περιέχει δύο εμπόδια φαίνεται στην επόμενη Εικόνα 7:



Εικόνα 4 - Απεικόνιση πεδίου

Έχοντας κατασκευάσει το πεδίο, από εκεί και μετά το πρόβλημα αντιμετωπίζεται ως πρόβλημα κινηματικής και η πορεία που ακολουθεί το όχημα υπό την επήρεια του δυναμικού αποτελεί την λύση στο πρόβλημα του σχεδιασμού τροχιάς. Ένα συνηθισμένο φαινόμενο που εμφανίζεται ειδικά όταν το περιβάλλον αποκτά σύνθετη μορφή (πχ με πολλά εμπόδια) είναι η εμφάνιση τοπικών ελαχίστων, από τα οποία η μεθοδολογία ίσως και να μην μπορεί να ξεφύγει. Στην βιβλιογραφία υπάρχουν κάποιες λύσεις σε αυτό το πρόβλημα, όπως για παράδειγμα η χρήση αρμονικών συναρτήσεων για την αναπαράσταση του δυναμικού οι οποίες δεν παρουσιάζουν εκ κατασκευής τοπικά ελάχιστα [22] ή χρησιμοποιώντας τεχνικές που εισάγουν μια φραγμένη διαταραχή στο δυναμικό ή το όχημα που εγγυάται υπό προϋποθέσεις την αποφυγή τέτοιων ελαχίστων [23].

Η λύση του προβλήματος σχεδιασμού τροχιάς με την μέθοδο αυτή, για δεδομένο αρχικό σημείο και σταθερό δυναμικό πεδίο, είναι καθαρά αιτιοκρατική. Επομένως, για την εύρεση μιας βέλτιστης λύσης είναι αναγκαίος ο συνδυασμός της μεθόδου αυτής με κάποιο εργαλείο βελτιστοποίησης όπως για παράδειγμα εξελικτικοί αλγόριθμοι [24] ή μέθοδοι που εμπίπτουν στην κατηγορία βελτιστοποίησης σμήνους σωματιδίων (particle swarm optimization – PSO) [25].

Κεφάλαιο 2 - Προαπαιτούμενα

2.1 Εισαγωγή

Σε αυτό το κεφάλαιο θα παρατεθούν έννοιες και εξειδικευμένες τεχνικές οι οποίες απαιτούνται για την λειτουργία των αλγορίθμων που θα χρησιμοποιηθούν. Θα αναφερθούν ζητήματα όπως η αναπαράσταση ενός εδαφικού ανάγλυφου στον αλγόριθμο, η αναπαράσταση της τροχιάς, καθώς και διάφορες μέθοδοι αντιμετώπισης της διακριτής φύσης των δεδομένων σε σχέση με ένα πρόβλημα στον συνεχή πραγματικό χώρο.

2.2 Αναπαράσταση ανάγλυφου

2.2.1 Δεδομένα

Για την λειτουργία ενός αλγορίθμου σχεδιασμού τροχιάς, είναι απαραίτητη η αναπαράσταση ενός χάρτη του περιβάλλοντος (είτε εικονικού, είτε πραγματικού). Στην πραγματικότητα, οι χάρτες ανάλογα με την χρήση για την οποία προορίζονται δημιουργούνται με διαφορετικούς τρόπους απεικόνισης πληροφορίας όπως πχ με ισοδυναμικές γραμμές για όποια ιδιότητα μας ενδιαφέρει ή κάνοντας χρήση χρώματος ως 3^η διάσταση. Σε κάθε περίπτωση, αυτό που έχει σημασία είναι τα δεδομένα από τα οποία δημιουργείται ο χάρτης, και όχι ο τρόπος απεικόνισης.

Στην περίπτωση μας χρειαζόμαστε μια αναπαράσταση του ανάγλυφου της επιφάνειας, τουλάχιστον από την επιφάνεια της θάλασσας και πάνω ($z > 0$), την οποία μπορούμε να έχουμε από έναν τοπογραφικό χάρτη. Πιο συγκεκριμένα, χρειάζονται δεδομένα της μορφής (x, y, z) , όπου z θα είναι το υψόμετρο του σημείου (x, y) από την επιφάνεια της θάλασσας (altitude ASL), ενώ για τις συντεταγμένες (x, y) οι πιο λογικές επιλογές είναι το γεωγραφικό μήκος και πλάτος.

Τέτοιου είδους σετ δεδομένων είναι διαθέσιμα από τοπογραφικές υπηρεσίες, όπως για παράδειγμα η Γεωγραφική Υπηρεσία Στρατού της Ελλάδας ή ακόμη και η τοπογραφική υπηρεσία των ΗΠΑ (USGS – United States Geological Survey), φορείς οι οποίοι παρέχουν

τοπογραφικά δεδομένα με ποικίλα επίπεδα ανάλυσης/λεπτομέρειας, και συνήθως προσφέρονται στην μορφή DTED – Digital Terrain Elevation Data.

Το πρόβλημα ωστόσο με αυτής της μορφής τα δεδομένα είναι πως είναι διακριτά, όπως, για παράδειγμα, παρουσιάζεται στον Πίνακα 1.

| # | x | y | z |
|---|------|------|------|
| 1 | 0.00 | 0.00 | 0.00 |
| 2 | 1.00 | 0.00 | 0.00 |
| 3 | 0.00 | 1.00 | 0.00 |
| 4 | 1.00 | 1.00 | 1.00 |

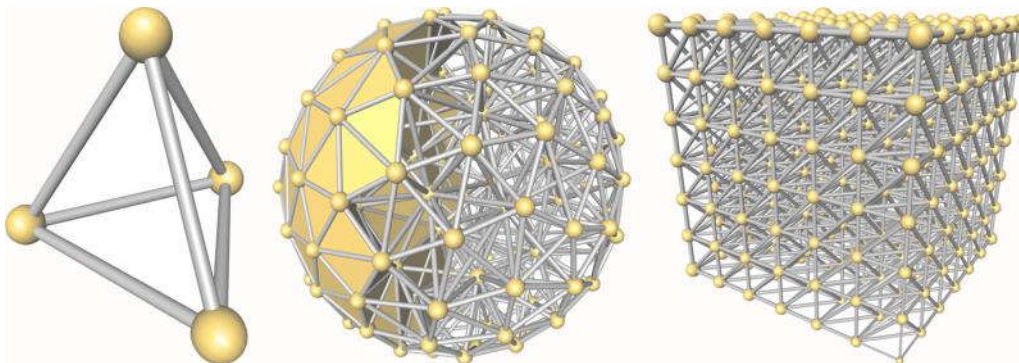
Πίνακας 1 – Εικονικό σκετ τοπογραφικών δεδομένων

Ο Πίνακας 1 απεικονίζει τις μετρήσεις για ένα τυχαίο ανάγλυφο οι οποίες έχουν πραγματοποιηθεί στα σημεία (0,0), (0,1), (1, 0) και (1,1). Από τα δεδομένα αυτά μπορούμε πολύ εύκολα να απαντήσουμε τι γίνεται στα σημεία (0,0) ή (0,1) για παράδειγμα, αλλά εφόσον μια επιφάνεια δεν μπορεί να είναι διακριτή και φυσικά δεν είναι εφικτό να υπάρχουν μετρήσεις σε άπειρα σημεία, θα πρέπει με κάποιον τρόπο να μπορέσουμε να εξάγουμε συμπεράσματα για το υψόμετρο σημείων στα οποία δεν έχει πραγματοποιηθεί μέτρηση, όπως για παράδειγμα το σημείο (0.5, 0.5).

2.2.2 Τριγωνοποίηση

Ένας πιθανός τρόπος να λυθεί το πρόβλημα της έλλειψης δεδομένων για ένα τυχαίο σημείο (x, y) είναι με χρήση παρεμβολής (interpolation). Για να μπορέσουμε να κάνουμε χρήση όμως της διαδικασίας της παρεμβολής στα δεδομένα, θα πρέπει να μετατραπεί το σύνολο σημείων που έχουμε σε ένα σύνολο επιφανειών δημιουργώντας μια τριγωνοποίηση (triangulation) της επιφάνειας με βάση τα σημεία για τα οποία έχουμε διαθέσιμα δεδομένα. Η τριγωνοποίηση είναι μια ευρέως διαδεδομένη διαδικασία αλλά δεν αποτελεί τετριμμένο πρόβλημα, καθώς υπάρχουν αρκετές μέθοδοι & διαφορετικοί αλγόριθμοι υλοποίησης της.

Παρόλο που το πρόβλημα που αντιμετωπίζεται είναι τρισδιάστατο⁵ δεν χρειαζόμαστε μια 3-D τριγωνοποίηση όπως αυτές που φαίνονται στην παρακάτω Εικόνα 8, καθώς δε μας ενδιαφέρει σε αυτή την περίπτωση το εσωτερικού του περικλειόμενου από την επιφάνεια όγκου (το οποίο και τριγωνοποιείται στην 3-D περίπτωση).



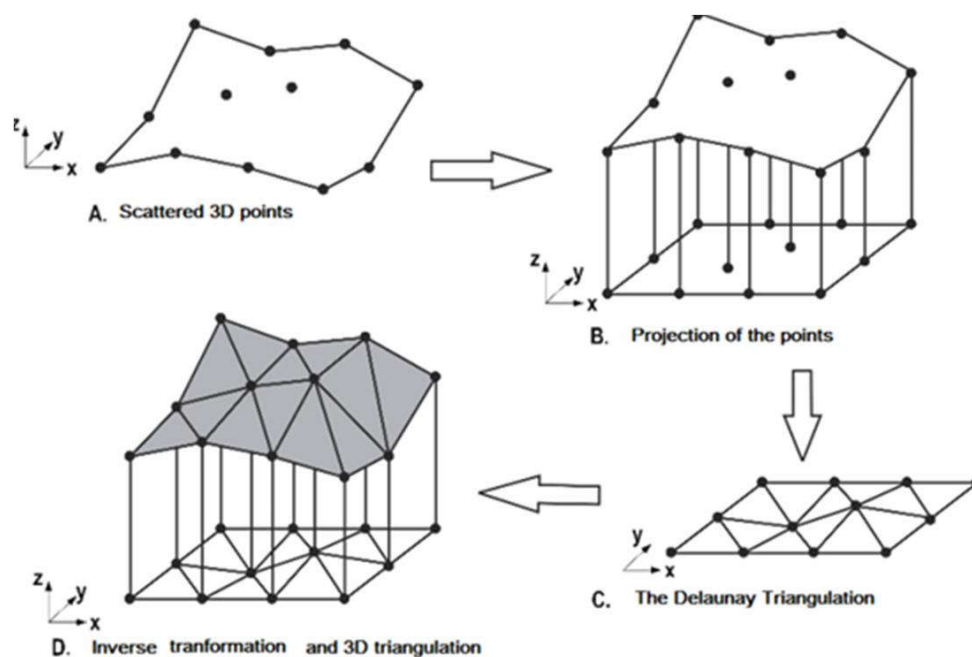
Εικόνα 5 - Τριγωνοποίηση σε 3 διαστάσεις

Αντίθετα, υπενθυμίζεται ότι το πρόβλημα είναι πως έχουμε πεπερασμένο αριθμό δεδομένων (datapoints) τα οποία όμως αφορούν αποκλειστικά και μόνο την επιφάνεια, παρότι αυτή είναι τρισδιάστατη. Ένας πιθανός αλγόριθμος για να πετύχουμε το επιθυμητό αποτέλεσμα είναι ο εξής (Εικόνα 9):

- Προβολή του συνόλου των σημείων (x, y, z) στο επίπεδο xy
- Δημιουργία μιας τριγωνοποίησης από το νέο σύνολο των 2-D σημείων
- Αναστροφή της προβολής, επαναφέροντας κάθε σημείο στο αρχικό ύψος z
- Δημιουργία μιας τριγωνοποίησης με βάση τα αρχικά (3-D) σημεία και τα τρίγωνα αυτών όπως δημιουργήθηκαν από την τριγωνοποίηση των αντίστοιχων σημείων (x, y)

⁵ Για την ακρίβεια, η επιφάνεια είναι δύο διαστάσεων αλλά βρίσκεται ενσωματωμένη σε έναν τρισδιάστατο χώρο

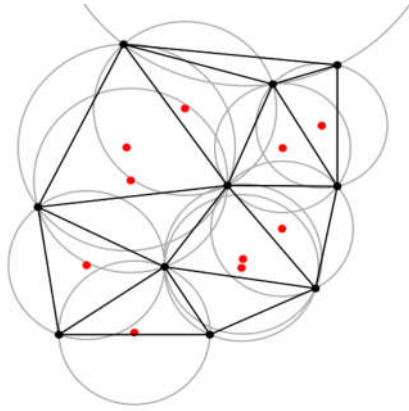
Ένας μεγάλος περιορισμός αυτής της μεθοδολογίας είναι η αδυναμία χρήσης της σε περιπτώσεις όπου η επιφάνεια είναι κοίλη σε σχέση με το σημείο το οποίο βρισκόμαστε, και υπάρχουν τρίγωνα στο ίδιο (x, y) αλλά σε διαφορετικά ύψη z . Θεωρώντας ότι στην περίπτωση που πραγματεύεται αυτή η εργασία δεν θα ισχύει κάτι τέτοιο και ότι οι χάρτες δεν θα περιγράφουν κοίλες επιφάνειες, μπορούμε να εκμεταλλευτούμε την παραπάνω μέθοδο λόγω απλότητας και ευκολίας στην υλοποίηση.



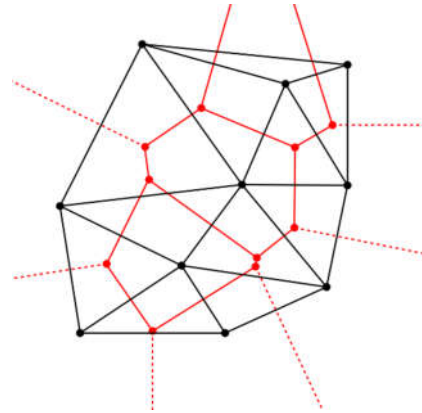
Εικόνα 6 - Τριγωνοποίηση Delaunay 2D επιφάνειας σε 3 διαστάσεις

Για την δημιουργία της τριγωνοποίησης (2-D) χρησιμοποιείται ο αλγόριθμος Delaunay [26] ο οποίος παράγει την ομώνυμη τριγωνοποίηση. Η τριγωνοποίηση αυτή σχετίζεται με το διάγραμμα Voronoi ενός συνόλου σημείων (για την ακρίβεια είναι δυϊκή οντότητα του), και οι βασικές της ιδιότητες στην περίπτωση των δύο διαστάσεων είναι οι εξής:

- Μεγιστοποίηση των ελάχιστων γωνιών που σχηματίζονται από τις άκρες τριγωνισμού
- Ελαχιστοποίηση των μέγιστων περιγεγραμμένων ακτινών (των κύκλων που σχετίζονται με τα στοιχεία τριγωνισμού)



Εικόνα 7 - Τριγωνοποίηση και περιγεγραμμένοι κύκλοι



Εικόνα 8 - Τριγωνοποίηση και διάγραμμα Voronoi

2.2.3 Παρεμβολή

Έχοντας διαθέσιμη πλέον μια τριγωνοποίηση της επιφάνειας του ανάγλυφου, μπορούμε να την εκμεταλλευτούμε προκειμένου να υπολογίσουμε το ύψος της επιφάνειας σε όποιο σημείο χρειάζεται, δηλαδή προκύπτει ότι $z(x, y) = f(x, y)$ οπότε και δεν υπάρχει πλέον το πρόβλημα ότι το ύψος z είναι γνωστό μόνο στα σημεία (x_i, y_i) για τα οποία είναι διαθέσιμη η πληροφορία από άμεση μέτρηση. Για να προχωρήσουμε στην αναζήτηση των συντεταγμένων ενός τυχαίου σημείου στο τρίγωνο με παλινδρόμηση, θα κάνουμε χρήση του συστήματος βαρυκεντρικών συντεταγμένων του τριγώνου αυτού⁶.

Για κάθε τρίγωνο $AB\Gamma$ ισχύει ότι για όλα τα εσωτερικά του σημεία P υπάρχει μια μοναδική αναπαράσταση τέτοια ώστε:

$$\overline{OP} = \lambda_1 \overline{OA} + \lambda_2 \overline{OB} + \lambda_3 \overline{O\Gamma} \quad (4a)$$

$$\lambda_1 + \lambda_2 + \lambda_3 = 1 \quad (4b)$$

$$\lambda_1, \lambda_2, \lambda_3 \geq 0 \quad (4c)$$

⁶ Υπενθυμίζεται ότι το κέντρο βάρους ή βαρύκεντρο ενός τριγώνου είναι το σημείο τομής των διχοτόμων των πλευρών. Το σημείο αυτό K βρίσκεται από το $\overline{OK} = \frac{1}{3}(\overline{OA} + \overline{OB} + \overline{O\Gamma})$

Υποθέτοντας ένα τρίγωνο με κορυφές τα σημεία $A(x_A, y_A, z_A)$, $B(x_B, y_B, z_B)$, και το σημείο $\Gamma(x_\Gamma, y_\Gamma, z_\Gamma)$ τότε για ένα τυχαίο σημείο $P(x_p, y_p, z_p)$ του τριγώνου θα ισχύει, με βάση τα παραπάνω, ότι:

$$\begin{pmatrix} x_p \\ y_p \\ z_p \end{pmatrix} = \begin{bmatrix} x_A & x_B & x_\Gamma \\ y_A & y_B & y_\Gamma \\ z_A & z_B & z_\Gamma \end{bmatrix} \begin{pmatrix} \lambda_1 \\ \lambda_2 \\ \lambda_3 \end{pmatrix}$$

$$\begin{aligned} x_p &= \lambda_1 x_A + \lambda_2 x_B + \lambda_3 x_\Gamma \\ \Leftrightarrow y_p &= \lambda_1 y_A + \lambda_2 y_B + \lambda_3 y_\Gamma \\ z_p &= \lambda_1 z_A + \lambda_2 z_B + \lambda_3 z_\Gamma \end{aligned}$$

$$\Leftrightarrow P = T\lambda$$

Δεδομένου όμως ότι $\lambda_1 + \lambda_2 + \lambda_3 = 0$ από την (4b), τότε το σύστημα μπορεί να απλοποιηθεί σε ένα με δύο εξισώσεις αντί για τρείς, αν θέσουμε $\lambda_3 = 1 - \lambda_1 - \lambda_2$:

$$\begin{aligned} \lambda_1(x_A - x_\Gamma) + \lambda_2(x_B - x_\Gamma) + x_\Gamma - x_p &= 0 \\ \lambda_1(y_A - y_\Gamma) + \lambda_2(y_B - y_\Gamma) + y_\Gamma - y_p &= 0 \end{aligned}$$

Και ισοδύναμα μπορεί να γραφεί με μορφή πινάκων:

$$\begin{bmatrix} x_A - x_\Gamma & x_B - x_\Gamma \\ y_A - y_\Gamma & y_B - y_\Gamma \end{bmatrix} \begin{pmatrix} \lambda_1 \\ \lambda_2 \end{pmatrix} - \begin{pmatrix} x_p - x_\Gamma \\ y_p - y_\Gamma \end{pmatrix} = \mathbf{0}$$

$$\Leftrightarrow T\tilde{\lambda} = u$$

Το σύστημα αυτό είναι επιλύσιμο με μοναδική λύση $\tilde{\lambda} = T^{-1}u$, καθώς ο πίνακας T είναι αντιστρέψιμος. Η αντιστρεψιμότητα του πίνακα T προκύπτει από το γεγονός ότι οι δύο στήλες του αναπαριστούν τις προβολές των διανυσμάτων $(\overline{OA} - \overline{O\Gamma}) = \overline{\Gamma A}$ και

$(\overline{OB} - \overline{OA}) = \overline{AB}$ στο επίπεδο xy και επομένως είναι γραμμικά ανεξάρτητα διανύσματα, καθώς εάν ήταν συνευθειακά τότε δεν θα υπήρχε το τρίγωνο $ABΓ$ εξ'αρχής.

Γνωρίζοντας πλέον το $\tilde{\lambda} = (\lambda_1, \lambda_2)^T$ μπορούμε να υπολογίσουμε τον τελευταίο συντελεστή από τη σχέση (4b) και επομένως είναι πλέον γνωστό το διάνυσμα $\lambda = (\lambda_1, \lambda_2, \lambda_3)^T$ και κατ' επέκταση το ύψος z_p του σημείου P .

Η παραπάνω λύση προφανώς εκτιμάει το ύψος ενός σημείου θεωρώντας πως αυτό ανήκει στην επιφάνεια του τριγώνου, επομένως είναι τόσο ορθή όσο και η αντιμετώπιση της επιφάνειας του ανάγλυφου ως αποτελούμενη από τρίγωνα. Η ποιότητα και η πιστότητα της παρεμβολής αυτής εξαρτάται από την χωρική πυκνότητα των αρχικών δεδομένων, καθώς εάν η αρχική καταγραφή δεν έχει αρκετά υψηλή ανάλυση και κατ' επέκταση χάνεται πληροφορία για το ανάγλυφο τότε και η παρεμβολή αντίστοιχα θα κρύψει χαρακτηριστικά της επιφάνειας. Δηλαδή, χαμηλότερη ακρίβεια στην χαρτογράφηση θα οδηγεί αντίστοιχα σε χαμηλότερη ακρίβεια κατά την παρεμβολή για τον υπολογισμό του ύψους των ενδιαμέσων σημείων.

Με βάση το θεώρημα των Shannon-Nyquist σχετικά με τη δειγματοληψία ενός σήματος του οποίου μια αποτύπωση είναι «*Η ελάχιστη συχνότητα δειγματοληψίας ενός σήματος, η οποία δεν εισάγει αλλοίωση στην υπάρχουσα σε αυτό πληροφορία, είναι διπλάσια της συχνότητας της πλέον υψίσυχνης συνιστώσας του*», μπορούμε μεταφέροντας το στον χώρο της γεωδαισίας να δεχθούμε ως επαρκή χωρική πυκνότητα δειγματοληψίας μια τιμή ρ η οποία θα ικανοποιεί την επόμενη σχέση, όπου $\Phi(x, y)$ είναι μια συνάρτηση που αναπαριστά το υψόμετρο.

$$\frac{1}{\rho} \geq \frac{1}{2} |\nabla \Phi| \quad (5)$$

2.3 Μετατροπή συστημάτων συντεταγμένων

Όπως είδαμε ήδη, για την εκτέλεση της τριγωνοποίησης καθώς και της διαδικασίας της παρεμβολής για τον υπολογισμό του ύψους ενός σημείου, δεχθήκαμε ότι οι χαρτογραφικές συντεταγμένες είναι καρτεσιανές. Στην πραγματικότητα όμως, οι χαρτογράφοι συνήθως χρησιμοποιούν γεωδαιτικές συντεταγμένες και συγκεκριμένα τα δεδομένα τύπου DTED είναι σε μορφή τριπλέτων γεωγραφικού μήκους, πλάτους και

ύψους, με το τελευταίο να μετράται με βάση το γεωειδές EGM96 και όχι το συνηθέστερο ελλειψοειδές αναφοράς WGS84⁷.

Σε κάθε περίπτωση, οι δύο επιλογές που υπάρχουν για τους αλγορίθμους είναι να χρησιμοποιηθούν οι συντεταγμένες ως έχουν, ή να μετατραπούν σε καρτεσιανές. Δεδομένου ότι η χρήση των πρώτων θα ήταν ιδιαίτερα πολύπλοκη σε θέματα υπολογισμών, στην παρούσα μελέτη έχει επιλεγεί η μετατροπή των συντεταγμένων σε καρτεσιανές προκειμένου να είναι πιο εύκολη η γεωμετρία και οι πράξεις στα βήματα των αλγορίθμων. Γνωρίζοντας πάντα ότι δεν είναι δυνατόν να προβληθεί μια σφαιρική επιφάνεια σε δύο διαστάσεις χωρίς αυτή να παραμορφωθεί καθώς δεν είναι δυνατόν να υπάρχει ομοιομορφισμός από την σφαίρα στο επίπεδο λόγω διαφορετικής συμπίεσης, καθώς και επίσης κρατώντας υπόψη ότι τα δεδομένα DTED δεν είναι σφαιρικές συντεταγμένες αλλά τύπου (lat, lon, z_h) μπορούμε να κάνουμε μια μεταφορά σε καρτεσιανές συντεταγμένες σκεπτόμενοι ως εξής:

$$x = R_e \sin(lon) \quad (6a)$$

$$y = R_e \sin(lat) \quad (6b)$$

$$z = z_h \quad (6c)$$

Με αυτόν τον μετασχηματισμό, αποκτούμε μια αναπαράσταση εν είδη προβολής των δεδομένων, στην οποία η διεύθυνση y αντιστοιχεί στην διεύθυνση Βορράς - Νότος, και η διεύθυνση x αντιστοιχεί στην διεύθυνση Ανατολή – Δύση. Για την μετατροπή μεταξύ DTED και αυτών των συντεταγμένων δεχόμαστε ότι τα ανατολικά γεωγραφικά μήκη θα είναι θετικά, και ομοίως και για τα βόρεια γεωγραφικά πλάτη. Η αυθαίρετη φορά της θετικότητας δεν επηρεάζει σε κάτι τους αλγόριθμους, και στο τέλος οι βοηθητικές αυτές καρτεσιανές συντεταγμένες θα επαναφερθούν σε γεωγραφικές, με τους αντίστοιχους αντίστροφους μετασχηματισμούς. Τέλος, οι μονάδες μέτρησης των αξόνων x, y θα είναι μονάδες μήκους ενώ το σημείο $(x, y, z_0) = (0, 0, z_0)$ θα αντιστοιχεί στο σημείο τομής του ισημερινού με τον 1^ο μεσημβρινό.

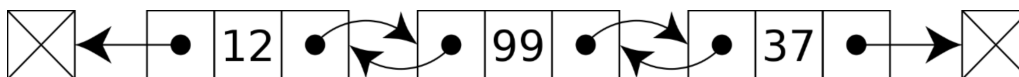
2.4 Αναπαράσταση τροχιάς

Στην παρούσα μελέτη η τροχιά ενός οχήματος αναπαρίσταται από μια τεθλασμένη γραμμή, δηλαδή από μια αλυσίδα διαδοχικών ευθύγραμμων τμημάτων. Τα άκρα των

⁷ U.S. Military Specification Digital Terrain Elevation Data (DTED) MIL-PRF-89020B

ευθύγραμμων αυτών τμημάτων μπορούν να λειτουργήσουν για το όχημα και ως waypoints, αφήνοντας τελικά τον ελεγκτή που έχει είναι υπεύθυνος για την πλοήγηση του οχήματος να αποφασίσει για την βέλτιστη διαδρομή μεταξύ αυτών. Ωστόσο, όταν η πυκνότητα αυτών των κορυφών είναι σχετικά μεγάλη, τότε ουσιαστικά μπορούμε να κατευθύνουμε τον ελεγκτή ώστε να ακολουθήσει την επιθυμητή τροχιά, αφήνοντας του πάντα ένα μικρό περιθώριο στον έλεγχο του οχήματος.

Η αλυσίδα αυτή στον αλγόριθμο καταλήγει να αναπαρίσταται μέσω μιας δομής δεδομένων που ονομάζεται συνδεδεμένη λίστα, και ειδικότερα στην προκειμένη περίπτωση χρησιμοποιείται η διπλά συνδεδεμένη λίστα. Ως δομή δεδομένων, η διπλά συνδεδεμένη λίστα αποτελείται από ένα σύνολο στοιχείων, όπου το κάθε ένα στοιχείο - πέρα από τις όποιες άλλες ιδιότητες εμπεριέχει - κρατάει δύο ενδείξεις οι οποίες αφορούν το ποια είναι τα γειτονικά του στοιχεία. Έτσι, το πρώτο άκρο της διπλά συνδεδεμένης λίστας είναι ένα στοιχείο που δεν έχει προηγούμενο γείτονα (αλλά έχει επόμενο), τα ενδιάμεσα στοιχεία έχουν και τους δύο γείτονες, και τέλος το δεύτερο άκρο της είναι ένα στοιχείο το οποίο έχει προηγούμενο γείτονα αλλά όχι επόμενο.



Εικόνα 9 – Αναπαράσταση διπλά συνδεδεμένης λίστας

Όπως φαίνεται και στο παραπάνω παράδειγμα της Εικόνας 12, τα στοιχεία με περιεχόμενο «12» και «37» είναι τα άκρα της λίστας (καθώς έχουν μόνο από έναν γείτονα) ενώ το στοιχείο με περιεχόμενο «99» είναι εσωτερικό σημείο. Η διπλά συνδεδεμένη λίστα επιτρέπει με εύκολο τρόπο την προσθαφαίρεση στοιχείων, τόσο στο εσωτερικό της όσο και στα άκρα της.

Για την αφαίρεση ενός στοιχείου, αρκεί να αλλαχθούν οι αναφορές των γειτόνων του ως εξής (υποθέτοντας ότι στην αλυσίδα A-B-Γ θέλουμε να προσθέσουμε το Δ μεταξύ των σημείων A και B):

1. Ως επόμενος γείτονας του A ορίζεται το Δ
2. Ως προηγούμενος γείτονας το B ορίζεται το Δ

3. Ως προηγούμενος γείτονας του Δ ορίζεται ο παλιός προηγούμενος γείτονας του επόμενου γείτονα του Δ (σημείο Β), δηλαδή το Α
4. Ως επόμενος γείτονας του Δ ορίζεται ο παλιός επόμενος γείτονας του του προηγούμενου γείτονα του Δ (σημείο Α), δηλαδή το Β

Όπως φαίνεται από τα παραπάνω, με τέσσερις (λιγότερες αν πρόκειται για ακραίο σημείο) κινήσεις/πράξεις έχουμε επιτυχώς αλλάξει την μορφή της λίστας χωρίς να χρειαστεί να αντιγραφεί ολόκληρη σε κάποια νέα θέση στην μνήμη του υπολογιστή και να αναπαρασταθεί εκ νέου.

Αντίστοιχα εύκολη και απλή διαδικασία είναι η αφαίρεση ενός σημείου από την λίστα, τόσο σε ενδιάμεσα στοιχεία όσο και σε ακραία στοιχεία (υποθέτοντας ότι από την λίστα Α-Β-Γ-Δ θέλουμε να αφαιρέσουμε το Γ):

1. Ο επόμενος γείτονας του σημείου Γ (δηλαδή το σημείο Δ) ορίζεται ως ο επόμενος γείτονας του προηγούμενου γείτονα του Γ, δηλαδή του σημείου Β.
2. Ο προηγούμενος γείτονας του σημείου Γ (δηλαδή το σημείο Β) ορίζεται ως ο προηγούμενος γείτονας του επόμενου γείτονα του Γ, δηλαδή του σημείου Δ.

Και πάλι με έναν εύκολο τρόπο είναι δυνατή η αφαίρεση ενός σημείου από την λίστα, επομένως για μια λίστα με δυναμικό πλήθος στοιχείων όπου η προσθαφαίρεση σημείων θα είναι συχνό φαινόμενο, από υπολογιστικής άποψης συμφέρει η υλοποίηση της στα πλαίσια ενός αλγορίθμου με την δομή της διπλά συνδεδεμένης λίστας.

Γενικά οι όροι «προηγούμενος» και «επόμενος» γείτονας δεν έχουν κάποια ιδιαίτερη φυσική έννοια στις διπλά συνδεδεμένες λίστες, καθώς αυτό που μας ενδιαφέρει κατά κύριο λόγο είναι η σειρά διαδοχής των σημείων και όχι ο προσανατολισμός ή η φορά κατά την οποία διανύεται η λίστα. Ωστόσο, στην περίπτωση αυτή καθώς θέλουμε να αναπαραστήσουμε μια τροχιά αεροσκάφους με αρχή και τέλος μπορούμε να υιοθετήσουμε τους όρους αυτούς καθώς υπάρχει μια λογική συνέχεια και φορά διάνυσης των σημείων.

2.5 Αριθμητική επίλυση Διαφορικών Εξισώσεων (ΔΕ)

Για την επίλυση διαφορικών εξισώσεων με αριθμητικό τρόπο, μπορεί να χρησιμοποιηθεί μια απλή μέθοδος που ονομάζεται μέθοδος Euler. Η μέθοδος αυτή προκύπτει από τον ορισμό της παραγώγου μιας συνάρτησης, δηλαδή

$$\frac{df}{dt} = f'(t) = \lim_{t \rightarrow t_0} \frac{f(t_0) - f(t)}{t_0 - t} \quad (7a)$$

Ο οποίος με μια κατάλληλη αλλαγή μεταβλητής μπορεί να γραφεί και ως εξής

$$\frac{df}{dt} = f'(t) = \lim_{dt \rightarrow 0} \frac{f(t_0 + dt) - f(t_0)}{dt} \quad (7b)$$

Επομένως, μία διαφορική εξίσωση της μορφής $\frac{df}{dt} = g(t)$ μπορεί να γραφεί ως

$$\lim_{dt \rightarrow 0} \frac{f(t + dt) - f(t)}{dt} = g(t) \leftrightarrow f(t + dt) = f(t) + g(t)dt \quad (7c)$$

Ξεκινώντας λοιπόν με μια γνωστή αρχική συνθήκη, $f(t = t_0)$ και υπολογίζοντας την αντίστοιχη $g(t_0)$, μπορούμε να υπολογίσουμε την εξέλιξη της f και αντίστοιχα να συνεχίσουμε με τον ίδιο τρόπο για όσα βήματα θέλουμε. Η μέθοδος Euler είναι ο απλούστερος τρόπος αριθμητικής επίλυσης διαφορικών εξισώσεων, και έχει χρησιμοποιηθεί χάριν ευκολίας. Για καλύτερη σταθερότητα στους υπολογισμούς, έχει νόημα να ερευνηθεί η υλοποίηση διαφορετικών solver, οι οποίοι ενδεχομένως να κάνουν χρήση πιο σύνθετων μεθόδων όπως αυτών της οικογένειας Runge-Kutta (πχ RK4).

Στις παραπάνω εξισώσεις (7) εμφανίζεται ο όρος dt , ο οποίος αντιστοιχεί στο timestep της επίλυσης. Με βάση την θεωρία, οι παραπάνω τύποι ισχύουν όταν $dt \rightarrow 0$ και επομένως θεωρητικά μιλώντας θα έπρεπε η επιλογή του να είναι κάποιος πολύ μικρός αριθμός. Στην πραγματικότητα, ο αριθμός των επαναλήψεων που θα χρειάζονταν προκειμένου να επιλυθεί μια διαφορική εξίσωση σε συγκεκριμένο εύρος είναι αντιστρόφως ανάλογος του dt και επομένως πρέπει να επέλθει ένας συμβιβασμός μεταξύ υπολογιστικού χρόνου και πιστότητας των αποτελεσμάτων.

Για παράδειγμα, ας υποθέσουμε ότι θέλουμε να λύσουμε υπολογιστικά την εξίσωση $\frac{df}{dt} = \cos t$ στο διάστημα $t \in [0,1]$, με συνοριακή συνθήκη την $f(0) = 0$. Ο αναδρομικός τύπος που θα χρησιμοποιηθεί σε κάθε περίπτωση είναι ο εξής:

$$f(t + dt) = f(t) + t * dt$$

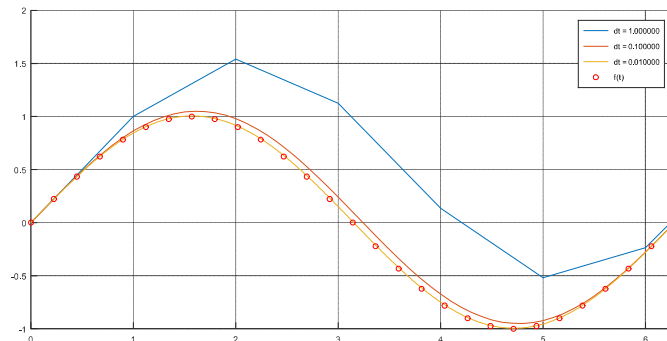
Επιλέγοντας για το dt τις τιμές $dt \in \{0.001, 0.01, 0.1\}$, υπολογίζουμε κατ' αρχήν τον αριθμό των επαναλήψεων N που θα χρειαστούν για κάθε περίπτωση προκειμένου να βρεθεί η λύση στο διάστημα που θέλουμε:

| dt | N |
|-------|--------------------------|
| 0.001 | $\frac{1}{0.001} = 1000$ |
| 0.01 | $\frac{1}{0.01} = 100$ |
| 0.1 | $\frac{1}{0.1} = 10$ |

Πίνακας 2 - Επαναλήψεις (N) σε σχέση με το θήμα dt

Επομένως, εύκολα φαίνεται ότι ανάλογα την επιλογή του χρόνου dt επηρεάζεται άμεσα ο αριθμός των επαναλήψεων που απαιτούνται, και υποθέτοντας πως ο χρόνος εκτέλεσης της κάθε επανάληψης είναι ανεξάρτητος του dt και σταθερός, αντίστοιχα προκύπτει ότι για μικρότερο dt θα χρειαστεί περισσότερος υπολογιστικός χρόνος προκειμένου να επιλυθεί το ίδιο πρόβλημα.

Ωστόσο, από την επιλογή του χρόνου dt εξαρτάται και η ακρίβεια της λύσης καθώς η προσέγγιση αυτή γραμμικοποιεί την συνάρτηση $f(t)$ στην περιοχή γύρω από το t και επομένως όσο πιο μακριά βρεθούμε (μεγάλο dt) η προσέγγιση δεν θα δώσει ικανοποιητικά αποτελέσματα, όπως αποτυπώνεται στην Εικόνα 13:



Εικόνα 10 - Επίλυση με διαφορετικά βήματα dt

Άρα τελικά για ένα σταθερό dt η επιλογή καταλήγει να είναι ένας συμβιβασμός μεταξύ υπολογιστικού χρόνου και ακρίβειας επίλυσης. Μια πιθανή λύση το πρόβλημα είναι η χρήση μεταβλητού βήματος το οποίο θα προσαρμόζεται στην κατά περίπτωση τοπική τιμή της παραγώγου συνάρτησης $g(t)$, καθώς εάν $g(t) \rightarrow 0$ τότε μπορούμε να αυξήσουμε το βήμα dt χωρίς αυτό να επιφέρει μεγάλη απόκλιση κατά την επίλυση και αντίστοιχα όταν αυξάνεται η τιμή της $g(t)$ μπορούμε να χρησιμοποιούμε μικρότερες τιμές για το dt .

Σε κάθε περίπτωση, οι αλγόριθμοι για την αριθμητική επίλυση διαφορικών εξισώσεων ποικίλλουν αρκετά και εδώ παρουσιάστηκε η πιο απλή μεθοδολογία. Χρησιμοποιώντας κατάλληλο λογισμικό ή βιβλιοθήκες, μπορούμε πάντα να εκμεταλλευτούμε πολύ πιο περίπλοκους αλγόριθμους οι οποίοι αναφέρονται ως *solvers*.

2.6 Γενετικοί αλγόριθμοι

Οι γενετικοί αλγόριθμοι (ΓΑ) είναι μια κατηγορία αλγορίθμων οι οποίοι εμπίπτουν στους εξελικτικούς αλγόριθμους (ΕΑ), και αποτελούν έναν πιθανό τρόπο επίλυσης προβλημάτων ελαχιστοποίησης. Ονομάζονται έτσι καθώς υπάρχουν αρκετά κοινά στοιχεία με την βιολογική διαδικασία της εξέλιξης ενός είδους, και όπως θα δούμε υπάρχει κοινή ορολογία με όρους βιολογίας σε αρκετά σημεία.

Η γενική ιδέα ενός γενετικού αλγορίθμου ο οποίος χρησιμοποιείται για την επίλυση ενός προβλήματος ακολουθεί τα εξής λογικά βήματα:

1. Δημιουργία ενός *πληθυσμού* (σύνολο αποτελούμενο από πιθανές λύσεις στο εκάστοτε πρόβλημα).
2. Συγκριτική αξιολόγηση της ποιότητας των λύσεων, με βάση κάποιο σύνολο κριτηρίων τα οποία προσδιορίζουν μονοσήμαντα μια *βαθμολογία* στην κάθε λύση.
3. Δημιουργία νέων πιθανών λύσεων:
 - a. Λαμβάνοντας υπ' όψη τα χαρακτηριστικά των λύσεων που ήδη υπάρχουν στον πληθυσμό.
 - b. Κάνοντας χρήση στοχαστικών μεθόδων, οι οποίες προκαλούν απρόβλεπτες αλλαγές στα χαρακτηριστικά του πληθυσμού.
4. Επανάληψη της διαδικασίας από το βήμα #2 μέχρις ότου επέλθει σύγκλιση, όπου τα κριτήρια για την αναγνώριση της εξαρτώνται κάθε φορά από το πρόβλημα προς επίλυση, αλλά σε κάθε περίπτωση θα πρέπει να βασίζονται στην σχετική μεταβολή της ποιότητας των λύσεων με την πάροδο των επαναλήψεων των βημάτων #2 και #3.

2.6.1 Ονοματολογία

Όπως έχει ήδη αναφερθεί, οι γενετικοί αλγόριθμοι προσπαθούν να προσομοιώσουν την βιολογική διαδικασία της εξέλιξης, και εμφανίζουν σε πολλά σημεία ορολογία κοινή με το πεδίο της βιολογίας. Ακολουθεί μια αναφορά στις έννοιες που είναι χρήσιμες σε κάθε γενετικό αλγόριθμο, και αντίστοιχα θα εφαρμοστούν και στην παρούσα μελέτη:

- Πληθυσμός (Population)
Ένας σύνολο από πιθανές λύσεις για ένα δεδομένο πρόβλημα. Τα μέλη του συνόλου αυτού ονομάζονται χρωμοσώματα.
- Χρωμόσωμα (Chromosome)
Μια πιθανή λύση ενός προβλήματος, η οποία αποτελείται από ένα σύνολο τιμών, μία για κάθε παράμετρο του προβλήματος. Τα μέλη του συνόλου αυτού ονομάζονται γονίδια.
- Γονίδιο (Gene)
Μια παράμετρος για το πρόβλημα προς επίλυση. Δεν αναλύεται περαιτέρω.

- Γενιά (Generation)
Ο πληθυσμός μια δεδομένη χρονική στιγμή, ή αλλιώς το σύνολο των χρωμοσωμάτων κατά την διάρκεια μιας επανάληψης του αλγορίθμου.
- Εξέλιξη (Evolution)
Το σύνολο των διεργασιών οι οποίες συμβάλλουν στην αλλαγή των ιδιοτήτων των στοιχείων ενός πληθυσμού (γονιδίων). Η υλοποίηση του μηχανισμού αυτής αποτελεί και το ένα από τα δύο μεγάλα τμήματα ενός αλγορίθμου.
- Διασταύρωση (Crossover)
Η διαδικασία κατά την οποία δύο (ή και περισσότερων) χρωμοσώματα συνδυάζονται με αποτέλεσμα την δημιουργία ενός (ή και περισσότερων) νέων χρωμοσωμάτων. Είναι μια διαδικασία η οποία κατά κύριο λόγο είναι αιτιοκρατική, και επομένως η διασταύρωση παρόμοιων αρχικών συνόλων χρωμοσωμάτων επιφέρει ως αποτέλεσμα την δημιουργία παρόμοιων τελικών συνόλων απογόνων.
- Μετάλλαξη (Mutation)
Η διαδικασία κατά την οποία (συνήθως) οι απόγονοι μιας διασταύρωσης υπόκεινται σε στοχαστικές αλλαγές των γονιδίων τους.
- Φυσική επιλογή (Natural selection)
Η διαδικασία κατά την οποία τα χρωμοσώματα ενός πληθυσμού αξιολογούνται (το κάθε ένα ως προς τα υπόλοιπα), και με βάση την ταξινόμηση τους ως προς την σχετική ποιότητα του κάθε ενός εκείνα τα οποία εμφανίζουν την χειρότερη ωθούνται προς την απομάκρυνση τους από τον πληθυσμό.
- Συνάρτηση καταλληλότητας (Fitness function)
Πρόκειται για τον τρόπο με βάση τον οποίο αξιολογούνται τα χρωμοσώματα ενός πληθυσμού, και ο οποίος έχει την μορφή μιας μαθηματικής συνάρτησης $f(\lambdaύση) = c \in \mathbb{R}$.

Πρόκειται για το δεύτερο μεγάλο ζήτημα που πρέπει να αντιμετωπιστεί κατά την ανάπτυξη ενός γενετικού αλγορίθμου, καθώς η διαδικασία της φυσικής επιλογής στηρίζεται στην δυνατότητα ποσοτικοποίησης των αποτελεσμάτων της συγκριτικής αξιολόγησης ενός πληθυσμού. Σε ένα πολυδιάστατο πρόβλημα συνήθως έχει την μορφή αθροίσματος διαφόρων όρων, όπου ο κάθε ένας είναι σχεδιασμένος για να αξιολογεί μια πτυχή της λύσης.

Το μεγάλο πλεονέκτημα στην χρήση γενετικών αλγορίθμων για επίλυση ενός προβλήματος είναι η δυνατότητα συγκριτικής αξιολόγησης των αποτελεσμάτων καθώς η συνάρτηση καταλληλότητας επιτρέπει την ποσοτική σύγκριση δύο λύσεων μεταξύ τους. Σε κάθε περίπτωση, παρά τις δυνατότητες που προσφέρουν οι ΕΑ και την μεγάλη προσαρμοστικότητα τους, είναι σημαντικό να είναι καλά ορισμένο τι θα αναπαριστά το ένα χρωμόσωμα, καθώς και οι διαδικασίες της διασταύρωσης [27] [28] και μετάλλαξης [29] με βάση το εκάστοτε πρόβλημα.

2.7 Σύνοψη

Στο κεφάλαιο αυτό αναπτύχθηκαν κάποια εργαλεία τα οποία θα χρησιμοποιηθούν από τον τελικό αλγόριθμο για την λειτουργία του, όπως η διαδικασία εισαγωγής χαρτογραφικών δεδομένων και η μετατροπή τους σε μια ψηφιακή αναπαράσταση του ανάγλυφου, μια περιγραφή της γεωμετρικής οντότητας που θα περιγράφει την τροχιά, και ένας τρόπος αναπαράστασης της σε έναν υπολογιστή. Στην συνέχεια παρατέθηκε ένας απλός τρόπος υπολογισμού λύσεων διαφορικών εξισώσεων με την μέθοδο του Euler, και σχολιασμός στο θέμα αριθμητικής επίλυσης. Στο τέλος αναφέρθηκαν κάποια βασικά στοιχεία για τους γενετικούς αλγορίθμους, οι οποίοι θα ξανασυζητηθούν εκτενέστερα σε μετέπειτα κεφάλαιο όπου θα συζητηθεί η εφαρμογή στο συγκεκριμένο πρόβλημα.

Στο επόμενο κεφάλαιο θα αναπτυχθεί ο μηχανισμός με βάση τον οποίο δημιουργείται μια τροχιά, όπου και θα είναι εμφανές πως οι παραπάνω διαδικασίες συνεισφέρουν στην ευκολότερη διαχείριση του προβλήματος.

Κεφάλαιο 3^ο – Μηχανισμός δημιουργίας τροχιάς

3.1 Εισαγωγή

Στο κεφάλαιο αυτό θα παρουσιαστεί η λογική με βάση την οποία λειτουργεί το τμήμα του αλγορίθμου που παράγει μια τροχιά από ένα αρχικό σημείο έως ένα τελικό, δεδομένου ενός ανάγλυφου. Ο μηχανισμός αυτός είναι υπεύθυνος για την παραγωγή μιας τροχιάς που συμμορφώνεται σε κάποιους γεωμετρικούς περιορισμούς, όπως για παράδειγμα η αποφυγή συγκρούσεων, ενώ τελικά για τον υπολογισμό του αποτελέσματος εκτελεί μια προσομοίωση μηχανικής με αριθμητική επίλυση των διαφορικών εξισώσεων που προκύπτουν. Η προσομοίωση αυτή χτίζεται, όπως θα αναπτυχθεί σε αυτό το κεφάλαιο, με βάση το περιβάλλον στο οποίο θα δημιουργηθεί η τροχιά καθώς επίσης και με κάποιες σταθερές αρχές που ισχύουν πάντα ανεξάρτητα του περιβάλλοντος.

3.2 Περιγραφή της διαδικασίας προσομοίωσης

Ο μηχανισμός με τον οποίο δημιουργείται η τροχιά στον συγκεκριμένο αλγόριθμο ακολουθεί μια αιτιοκρατική πορεία, δεδομένης της αρχικής κατάστασης με την οποία τροφοδοτείται και του περιβάλλοντος (δηλαδή, του χάρτη) ενώ η τροχιά αναπαρίσταται όπως έχει ήδη αναφερθεί ως συνεχόμενα ευθύγραμμα τμήματα τα οποία σχηματίζουν μια τεθλασμένη γραμμή. Ο αλγόριθμος που δημιουργεί την τροχιά έχει σχεδιαστεί ώστε να δέχεται ως είσοδο έναν χάρτη, τα όρια της επιθυμητής τροχιάς (αρχικό και τελικό σημείο), και ενδεχομένως κάποια ενδιάμεσα σημεία τα οποία θα χρησιμοποιηθούν ως γεννήτορες για την δημιουργία της τροχιάς.

Η λειτουργία του αλγορίθμου συνοψίζεται στις εξής δύο βασικές λειτουργίες:

1. Προσθήκη ενδιάμεσων σημείων μεταξύ των ακραίων σημείων της τροχιάς
2. Μετατόπιση τους προκειμένου να συμμορφώνονται με τους όποιους περιορισμούς έχουν τεθεί στην τροχιά.

Για την προσθήκη σημείων ο αλγόριθμος υπολογίζει το συνολικό μήκος της τροχιάς με όσα σημεία έχει διαθέσιμα, και εάν η (γραμμική) πυκνότητα των σημείων είναι μικρότερη από κάποιο εκ των προτέρων ρυθμισμένο όριο τότε προσθέτει ένα σημείο στο μέσο του μεγαλύτερου σε μήκος ευθύγραμμου τμήματος που υπάρχει στην τροχιά εκείνη την στιγμή. Μετά την προσθήκη τους, ωστόσο, όλα τα ενδιαμέσα σημεία που δεν είναι ρυθμισμένα ώστε να μην μετακινούνται μετατοπίζονται κατάλληλα ανάλογα με τους παράγοντες που επηρεάζουν την θέση τους. Τέτοιοι παράγοντες είναι η θέση τους σε σχέση με το ανάγλυφο (πρέπει να μεταφερθούν «έξω» από τον όγκο που περικλείει η επιφάνεια του ανάγλυφου), η απόσταση τους από την επιφάνεια του ανάγλυφου, καθώς και το γενικότερο τους ύψος μετρούμενο από το επίπεδο $z = 0$ της αναπαράστασης του χάρτη.

Για να επιτευχθούν τα παραπάνω, ο αλγόριθμος που δημιουργεί την τροχιά αντιμετωπίζει τα σημεία της ως σημειακές μάζες, και τρέχει μια προσομοίωση μηχανικής προκειμένου να εντοπίσει την θέση των σημείων στην οποία το σύστημα ισορροπεί. Η προσομοίωση αυτή γίνεται με επίλυση διαφορικών εξισώσεων κίνησης, οι οποίες προκύπτουν από τους νόμους Νεύτωνα, και η επίλυση τους γίνεται με την μέθοδο του Euler για απλούστευση της διαδικασίας, όπως συζητήθηκε στο προηγούμενο κεφάλαιο:

$$\frac{d\vec{p}}{dt} = \sum \vec{F} \leftrightarrow \vec{p}(t_0 + dt) = \vec{p}(t_0) + dt \sum \vec{F}$$

Ο όρος $\sum \vec{F}$ υπολογίζεται σε κάθε επανάληψη του αλγορίθμου ξεχωριστά για κάθε σημείο που υπάρχει στην προσομοίωση κατά την διάρκεια της επανάληψης, και αποτελείται από υπό-όρους οι οποίοι μπορούν να κατηγοριοποιηθούν ως εσωτερικές δυνάμεις του συστήματος των σωμάτων της τροχιάς ή ως δυνάμεις που εξαρτώνται από το περιβάλλον.

Οι όροι που προέχονται από το ίδιο το σύστημα των σωμάτων είναι οι εξής και αντιστοιχούν στις εσωτερικές δυνάμεις του συστήματος είναι:

1) Δυνάμεις ελατηρίου μεταξύ γειτονικών μαζών

Έχουν την μορφή $\vec{F}_{sij} = -k(|\vec{r}_j - \vec{r}_i| - l_0)\hat{r}_{ij}$, και έχουν προστεθεί ώστε να διατηρείται η συνοχή της τροχιάς. Εξαναγκάζουν όλη την τροχιά να αποκρίνεται στην μετατόπιση ενός μόνο σημείου της, προσθέτοντας και μιας μορφής αδράνεια στην τροχιά ως σύνολο που αντιστέκεται σε απότομες μεταβολές μεταξύ επαναλήψεων. Το φυσικό μήκος των εικονικών αυτών ελατηρίων l_0 ορίζεται σε σχέση με την επιθυμητή γραμμική πυκνότητα των σημείων κατά μήκος της τροχιάς.

2) Δυνάμεις τριβής

Έχουν την μορφή $\vec{F}_d = -b\vec{v}$, δηλαδή προκύπτουν από διαφορική εξίσωση της μορφής $\frac{d\vec{v}}{dt} \propto -\vec{v}$, και εφαρμόζονται σε κάθε ένα σημείο ξεχωριστά. Ο λόγος που συμπεριλαμβάνονται σε αυτή την προσομοίωση είναι προκειμένου να εξασφαλιστεί η απόσβεση της ενέργειας που αποθηκεύεται στις συντηρητικές δυνάμεις (πχ τις δυνάμεις ελατηρίου), και εξασφαλίζουν την τελική ισορροπία του συστήματος. Ο συντελεστής b μπορεί να ρυθμιστεί ώστε η απόσβεση της ενέργειας να έχει γρηγορότερο ή αργότερο ρυθμό, προκειμένου να επιτυγχάνεται νωρίτερα ο εφησυχασμός του συστήματος ή να δίνεται περισσότερος χρόνος στο σύστημα για να βρεθεί η θέση ισορροπίας κατά περίπτωση.

3) Ελκτική δύναμη μεταξύ μη διαδοχικών σημείων

Η δύναμη αυτή έχει την μορφή $\vec{F}_{a ij} = -\frac{d}{dr}\left(\frac{b}{|r|^n}\right)\hat{r}_{ij}$, και είναι σχεδιασμένη ώστε να αναγκάζει τα μη διαδοχικά σημεία να έλκουν το ένα το άλλο. Έχει πολύ μικρή εμβέλεια, και ωφελεί στην δημιουργία συγκρούσεων μεταξύ μη διαδοχικών σημείων, οι οποίες μπορούν να ανιχνευθούν από τον αλγόριθμο. Σε αυτή την περίπτωση, όταν δηλαδή δυο μη διαδοχικά σημεία βρεθούν πάρα πολύ κοντά το ένα με το άλλο, τότε αναγκαστικά η τροχιά κάνει άσκοπους κύκλους αφού περνάει για 2^η φορά από το ίδιο σημείο. Το γεγονός αυτό μπορεί να αναγνωρίζεται, και ο αλγόριθμος αντίστοιχα να απορρίπτει το μέρος της τροχιάς που είναι περιττό μεταξύ των δύο γειτονικών σημείων.

Αντίστοιχα, οι δυνάμεις που εξαρτώνται από την θέση των σημείων σε σχέση με το περιβάλλον είναι οι εξής:

4) Απωστική δύναμη από την επιφάνεια του ανάγλυφου

Έχει την μορφή $\vec{F}_{rep} = -\frac{d}{dr}\left(\frac{a}{r}\right)\hat{r}$, και επιδρά πάνω σε κάθε ένα σημείο της τροχιάς ξεχωριστά. Είναι σχεδιασμένη ώστε τα σημεία της τροχιάς να απωθούνται από την επιφάνεια του ανάγλυφου όταν πλησιάζουν σε αυτό,

5) Απωστική δύναμη από το εσωτερικό του ανάγλυφου

Λόγω της πιθανότητας να δημιουργηθεί ένα νέο σημείο της τροχιάς μέσα στον όγκο που ορίζει η επιφάνεια του ανάγλυφου, στην περίπτωση που ένα σημείο βρίσκεται σε αυτή την κατάσταση τότε επιδρά πάνω του μια δύναμη η οποία το επιταχύνει ώστε να κατευθυνθεί προς την «κοντινότερη έξοδο», δηλαδή το σημείο της επιφάνειας στο οποίο βρίσκεται πιο κοντά. Η δύναμη αυτή είναι σχεδιασμένη ώστε να είναι δυσανάλογη σε σχέση με τις υπόλοιπες, ώστε τα νέα σημεία που προστίθενται να μπορούν να προσφέρουν εποικοδομητικά στην όλη τροχιά το συντομότερο δυνατόν μετά την δημιουργία τους, ακόμη και εάν είναι αρχικά είναι παγιδευμένα στο εσωτερικό του ανάγλυφου.

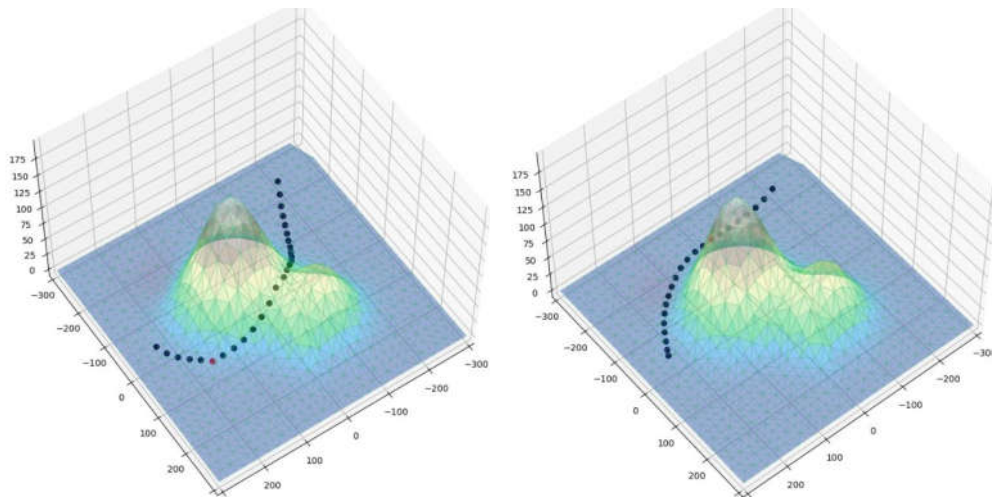
Δεν ακολουθεί κάποιο μαθηματικό μοντέλο, παρά μόνο όπως ήδη αναφέρθηκε έχει κατεύθυνση προς την «κοντινότερη έξοδο» από το εσωτερικό της επιφάνειας ανάλογα της θέσης του σημείου, και είναι δυσανάλογα ισχυρότερη από όλες τις υπόλοιπες δυνάμεις του συστήματος.

6) Δύναμη άνωσης

Έχει την μορφή $\vec{F}_{bw} \propto -\frac{dV(z)}{dz}\hat{z}$, όπου $V(z) = a \cosh\{(z - z_0)^c\}$, με $a, c > 0$ και είναι σχεδιασμένη ώστε να προσπαθεί να φέρει τα σημεία της τροχιάς σε κάποιο προκαθορισμένο ύψος. Οι συντελεστές a, c δίνουν την ευκαιρία για ρύθμιση τόσο της εμβέλειας όσο και της έντασης της δύναμης σε διαφορετικές αποκλίσεις από το επιθυμητό επίπεδο z_0 .

Ο αλγόριθμος μπορεί να λειτουργήσει τροφοδοτώντας τον μόνο με τα δύο ακριανά σημεία της τροχιάς, οπότε και θα ξεκινήσει από μόνος του να προσθέτει ενδιάμεσα σημεία με βάση τα όσα έχουν αναφερθεί παραπάνω. Ωστόσο, εάν δοθούν από την αρχή κάποια ενδιάμεσα σημεία τότε αυτά μπορούν να επηρεάσουν σε μεγάλο βαθμό την έκβαση του αλγορίθμου, και αντίστοιχα αυτή η ιδιότητα θα εκμεταλλευθεί για την χρήση του σε συνδυασμό με έναν γενετικό αλγόριθμο όπως θα δούμε στα επόμενα κεφάλαια.

Ένα παράδειγμα δύο τέτοιων τροχιών οι οποίες δημιουργούνται στον ίδιο χάρτη, με τα ίδια ακραία σημεία, αλλά χρησιμοποιώντας ένα αρχικό ενδιάμεσο σημείο με διαφορετικές συντεταγμένες σε κάθε περίπτωση για να επηρεαστεί ο αλγόριθμος φαίνεται στην Εικόνα 14:



Εικόνα 11 - Δύο διαφορετικές λύσεις του αλγορίθμου για το ίδιο πρόβλημα

Algorithm 1: Trajectory generation using mechanics principles

Data: Problem Setup, waypoint seeds (if any)**Result:** A polygonal chain path between initial and terminal points

```
/* initialize with a seed, if none are given */
if seed count == 0 then
    | seeds  $\leftarrow$   $0.5 * [r_{initial} + r_{final}]$ 
end

waypoints  $\leftarrow$  [start point, seeds, endpoint]

while not converged do
    /* calculate the new dynamics, and apply them at the same
       time */
    for waypoint in waypoints do
        | acceleration  $\leftarrow$  sum_of_forces / point_mass
    end

    for waypoint in waypoints do
        | velocity = velocity + acceleration * dt
        | position = position + velocity * dt
    end

    /* Check for knots */
    for waypoint_pair in waypoints do
        | if pair_distance < minimum_distance then
        | | remove_chain_segment_between_pair
        | | remove_one_node_of_pair
        | end
    end

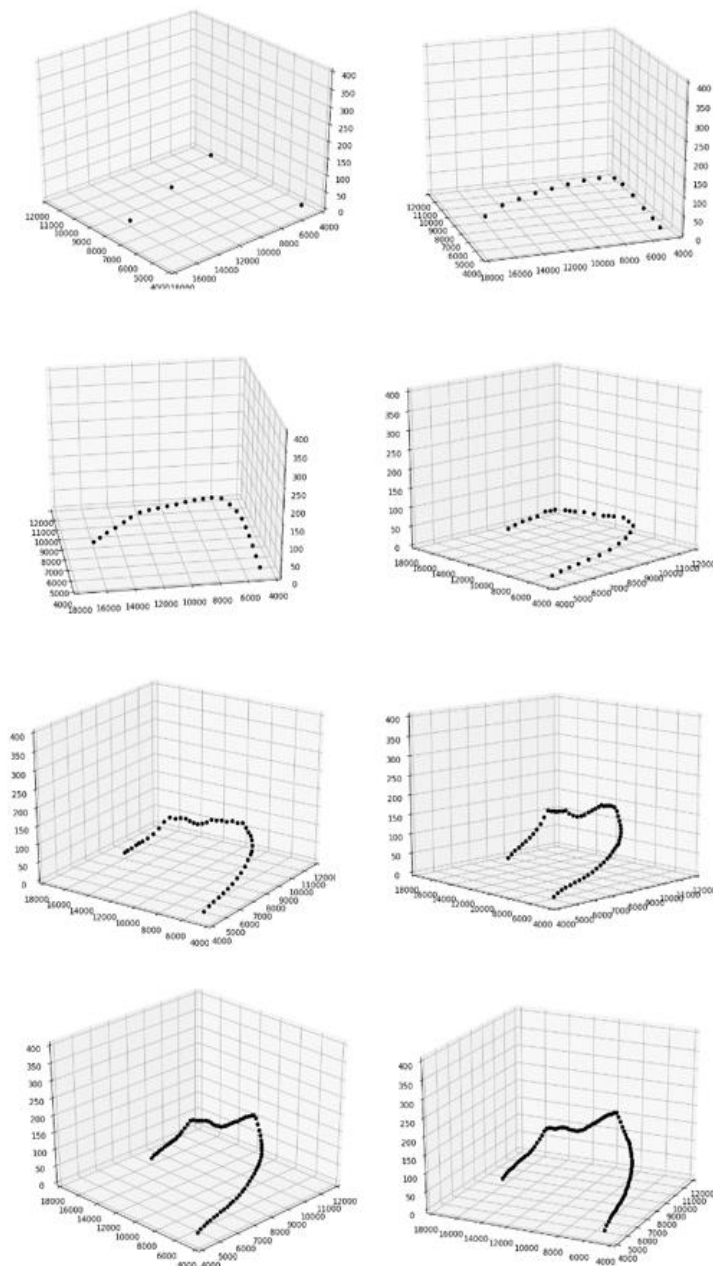
    /* Check for rest and equilibrium */
    if average_velocity < minimum_velocity then
        | if average_acceleration < minimum_acceleration then
        | | if average_spring_elongation >  $0.10 * spring\_equilibrium\_length$  then
        | | | /* insert another node */
        | | | target_leg  $\leftarrow$  locate_longest_leg
        | | | insert_waypoint_in_middle(target_leg)
        | | else
        | | | converged  $\leftarrow$  true
        | | end
        | end
    end
end
```

2

Εικόνα 12 - Ψευδοκώδικας για τον αλγόριθμο δημιουργίας τροχιάς

Η πορεία της εξέλιξης της τροχιάς φαίνεται στα στιγμιότυπα στην παρακάτω Εικόνα 16, όπου έχει αφαιρεθεί το ανάγλυφο του εδάφους για να είναι πιο ευδιάκριτη η μεταβολή

των θέσεων των σημείων. Αυτό που μπορεί να παρατηρηθεί εύκολα είναι το αρχικό πύκνωμα των σημείων στα αρχικά στάδια του αλγορίθμου και την κατόπιν συμμόρφωση τους με το ανάγλυφο, καθώς και την συνεχόμενη αναπροσαρμογή των θέσεων τους σε κάθε εικόνα.



Εικόνα 13 - Στιγμιότυπα διαδικασίας δημιουργίας τροχιάς

3.3 Σύνοψη

Στο κεφάλαιο αυτό αναπτύχθηκε μια μεθοδολογία με βάση την οποία μπορεί να δημιουργηθεί μια τροχιά μεταξύ δύο σημείων, η οποία συμμορφώνεται στο ανάγλυφο και σε λοιπούς περιορισμούς γεωμετρικής φύσεως. Η διαδικασία δημιουργίας της τροχιάς βασίζεται σε κάποιες απλές αρχές μηχανικής, και προκειμένου να χτιστεί ένας αλγόριθμος που θα ακολουθεί αυτή την λογική το πρόβλημα της εύρεσης τροχιάς ανάγεται σε ένα πρόβλημα διαφορικών εξισώσεων το οποίο λύνεται με αριθμητικές μεθόδους.

Στο επόμενο κεφάλαιο θα συζητηθεί πως αυτή η καθαρά αιτιοκρατική διαδικασία μπορεί να συνδυαστεί με έναν γενετικό αλγόριθμο, προκειμένου να μπορέσει τελικά να παραχθεί μια βελτιστοποιημένη λύση στο πρόβλημα.

Κεφάλαιο 4° – Μεθοδολογία με γενετικούς αλγόριθμους

4.1 Εισαγωγή

Στο προηγούμενο κεφάλαιο παρουσιάστηκε ένας αλγόριθμος ο οποίος μπορεί να σχεδιάσει μια τροχιά, βασισμένος σε γεωμετρικές αρχές και αρχές μηχανικής. Από μόνος του αυτός ο αλγόριθμος λειτουργεί αιτιοκρατικά και επομένως δεν μπορεί να χρησιμοποιηθεί για την εύρεση μιας τροχιάς βελτιστοποιημένης ως προς κάποιο γνώρισμα. Σε αυτό το κεφάλαιο, θα συζητηθεί το πάντρεμα του αλγορίθμου αυτού με έναν γενετικό αλγόριθμο ο οποίος θα αποτελέσει το εργαλείο με το οποίο θα μπορούσαμε να αξιολογήσουμε τις παραγόμενες τροχιές ως προς τις ιδιότητες τους, και να οδηγηθούμε στην εύρεση βελτιστοποιημένων λύσεων.

Τα βασικά σημεία που θα παρατεθούν είναι η παρουσίαση του προβλήματος στην γλώσσα των γενετικών αλγορίθμων, καθώς και η δημιουργία μιας συνάρτησης καταλληλότητας που να αντιστοιχεί στο πρόβλημα που θέλουμε να λύσουμε. Οι ιδέες που θα παρουσιαστούν στη συνέχεια βασίζονται στη μεθοδολογία που αναπτύχθηκε στην [18].

4.2 Ο γενετικός αλγόριθμος

Η διαδικασία δημιουργίας τροχιάς που αναπτύχθηκε στην προηγούμενη παράγραφο είναι καθαρά αιτιοκρατική, επομένως για κάθε ίδιο πρόβλημα με το οποίο τροφοδοτείται (χάρτης, αρχικές συνθήκες) θα παράγει πάντα την ίδια ακριβώς λύση. Προκειμένου να βρεθεί μια βελτιστοποιημένη λύση στο πρόβλημα υπό εξέταση, σε σχέση με κάποιο ή κάποια κριτήρια, έχει σχεδιαστεί ένας γενετικός αλγόριθμος ο οποίος για το δεδομένο πρόβλημα τρέχει τον αλγόριθμο παραγωγής τροχιάς με διαφορετικές αρχικές συνθήκες και αξιολογεί τα αποτελέσματα, προσπαθώντας να εκμαιεύσει μια βέλτιστη λύση.

Όπως έχει ήδη αναφερθεί προηγουμένως, στον γενετικό αλγόριθμο ένα χρωμόσωμα αντιστοιχεί σε ένα σύνολο αρχικών θέσεων των ενδιάμεσων σημείων της τροχιάς, και αντίστοιχα τα γονίδια του είναι οι συντεταγμένες του κάθε ενός σημείου. Αυτό σημαίνει ότι τα γονίδια δεν έχουν απαραίτητα σταθερό πλήθος χρωμοσωμάτων από πρόβλημα σε

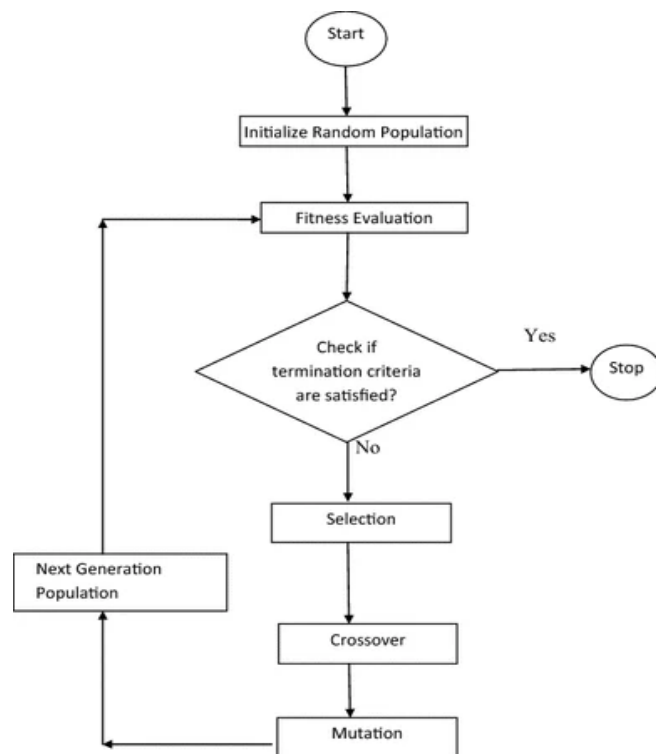
πρόβλημα, αλλά σε κάθε περίπτωση ο αλγόριθμος θα πρέπει να προσαρμόζεται στο μήκος των χρωμοσωμάτων που χρησιμοποιούνται.

Οι διαδικασίες της μετάλλαξης και της διασταύρωσης στην προκειμένη περίπτωση έχουν οριστεί με βάση γενικές κατευθύνσεις από την βιβλιογραφία [30]. Πιο συγκεκριμένα, για την μετάλλαξη έχει οριστεί μια καθαρά τυχαία προσέγγιση μετατόπισης ενός σημείου ελέγχου μέσα στον χώρο ενώ για την διασταύρωση δύο χρωμοσωμάτων επιλέγεται ένα τυχαίο γονίδιο και προκύπτει ένα νέο χρωμόσωμα όπου τα γονίδια του μέχρι το τυχαία επιλεγμένο προκύπτουν από τον ένα γονέα, ενώ τα υπόλοιπα από τον δεύτερο.

Στην συνέχεια ο αλγόριθμος προχωράει στην εκτίμηση των αποτελεσμάτων που δημιουργούνται από κάθε σύνολο αρχικών θέσεων, και με την πάροδο των γενιών του πληθυσμού προσπαθεί να μεταβάλλει τις τιμές των αρχικών θέσεων προκειμένου να βρεθεί η καλύτερη δυνατή λύση το πρόβλημα με βάση την αξιολόγηση μέσω της συνάρτησης καταλληλότητας. Για να γίνει πιο ρεαλιστική/εφαρμόσιμη η υλοποίηση, τα κριτήρια με τα οποία βαθμολογεί ο γενετικός αλγόριθμος μια τροχιά είναι κριτήρια που σχετίζονται με τα επιθυμητά χαρακτηριστικά της αποστολής που θέλουμε να εκτελέσει το μη επανδρωμένο όχημα, αλλά εμπεριέχονται και κριτήρια που σχετίζονται με την γεωμετρία της τροχιάς και αντιστοιχούν στην ικανότητα πλοήγησης του οχήματος, όπως για παράδειγμα πόσο απότομες αλλαγές κατεύθυνσης μπορεί να πραγματοποιήσει ένα όχημα ή κάποιο υψόμετρο που θεωρείται όριο για την λειτουργία του.

Επιπλέον, για την λειτουργία του γενετικού αλγορίθμου χρειάζεται να οριστεί μια παράμετρος που θα αντιστοιχεί στο πλήθος των χρωμοσωμάτων που θα διατηρεί ο αλγόριθμος σε κάθε γενιά. Γενικά, μεγαλύτερος πληθυσμός μπορεί να οδηγήσει σε αυξημένη ποικιλότητα των λύσεων αλλά αντίστοιχα θα απαιτεί περισσότερο υπολογιστικό χρόνο για τις περισσότερες διαδικασίες, επομένως αφήνεται ως παράμετρος προς ρύθμιση κατά περίπτωση.

Παρότι υπάρχει μια πιθανότητα για όσο και να τρέχει ο αλγόριθμος αυτός να βρίσκει ολοένα και καλύτερες λύσεις, έχει οριστεί ως κριτήριο σύγκλισης για τον τερματισμό των επαναλήψεων η σταθεροποίηση της τιμής της συνάρτησης καταλληλότητας για παραπάνω από 100 γενιές (αυθαίρετα), ενώ παράλληλα χρησιμοποιείται και ένα κριτήριο τερματισμού με βάση τον αριθμό των γενιών που έχουν παρέλθει. Δεδομένου ότι η βέλτιστη λύση στο πρόβλημα δεν είναι γνωστή, δεν είναι δυνατόν να οριστεί κάποιο κριτήριο τερματισμού σχετικό με την τιμή της συνάρτησης καταλληλότητας, καθώς δεν μπορούμε να γνωρίζουμε *a priori* την τιμή που αντιστοιχεί σε αυτή τη λύση.



Εικόνα 14 - Διάγραμμα ροής γενετικού αλγορίθμου

Στο διάγραμμα αυτό, στο βήμα «fitness evaluation» κρύβεται ο προηγούμενος αλγόριθμος, δηλαδή η δημιουργία των τροχιών η οποία πραγματοποιείται με βάση τα seeds που δίνονται από τον γενετικό αλγόριθμο (τα γονίδια του αλγορίθμου). Αυτό είναι και το κομμάτι το οποίο απαιτεί τον περισσότερο χρόνο, καθώς η προκειμένου να πραγματοποιηθεί η αξιολόγηση πρέπει να δημιουργηθούν όλες οι τροχιές που αντιστοιχούν σε μια γενιά χρωμοσωμάτων του γενετικού αλγορίθμου.

4.3 Συνάρτηση καταλληλότητας

Η συνάρτηση καταλληλότητας, όπως συνηθίζεται σε σύνθετα προβλήματα, είναι ένα άθροισμα πολλών όρων. Κάθε όρος είναι σχεδιασμένος για να αντιπροσωπεύει την ποιότητα της λύσης ως προς ένα συγκεκριμένο χαρακτηριστικό, και ο συνδυασμός των όρων δίνει μια συνολική βαθμολογία για κάθε λύση. Δεδομένης της πολυπλοκότητας των κριτηρίων, μπορεί σε κάποια περίπτωση να δοθεί εύνοια σε μια λύση που ανταποκρίνεται καλύτερα σε κάποιον περιορισμό ο οποίος κρίνεται κατά περίπτωση πιο σημαντικός, απλά μεταβάλλοντας τους συντελεστές βαρύτητας των όρων.

Η γενική μορφή της συνάρτησης καταλληλότητας είναι γραμμικός συνδυασμός, δηλαδή $F(s) = \sum a_i f_i(s)$, επομένως τα βάρη a_i είναι οι συντελεστές που μπορούν να ωθήσουν τον γενετικό αλγόριθμο να προτιμήσει μια λύση που εμφανίζει κάποιο επιθυμητό χαρακτηριστικό.

Οι όροι που έχουν συμπεριληφθεί μέχρι στιγμής στην συνάρτηση καταλληλότητας, κρίνουν μια λύση με βάση την απόδοση της στις εξής κατηγορίες:

- 1) Αποφυγή συγκρούσεων με το ανάγλυφο

$$f_1 = \sum_{i=1}^N \frac{H(z_i - z(x_i, y_i))}{N}$$

Ο όρος αυτός υπολογίζει το ποσοστό των σημείων τα οποία βρίσκονται «μέσα» στον όγκο που ορίζεται από την επιφάνεια του ανάγλυφου. Κανονικά δεν θα πρέπει να υπάρχει κανένα σημείο που να πληροί αυτές τις προϋποθέσεις, με βάση την σχεδίαση του αλγορίθμου δημιουργίας τροχιάς, ωστόσο σε περίπτωση που ο αλγόριθμος τερματίσει πρόωρα την προσομοίωση μπορεί να εμφανιστεί κάποιο τέτοιο σημείο. Σε κάθε περίπτωση, είναι το πλέον ανεπιθύμητο χαρακτηριστικό για μια τροχιά, επομένως συνοδεύεται και από αρκετά υψηλό συντελεστή α_1 .

Η συνάρτηση $H(x)$ είναι η συνάρτηση βήματος Heaviside, ενώ η συνάρτηση $z(x, y)$ είναι το ύψος (ASL) του ανάγλυφου στο σημείο (x, y) και z_i το ύψος (ASL) του i -οστού waypoint.

2) Απόσταση τροχιάς από το ανάγλυφο

$$f_2 = \sum_{i=1}^N \frac{-H(|(x_i, y_i, z_i) - (x'_i, y'_i, z'_i)| - l_0)}{N}$$

Ο όρος αυτός υπολογίζει το ποσοστό των σημείων της τροχιάς τα οποία βρίσκονται πιο κοντά στο ανάγλυφο από μια ελάχιστη απόσταση ασφαλείας (l_0), προκειμένου να ευνοήσει τροχιές που έχουν μεγαλύτερα περιθώρια. Η απόσταση από το ανάγλυφο υπολογίζεται εντοπίζοντας για κάθε σημείο της τροχιάς (x_i, y_i, z_i) το κοντινότερο σημείο από όλο το ανάγλυφο (x'_i, y'_i, z'_i) , και παίρνοντας κατόπιν την Ευκλείδεια απόσταση τους.

Ο ίδιος όρος μπορεί, αν αντιστραφεί το πρόσημο του, να εξυπηρετήσει και τροχιές με το ακριβώς ανάποδο προφίλ όπου το όχημα θα περνάει όσο πιο κοντά στο έδαφος γίνεται (σε σχέση με την απόσταση l_0)

3) Καμπυλότητα τροχιάς

$$f_3 = \sum_{i=2}^{N-1} \frac{H(r_{curv}(i-1, i, i+1) - r_{max})}{N-2}$$

Κάθε τροχιά που δημιουργείται αξιολογείται από αυτόν τον όρο σε σχέση με την ακτίνα καμπυλότητας της. Το ζητούμενο είναι να μην επιλεγεί τελικά μια τροχιά η οποία περιέχει καμπύλες που δεν μπορούν να ακολουθηθούν από το όχημα, καθώς κάθε όχημα έχει κάποιους περιορισμούς στις δυνατότητες του (flight envelope). Ο όρος r_{max} είναι μια παραδοχή με βάση τα χαρακτηριστικά του εκάστοτε οχήματος, καθώς σε καμία περίπτωση δεν μπορεί να αντικατοπτρισθεί με μια μόνο τιμή αλλά θα εξαρτάται από πολλές παραμέτρους. Ωστόσο, με τον συγκεκριμένο τρόπο εάν λάβουμε υπόψη την χερίστη πιθανή τιμή από όλο το flight envelope μπορούμε να ορίσουμε μια τιμή για την παράμετρο αυτή.

Για τον υπολογισμό της καμπυλότητας της τροχιάς χωρίζεται η τροχιά σε τμήματα τριών συνεχόμενων waypoints (δύο ευθύγραμμων τμημάτων της τεθλασμένης γραμμής), και από τα 3 αυτά σημεία υπολογίζουμε την ακτίνα του κύκλου ο οποίος διέρχεται από αυτά, ως εξής:

Υποθέτοντας τρία σημεία $P_1 = (x_1, y_1)$, $P_2 = (x_2, y_2)$, $P_3 = (x_3, y_3)$ τα οποία βρίσκονται στον κύκλο με εξίσωση

$$Ax^2 + Ay^2 + Bx + Cy + D = 0$$

και δεδομένου ότι η εξίσωση αυτή ικανοποιείται από όλα τα σημεία του κύκλου άρα και από τα τρία αυτά σημεία, μπορούμε να γράψουμε τις εξισώσεις αυτές σε μορφή πίνακα ως εξής

$$\begin{bmatrix} x^2 + y^2 & x & y & 1 \\ x_1^2 + y_1^2 & x_1 & y_1 & 1 \\ x_2^2 + y_2^2 & x_2 & y_2 & 1 \\ x_3^2 + y_3^2 & x_3 & y_3 & 1 \end{bmatrix} \begin{bmatrix} A \\ B \\ C \\ D \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

Το παραπάνω αποτελεί ένα σύστημα γραμμικών εξισώσεων ως προς τις μεταβλητές A, B, C, D , και εάν τα τρία σημεία δεν είναι συγγραμικά τότε εξ ορισμού ορίζουν έναν κύκλο και επομένως το σύστημα εξισώσεων έχει μια μοναδική λύση. Αυτό σημαίνει ότι ο πίνακας των συντελεστών έχει μηδενική ορίζουσα, και άρα μπορούμε να υπολογίσουμε τους άγνωστους όρους ως εξής

$$A = \begin{vmatrix} x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \\ x_3 & y_3 & 1 \end{vmatrix}, \quad B = - \begin{vmatrix} x_1^2 + y_1^2 & y_1 & 1 \\ x_2^2 + y_2^2 & y_2 & 1 \\ x_3^2 + y_3^2 & y_3 & 1 \end{vmatrix}$$

$$C = \begin{vmatrix} x_1^2 + y_1^2 & x_1 & 1 \\ x_2^2 + y_2^2 & x_2 & 1 \\ x_3^2 + y_3^2 & x_3 & 1 \end{vmatrix}, \quad D = - \begin{vmatrix} x_1^2 + y_1^2 & x_1 & y_1 \\ x_2^2 + y_2^2 & x_2 & y_2 \\ x_3^2 + y_3^2 & x_3 & y_3 \end{vmatrix}$$

Γνωρίζοντας πλέον τις τιμές των παραμέτρων, η ακτίνα του κύκλου μπορεί να υπολογιστεί ως

$$R = \sqrt{\frac{B^2 + C^2 - 4AD}{2A^2}}$$

Τα παραπάνω αφορούν έναν κύκλο στο επίπεδο, οπότε για τον υπολογισμό από τον αλγόριθμο της ακτίνας του κύκλου που περνάει από τα τρία σημεία στον χώρο πρώτα δημιουργείται η προβολή τους στο επίπεδο που ορίζουν με τον αντίστοιχο μετασχηματισμό συντεταγμένων και κατόπιν εφαρμόζεται η παραπάνω μεθοδολογία.

4) Διατήρηση επιπέδου πτήσης (flight level)

$$f_4 = \sum_{i=1}^N \frac{|z_i - z_{desired}|}{N}$$

Ο όρος αυτός υπολογίζει την απόκλιση των σημείων της τροχιάς από το επιθυμητό ύψος (ASL) που μπορεί να οριστεί ως παράμετρος. Σε κάθε περίπτωση αποκλίσεις μπορεί να υπάρχουν, εάν το επιθυμητό ύψος είναι χαμηλότερο από το μέγιστο ύψος του ανάγλυφου επομένως ο όρος δεν διακριτοποιείται με κάποια συνάρτηση και απλά λογίζεται η πραγματική απόκλιση.

5) Συνολικό μήκος τροχιάς

$$f_5 = \left(\sum_{i=1}^{N-1} L_{ij} \right) / L^*, \quad j = i + 1$$

Ο όρος αυτός υπολογίζει και βαθμολογεί την τροχιά ως σύνολο σε σχέση με το μήκος της, θεωρώντας το συνολικό μήκος ως το άθροισμα των επιμέρους μηκών για κάθε ευθύγραμμο τμήμα, $L_{ij} = |(x_i, y_i, z_i) - (x_j, y_j, z_j)|$.

Προκειμένου να υπάρχει ένα κριτήριο σύγκρισης, ο όρος L^* είναι κάθε φορά (για το δεδομένο πρόβλημα και την ως τώρα εξέλιξη του γενετικού αλγορίθμου) το βέλτιστο αποτέλεσμα από τους έως τώρα υπολογισμούς μήκους, επομένως ο όρος είναι ένα ποσοστό το οποίο δείχνει την ποσοστιαία διαφορά σε μήκος της τροχιάς υπό εξέταση σε σχέση με την βέλτιστη ως τώρα γνωστή λύση (βέλτιστη σε ότι αφορά το μήκος και μόνο)

6) Προσανατολισμός (heading) τροχιάς

$$f_6 = \sum_{i=1}^{N-1} \frac{\sin\theta(x_i, y_i)}{N-1}$$

Τέλος, κάθε τροχιά βαθμολογείται σε σχέση με την κατεύθυνση των επιμέρους ευθύγραμμων τμημάτων της. Μια τέτοια μέτρηση μπορεί να χρησιμοποιηθεί προκειμένου να επιβραβεύσει τροχιές οι οποίες δεν φέρνουν το όχημα αντιμέτωπο με διευθύνσεις ανέμου που δεν είναι επιθυμητές όπως για παράδειγμα σκέλη της τροχιάς με πλάγιο άνεμο. Επίσης, στην περίπτωση που λαμβάνεται υπόψη η διεύθυνση του ανέμου μπορούν να ευνοηθούν οι τροχιές με σκέλη που παρουσιάζουν μετωπικό άνεμο όπου και η διεύθυνση του ανέμου προσφέρει επιπλέον άνοση χωρίς επιπλέον ενεργειακό κόστος για το όχημα, ή ακόμη και σκέλη με ούριο άνεμο αναλόγως των απαιτήσεων της αποστολής.

Στον όρο αυτό η γωνία $\theta(x, y)$ είναι η απόκλιση από την διεύθυνση του ανέμου στην θέση (x, y) και ενώ στην παρούσα μελέτη υποθέτουμε σταθερό άνεμο σε όλο το εύρος του χάρτη, θα μπορούσαν να εισαχθούν και μετεωρολογικά δεδομένα που να αντικατοπτρίζουν την πραγματική κατάσταση την χρονική στιγμή που θα πραγματοποιηθεί η αποστολή.

Οι όροι της συνάρτησης καταλληλότητας που παρατέθηκαν αποτελούν έναν βασικό πυρήνα για την αξιολόγηση μιας λύσης, με τους περισσότερους να αντιστοιχούν σε κριτήρια τα οποία εμφανίζονται συχνά σε διάφορες προσεγγίσεις επίλυσης του προβλήματος σχεδιασμού τροχιάς. Σε κάθε περίπτωση, η συνάρτηση έχει σχεδιαστεί έτσι ώστε οι όροι της να μην εξαρτώνται από το πλήθος των σημείων της τροχιάς, ή από το συνολικό της μήκος στον βαθμό που αυτό είναι εφικτό. Αυτό δίνει την δυνατότητα στην συνάρτηση καταλληλότητας να μπορεί να αξιολογήσει τροχιές οι οποίες έχουν διαφορετική πυκνότητα και ποσότητα σημείων, αλλά ωστόσο αντιστοιχούν σε λύσεις του ίδιου προβλήματος.

Τα βήματα του τελικού αλγορίθμου (συνδυασμού γενετικού κώδικα με τον αλγόριθμο δημιουργίας τροχιάς) μπορούν να συνοψισθούν ως εξής:

Algorithm 1: Calculation of trajectory curvature from 3 discrete points

Result: The best candidate path/solution

Data: Problem setup, path generation algorithm configuration, genetic algorithm configuration

```

/* Initialize a random population of  $N_c$  chromosomes and  $N_g$ 
   genes, where each gene is a random point in the map */

population ← random_population( $N_c$ ,  $N_g$ )
best_fitness ←  $\infty$ 
iterations ← 0

while not converged do
    iterations ← iterations + 1
    /* Create new generation, with crossover and mutations */
    for each random pair of chromosomes in population do
        offspring ← crossover(chromosome1, chromosome2)
        offspring ← probabilistic_mutation(offspring)
        population ← population + offspring
    end

    /* Grade the generated paths according to the fitness
       function */
    for chromosome in population do
        chromosome_path ← generate_path_from_chromosome
        chromosome_score ← fitness_function_evaluation(chromosome_path)
    end

    sort_population_by_fitness

    /* Check for convergence */
    if current_best_score < best_score then
        best_score ← current_best_score
        /* Keep population number in check */
        while population count >  $N_c$  do
            discard_lowest_performing_chromosome()
        end
    else
        if iterations > max. iterations then
            converged ← true
        end
    end
end
end

```

Εικόνα 15 - Ψευδοκώδικας για τον συνολικό αλγόριθμο

4.4 Σύνοψη

Στο κεφάλαιο αυτό παρουσιάστηκε ο γενετικός αλγόριθμος που θα λειτουργεί επικουρικά με τον αιτιοκρατικό αλγόριθμο που αναπτύχθηκε εκτενώς στο προηγούμενο κεφάλαιο για την δημιουργία τροχιάς, και δόθηκε ιδιαίτερη έμφαση στην συνάρτηση καταλληλότητας και τους επιμέρους όρους της. Οι όροι που παρουσιάστηκαν αντιστοιχούν σε κριτήρια που εμφανίζονται συχνά σε προσεγγίσεις επίλυσης του προβλήματος σχεδιασμού τροχιάς, ενώ η κατά τα άλλα γενική υλοποίηση του αλγορίθμου είναι τυπική με βάση τη βιβλιογραφία.

Στο επόμενο κεφάλαιο θα δοκιμαστεί ο συνδυασμός αυτών των δύο σε μια πραγματική εφαρμογή, προκειμένου να μελετηθεί εάν ο αλγόριθμος πετυχαίνει τον στόχο του.

Κεφάλαιο 5^ο – Εφαρμογή σε πραγματικά χαρτογραφικά δεδομένα

5.1 Εισαγωγή

Για την δοκιμή του αλγορίθμου, θα χρησιμοποιηθούν δεδομένα μορφής DTED από την υπηρεσία NOAA των ΗΠΑ, τα οποία απεικονίζουν την ευρύτερη περιοχή του Αργοσαρωνικού κόλπου κοντά στην Αττική. Προκειμένου να χρησιμοποιηθεί ένα ανάγλυφο με ιδιομορφίες, έχει επιλεγεί ένα νησί και ζητείται από τον αλγόριθμο να υπολογίζει μια τροχιά για ένα μη επανδρωμένο σκάφος το οποίο θα διασχίσει το νησί αυτό.

Για την διερεύνηση των δυνατοτήτων του αλγορίθμου θα ζητηθεί να παραχθούν τροχιές οι οποίες να πληρούν διαφορετικά κριτήρια, μέσω της επιλογής κατάλληλων σετ συντελεστών για τους επιμέρους όρους στην συνάρτηση καταλληλότητας. Το αναμενόμενο αποτέλεσμα είναι να δημιουργηθούν διαφορετικές τροχιές με τα ίδια ακραία σημεία, οι οποίες να πληρούν διαφορετικές προδιαγραφές κάθε φορά.

Στην προκειμένη περίπτωση, ο αλγόριθμος όπως έχει περιγραφεί στα προηγούμενα κεφάλαια υλοποιήθηκε σε γλώσσα Python, χάριν ευκολίας, και δοκιμάστηκε σε έναν μέτριων επιδόσεων οικιακό υπολογιστή. Αυτό είχε ως αποτέλεσμα την μεγάλη χρονική απαίτηση για την επίλυση των προβλημάτων καθώς όπως έχει ήδη σχολιαστεί ο αλγόριθμος είναι ιδιαίτερα απαιτητικός σε επεξεργαστική ισχύ και ακόμη και η επιλογή της γλώσσας υλοποίησης μπορεί να έχει σημαντική διαφορά στις επιδόσεις.

5.2 Χαρτογραφικά δεδομένα

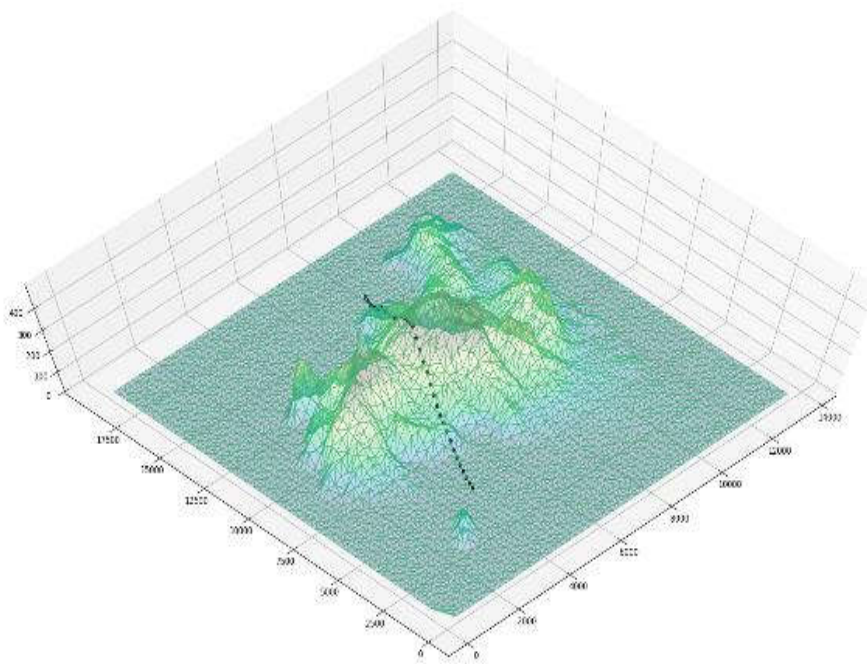
Για τον χάρτη, το σύνολο δεδομένων που χρησιμοποιήθηκε ήταν σε μορφή DTED επιπέδου 2, το οποίο μεταφράζεται ως ένα σημείο δεδομένων κάθε περίπου 20 μέτρα. Η αύξηση της πυκνότητας των αρχικών δεδομένων έχει επίδραση στον υπολογιστικό χρόνο του αλγορίθμου, καθώς αυξάνεται η πολυπλοκότητα του τριγωνισμού και επομένως έχει γίνει μια δειγματοληψία των δεδομένων σε κλίμακα 1:5. Ο ρυθμός δειγματοληψίας για καλύτερα αποτελέσματα μπορεί να είναι μεταβλητός, καθώς σε περιοχές που δεν υπάρχει μεγάλη διαφοροποίηση μπορεί να είναι πολύ πιο μικρός χωρίς να επηρεαστεί αρνητικά ο αλγόριθμος, ενώ σε περιοχές που παρουσιάζουν μεγάλη

ποικιλομορφία με μεγάλες αλλαγές στο υψόμετρο σε σχετικά μικρές αποστάσεις έχει νόημα να κρατηθεί σε πιο υψηλά επίπεδα.

5.3 Συντομότερη δυνατή τροχιά

Για την εύρεση της συντομότερης δυνατής τροχιάς, τροποποιείται κατάλληλα η συνάρτηση καταλληλότητας έτσι ώστε ο συντελεστής του όρου a_5 να έχει αρκετά μεγαλύτερη τιμή σε σχέση με τους υπόλοιπους.

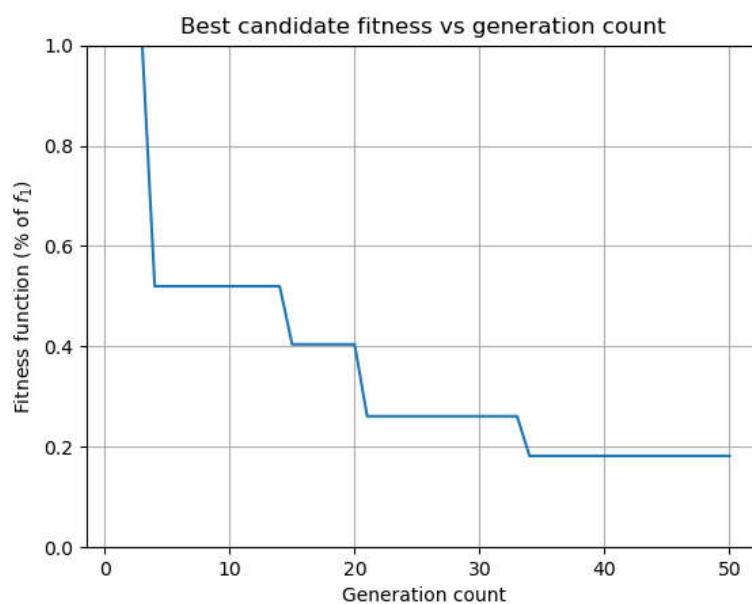
Το αποτέλεσμα του αλγορίθμου για αυτό το σενάριο φαίνεται στην Εικόνα 19:



Εικόνα 16 - Συντομότερη δυνατή τροχιά

Παρατηρούμε ότι ο αλγόριθμος εδώ έχει δημιουργήσει μια τροχιά η οποία είναι (προβεβλημένη στο επίπεδο xy) μια ευθεία γραμμή, η οποία σε πραγματικές συνθήκες αγκαλιάσει το ανάγλυφο και προσαρμόζεται ώστε να διατηρήσει τα επίπεδα ασφαλείας που έχουν ορισθεί. Το αποτέλεσμα είναι αναμενόμενο λόγω της τροποποίησης της συνάρτησης καταλληλότητας, και αναλύοντας τα επιμέρους σημεία σε σχέση με τα ζητούμενα φαίνεται η ικανοποίηση των περιορισμών.

Σε ότι αφορά το γενετικό κομμάτι του αλγορίθμου, η πορεία της συνάρτησης καταλληλότητας με την πάροδο των γενιών του πληθυσμού φαίνεται στην επόμενη γραφική παράσταση όπου απεικονίζεται η συνάρτηση καταλληλότητας για το καλύτερο χρωμόσωμα κάθε γενιάς.



Εικόνα 17 - Συνάρτηση καταλληλότητας

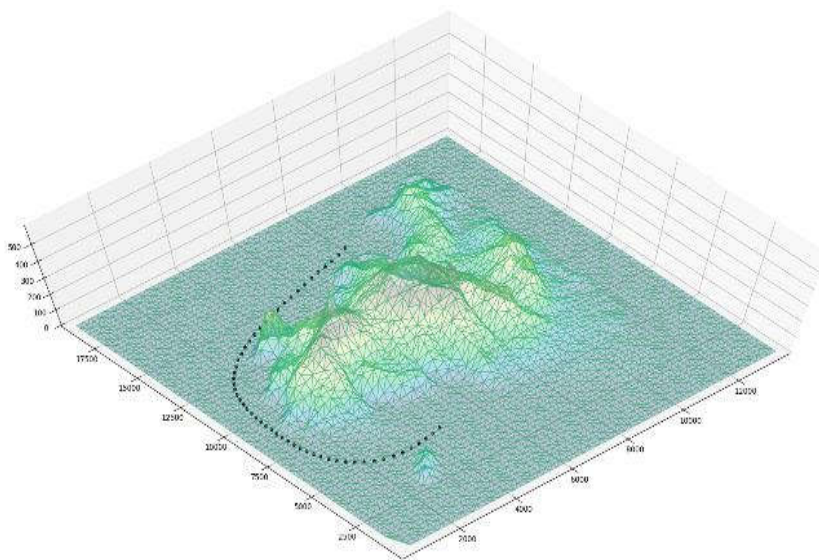
Τα «σκαλοπάτια» στην συνάρτηση καταλληλότητας στην Εικόνα 20 είναι τυπικό γνώρισμα ενός γενετικού αλγορίθμου, καθώς λόγω της στοχαστικής του φύσης δεν υπάρχει καμία εγγύηση για την βελτίωση του πληθυσμού από γενιά σε γενιά, και σε πολλές περιπτώσεις η συνάρτηση καταλληλότητας φαίνεται να μένει στάσιμη. Το

νούμερο που προκύπτει από την συνάρτηση καταλληλότητας δεν έχει κάποιο νόημα, όχι τουλάχιστον στην παρούσα υλοποίηση της, και αντιμετωπίζεται ως καθαρός αριθμός χωρίς μονάδες. Σημασία έχει ωστόσο η διαφοροποίηση της από γενιά σε γενιά, και γι' αυτό τον λόγο παρατίθεται ως ποσοστό σε σχέση με την αρχική γενιά του πληθυσμού.

5.4 Τροχιά με δύο κριτήρια

Για αυτή την δοκιμή, τροποποιήθηκε στον αλγόριθμο η συνάρτηση καταλληλότητας ώστε να δοθεί προτεραιότητα σε λύσεις που έχουν χαμηλότερο συνολικό υψόμετρο, και ταυτόχρονα αυτές που παρουσιάζουν λιγότερα σκέλη με πλάγιο άνεμο. Αυτό μεταφράζεται σε αυξημένο βάρος στους όρους α_6 , α_4 σε σχέση με τους υπόλοιπους στην συνάρτηση.

Η τελική τροχιά που προκύπτει από αυτή την συνάρτηση καταλληλότητας φαίνεται στην επόμενη Εικόνα 21:

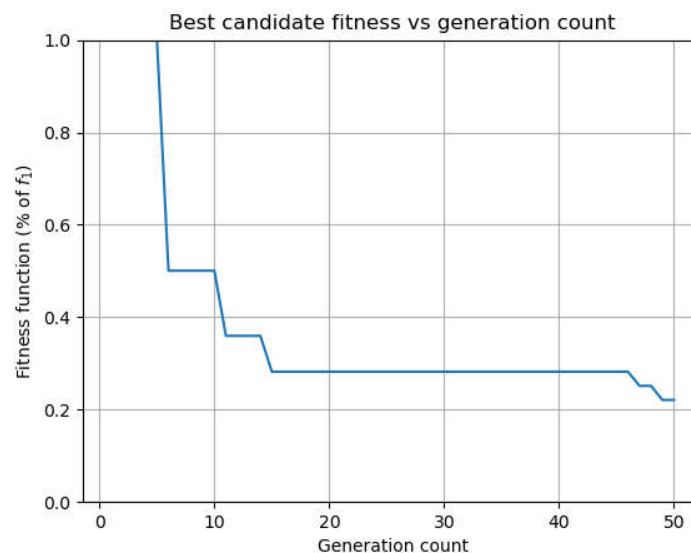


Εικόνα 18 – Βελτιστοποίηση τροχιάς ως προς δύο κριτήρια

Σε αυτή την περίπτωση ο αλγόριθμος έχει επιλέξει μεταξύ των δύο πιθανών λύσεων (η δεύτερη λύση είναι η ίδια τροχιά από την άλλη μεριά του νησιού) αυτή με το μικρότερο συνολικό μήκος, επομένως παρά την σχετική προσαύξηση των συγκεκριμένων συντελεστών ο γενετικός αλγόριθμος εξακολουθεί να επηρεάζεται από τους υπόλοιπους όρους.

Επίσης εμφανής είναι και η ομαλή κατανομή των σημείων καθ' όλο το μήκος της τροχιάς, απόρροια από τις δυνάμεις ελατηρίου που υπάρχουν στην προσομοίωση για την δημιουργία της

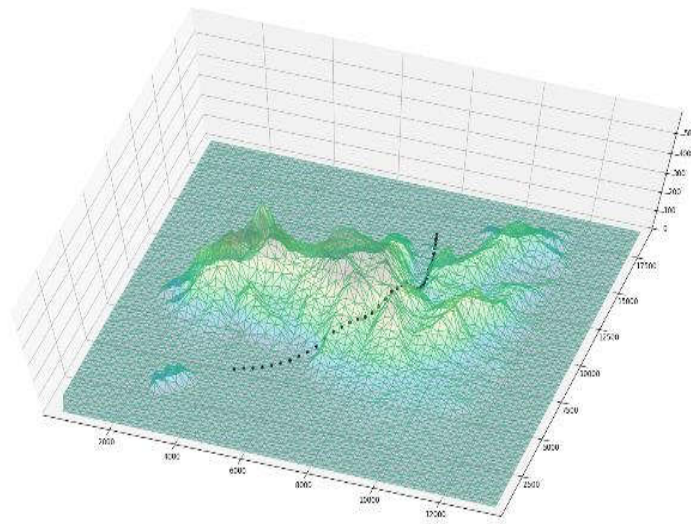
Στην Εικόνα 22 φαίνεται και η αντίστοιχη πορεία της συνάρτησης καταλληλότητας για το καλύτερο χρωμόσωμα κάθε γενιάς:



Εικόνα 19 - Συνάρτηση καταλληλότητας

5.5 Μια γενική λύση

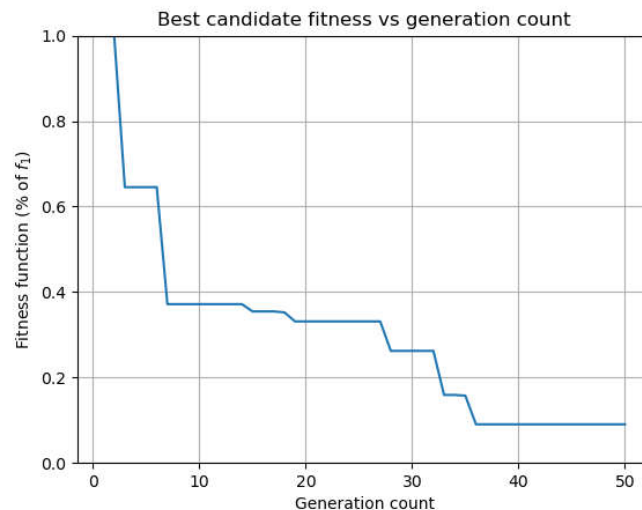
Τέλος, ο αλγόριθμος δοκιμάστηκε χωρίς να δοθεί κάποιο ιδιαίτερο βάρος σε μεμονωμένους όρους στην συνάρτηση καταλληλότητας. Αυτό είχε ως αποτέλεσμα τον ισομοιασμό των προτεραιοτήτων για τον αλγόριθμο, και η τελική λύση που δημιουργήθηκε φαίνεται στην Εικόνα 23:



Εικόνα 20 - Γενική λύση

Σε αυτή την λύση δεν φαίνεται να έχει υπερισχύσει κάποιο γνώρισμα, όπως ήταν ευδιάκριτο στις προηγούμενες δοκιμές, και η τροχιά μοιάζει με μια «μέση λύση», όπως ήταν αναμενόμενο.

Αντίστοιχα με τα προηγούμενα, η πορεία της εξέλιξης (βαθμολογία του καλύτερου χρωμοσώματος για κάθε γενιά του πληθυσμού) είναι παρόμοια, όπως φαίνεται στην Εικόνα 24:



Εικόνα 21 - Συνάρτηση καταλληλότητας

5.6 Σύνοψη

Στο κεφάλαιο αυτό παρουσιάστηκαν τα αποτελέσματα από την χρήση του συνδυασμού των δύο αλγορίθμων, σε ένα ρεαλιστικό πρόβλημα σχεδιασμού τροχιάς ενός ΜΕΑ. Εκ πρώτης όψεως φαίνεται η επιτυχής λύση του προβλήματος με βάση την μεθοδολογία που έχει αναπτυχθεί σε αυτή την διατριβή, καθώς και η δυνατότητα παραγωγής ποικιλόμορφων λύσεων, ανάλογα με τις ρυθμίσεις του αλγορίθμου.

Κεφάλαιο 6 – Επέκταση σε πολλαπλά οχήματα (AIROUS)

6.1 Εισαγωγή

Το project AIROUS είναι ένα εγχείρημα που έχει στόχο να παρέχει την δυνατότητα σχεδιασμού και εκτέλεσης αυτόνομων αποστολών μη επανδρωμένων εναέριων οχημάτων, οι οποίες είναι προσανατολισμένες σε μεταφορά εμπορευμάτων από και προς πλοία. Για την πραγματοποίηση των ανωτέρω, χρειάζεται ένας τρόπος σχεδιασμού των αποστολών για τα UAV ο οποίος να μπορεί να ικανοποιήσει τα απαραίτητα κριτήρια για την συγκεκριμένη περίπτωση. Για τον λόγο αυτό, τελικά επελέγη η μέθοδος που έχει παρουσιαστεί εδώ προς χρήση με τις απαραίτητες τροποποιήσεις και επεκτάσεις.

Οι ουσιαστικές διαφοροποιήσεις σε σχέση με τα μέχρι τώρα παρουσιασμένα σημεία είναι οι εξής:

- Επέκταση σε ταυτόχρονη επίλυση του προβλήματος για πολλαπλά οχήματα
- Δυνατότητα προσθήκης ενδιάμεσων στάσεων σε μια τροχιά
- Εισαγωγή επιπλέον περιορισμών στην τροχιά
 - Αποφυγή εμποδίων στην επιφάνεια (αγκυροβολημένα πλοία)
 - Αποφυγή no-fly zones (χωρίς κάποιο φυσικό εμπόδιο)
- Γενική ανακατασκευή του αλγορίθμου, με γνώμονα την απόδοση σε embedded hardware (αλλαγές προγραμματιστικής φύσεως, χωρίς διαφοροποίηση στην λογική των αλγορίθμων)

Στο κεφάλαιο αυτό θα αναλυθούν οι διαφοροποιήσεις, καθώς και θα παρουσιαστούν τα σχετικά αποτελέσματα από τις νέες δυνατότητες.

6.2 Πολλαπλά οχήματα

Για την λειτουργία της ταυτόχρονης επίλυση του προβλήματος με πολλαπλά οχήματα, ο γενικός αλγόριθμος δημιουργεί τις τροχιές των επιμέρους οχημάτων με την σειρά. Έτσι, κατά την δημιουργία των μετέπειτα της πρώτης τροχιών, κάθε νέα τροχιά μπορεί να είναι biased προς την αποφυγή των προηγούμενων. Αυτό δημιουργεί μια ραχοκοκαλιά στην αντιμετώπιση του προβλήματος της αποφυγής συγκρούσεων μεταξύ των οχημάτων,

οδηγώντας στον χωρικό αποκλεισμό των τροχιών, με παρόμοιο τρόπο όπως και το ανάγλυφο του χάρτη απωθεί τις τροχιές αποτρέποντας τις συγκρούσεις.

Εδώ θα πρέπει να τονιστεί ότι η προσέγγιση αυτή είναι ίσως υπερβολική, καθώς για να υπάρξει σύγκρουση μεταξύ δύο οχημάτων θα πρέπει να βρίσκονται στο ίδιο μέρος *την ίδια στιγμή*. Επομένως, ακόμη και τροχιές οι οποίες συγκλίνουν χωρικά θα μπορούσαν να είναι ασφαλείς εάν υπάρχει χρονική απόσταση μεταξύ των οχημάτων στα σημεία σύγκλισης. Αυτό όμως δεν είναι κάτι που μπορεί να υπολογιστεί από τον αλγόριθμο στην τρέχουσα μορφή του, και επομένως προκειμένου να είναι εγγυημένη η ασφάλεια των τροχιών έχει προτιμηθεί η χωρική απομόνωση τους, η οποία εξ ορισμού εγγυάται την αποφυγή συγκρούσεων

Επιπλέον του μηχανισμού που αναφέρθηκε ήδη, έχει επεκταθεί και η συνάρτηση καταλληλότητας στο κομμάτι του γενετικού αλγορίθμου για να μπορούν να αξιολογηθούν και να συγκριθούν ως προς την ποιότητα οι λύσεις που πλέον αφορούν πολλαπλά οχήματα.

Δεδομένης της γενικής μορφής της συνάρτησης καταλληλότητας για ένα όχημα,

$$F(s) = \sum_{i=1}^K a_i f_i(s)$$

η νέα συνάρτηση καταλληλότητας στην περίπτωση πολλαπλών οχημάτων τελικά έχει την μορφή

$$G(s) = h(s) + \sum_{j=1}^N F(s_j)$$

όπου $h(s)$ είναι μια συνάρτηση που αξιολογεί την ασφάλεια των τροχιών μεταξύ τους με βάση την ελάχιστη απόσταση τους, και $F(s)$ είναι η ήδη γνωστή συνάρτηση καταλληλότητας η οποία βαθμολογεί την κάθε επιμέρους τροχιά s_j για ένα όχημα, και υπολογίζεται αθροιστικά για το σύνολο των N οχημάτων.

6.3 Επιπλέον χωρικοί περιορισμοί

Για την κάλυψη της ανάγκης αποφυγής συγκρούσεων με εμπόδια σχετικά με την συγκεκριμένη εφαρμογή (πχ αγκυροβολημένα πλοία, ανεμογεννήτριες, ...) καθώς και την κάλυψη της ανάγκης αποφυγής πτήσεων σε γενικότερα ελεγχόμενες περιοχές, κατά το στάδιο της ανάγνωσης των χαρτογραφικών δεδομένων έχει προστεθεί η δυνατότητα προσθήκης παραμορφώσεων στο ανάγλυφο. Οι παραμορφώσεις αυτές δεν αντιστοιχούν σε πραγματικά features, και ουσιαστικά ξεγελάνε τον αλγόριθμο αποφυγής συγκρούσεων ώστε να αποφύγει τον όγκο που περικλείουν.

Για την περίπτωση της αποφυγής αγκυροβολημένων σκαφών, προστίθεται στο ανάγλυφο παραμορφώσεις που έχουν σχήμα ημιελλειψοειδούς εκ περιστροφής, με βάση την σχέση

$$\frac{x^2}{a} + \frac{y^2}{b} + \frac{z^2}{c} = 1$$

Κρατώντας από την σχέση μόνο τις τιμές $z \geq 0$, κεντράροντας το ελλειψοειδές σε όποιο επιθυμητό (x_0, y_0) , και μεταβάλλοντας κατάλληλα τις τιμές a, b, c μπορούμε να ρυθμίσουμε τις διαστάσεις των ημιαξόνων του ελλειψοειδούς καθώς και τον προσανατολισμό τους.

Με αυτή την τακτική μπορούμε να δημιουργήσουμε ημιελλειψοειδή εμπόδια όποιων διαστάσεων χρειάζεται τα οποία μπορούν να αποκλείσουν από μικρές περιοχές και όγκους (όπως για παράδειγμα ένα μεμονωμένο αγκυροβολημένο πλοίο), μέχρι τεράστιες εκτάσεις όπου απαγορεύονται ακόμα και οι υπερπτήσεις σε μεγάλο υψόμετρο.

6.4 Λοιπές μεταβολές

Για την προσθήκη ενδιάμεσων στάσεων, εκμεταλλευόμενοι την δυνατότητα πρόσθεσης ενδιάμεσων σημείων για να λειτουργήσει ο αλγόριθμος η προσέγγιση που ακολουθήθηκε ήταν να προστίθενται σημεία τα οποία έχουν δυσανάλογα μεγάλη αδράνεια σε σχέση με τα υπόλοιπα, καθιστώντας τα ακλόνητα.

Επίσης, γενικές μεταβολές πραγματοποιήθηκαν οι οποίες δεν αφορούν τα λογικά βήματα τα οποία ακολουθεί ο αλγόριθμος για την επίλυση του προβλήματος αλλά ήταν απαραίτητες για την βελτίωση του υπολογιστικού χρόνου.

Ενδεικτικά, αναφέρονται οι ακόλουθες μεταβολές:

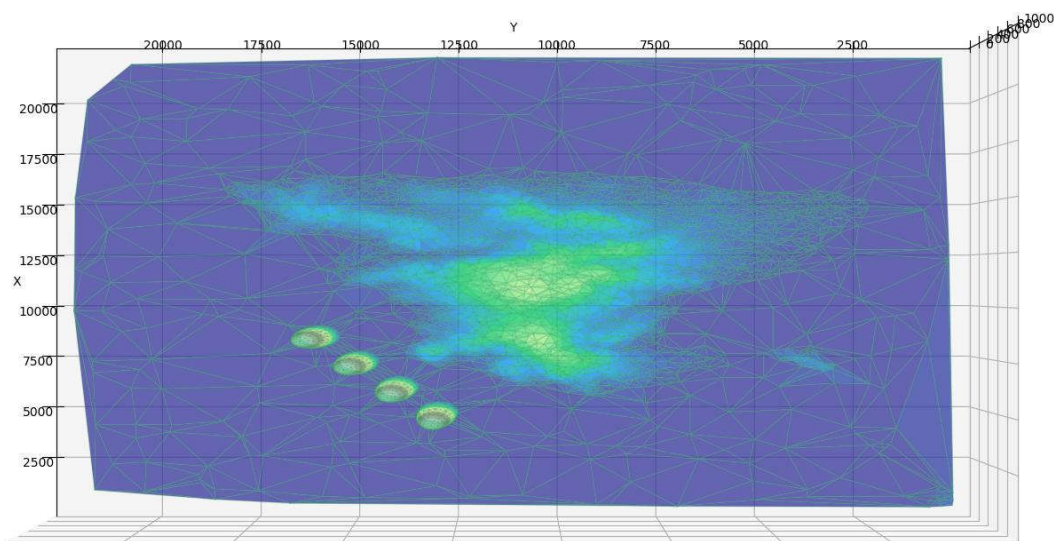
- Προ-υπολογισμός και αποθήκευση αποτελεσμάτων, προς αποφυγή άσκοπων επαναυπολογισμών (όπου ήταν εφικτό)
- Συμπύκνωση των περιεχομένων των δομών επανάληψης, προς αποφυγή πολλαπλών δομών
- Δειγματοληπτική χρήση τοπογραφικών δεδομένων σε περιοχές του χάρτη όπου το ανάγλυφο δεν μεταβάλλεται αρκετά (λ.χ. πεδιάδα ή επιφάνεια της θάλασσας), η οποία περιορίζεται μόνο σε μεγάλες περιοχές με σταθερό υψόμετρο

Τέλος, κάποιες λοιπές αλλαγές πραγματοποιήθηκαν οι οποίες όμως δεν αφορούν την λειτουργία του αλγορίθμου και τα βήματα επίλυσης αλλά αποσκοπούσαν στην ευκολότερη σύνδεση με το ευρύτερο λογισμικό που χρησιμοποιούταν από το project AIRIOUS και αφορούσαν κυρίως θέματα διασύνδεσης με τους ελεγκτές και την σουίτα λογισμικού σχεδιασμού τροχιάς/αποστολής.

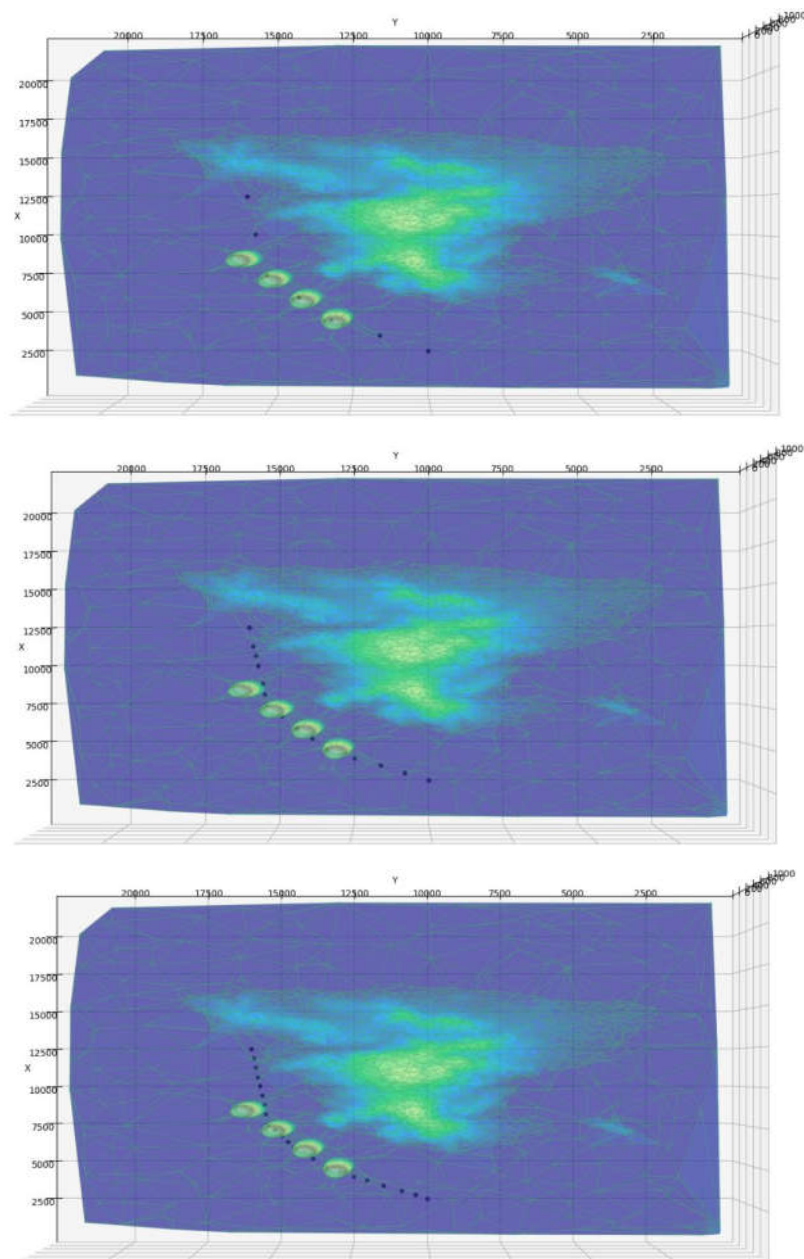
6.5 Ενδεικτικά αποτελέσματα

6.5.1 Ad-hoc εμπόδια

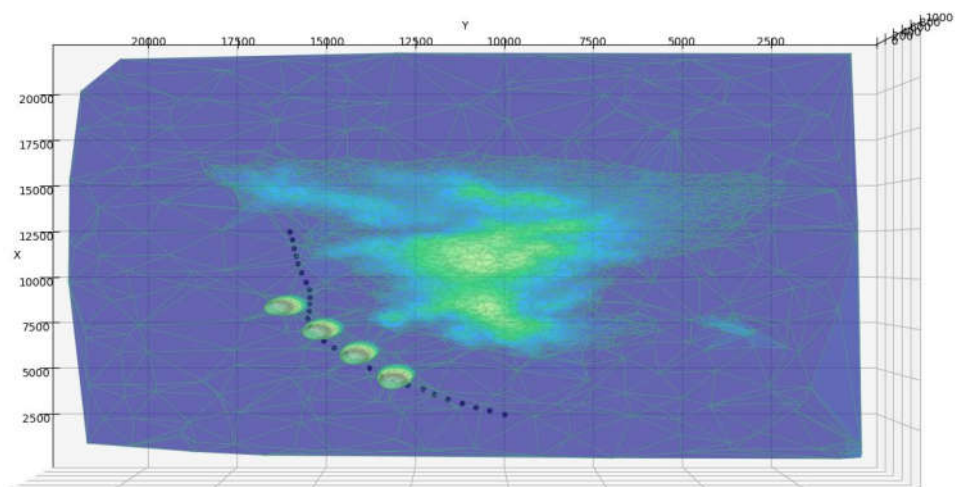
Αρχικά εξετάστηκε η προσθήκη εμποδίων προς αποφυγή με την διαδικασία που αναφέρθηκε παραπάνω, καθώς και η δυνατότητα του αλγορίθμου να τα αποφεύγει κατά την δημιουργία της τροχιάς. Ως πρώτη ανάγκη παρουσιάστηκε η αποφυγή εμποδίων που βρίσκονται στην επιφάνεια της θάλασσας (λ.χ. αγκυροβολημένα πλοία), επομένως τα εμπόδια τοποθετήθηκαν σε σημεία (x,y) που αντιστοιχούν σε θαλάσσια περιοχή. Συνήθως σε τέτοια σημεία απαντώνται παραπάνω από ένα πλοία (ειδικά κοντά σε λιμάνια), και παρακάτω στις Εικόνες 25-28 φαίνονται στιγμιότυπα από μια τέτοια περίπτωση, όπου έχουν προστεθεί τέσσερα εμπόδια (με υπερβολικές διαστάσεις προκειμένου να είναι ευδιάκριτα στις απεικονίσεις) και τα οποία έχουν τοποθετηθεί σκόπιμα μέσα στην διαδρομή της τροχιάς που θα ζητηθεί από τον αλγόριθμο να σχεδιάσει ώστε να εξεταστεί η δυνατότητα αποφυγής αυτών.



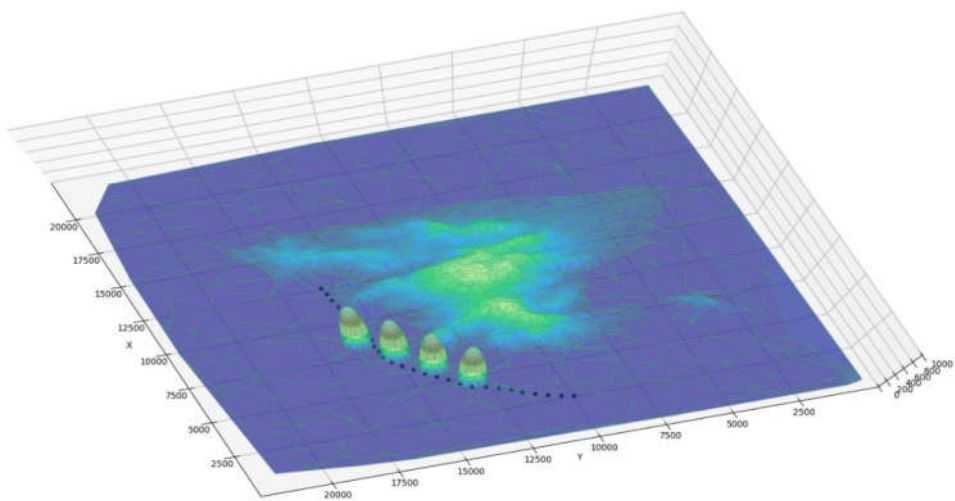
Εικόνα 22 - Ο χάρτης με τα επιπλέον εμπόδια



Εικόνα 23 - Στιγμιότυπα του αλγορίθμου δημιουργίας τροχιών, παρουσία εμποδίων



Εικόνα 24 - Η τελική λύση

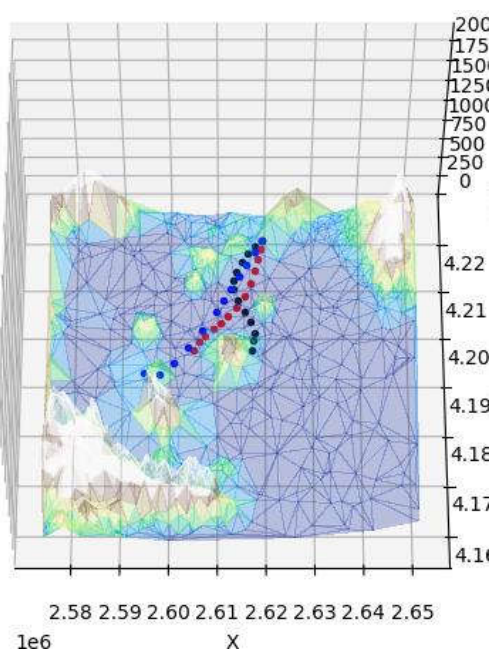


Εικόνα 25 - Η τελική λύση από άλλη οπτική γωνία

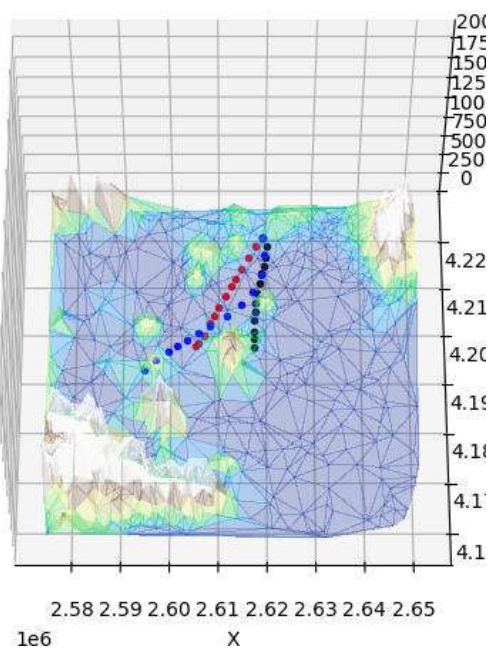
6.5.2 Σχεδιασμός αποστολής με πολλαπλά οχήματα

Για να δοκιμαστεί ο αρχικός στόχος της δημιουργίας τροχιάς για πολλαπλά οχήματα χρησιμοποιήθηκε το ίδιο περιβάλλον με τις αρχικές δοκιμές, διευρύνοντας το παράθυρο του διαθέσιμου χάρτη ώστε να περικλείει το μεγαλύτερο τμήμα του βόρειου Αργοναυτικού. Αφού προστέθηκαν κάποια εμπόδια στον χάρτη, ζητήθηκε από τον αλγόριθμο ο σχεδιασμός τροχιάς για τρία οχήματα, με κοινό σημείο αφετηρίας και διαφορετικούς τελικούς προορισμούς. Στιγμιότυπα από διάφορες γενιές, καθώς και η τελική λύση φαίνονται στις Εικόνες 29-32.

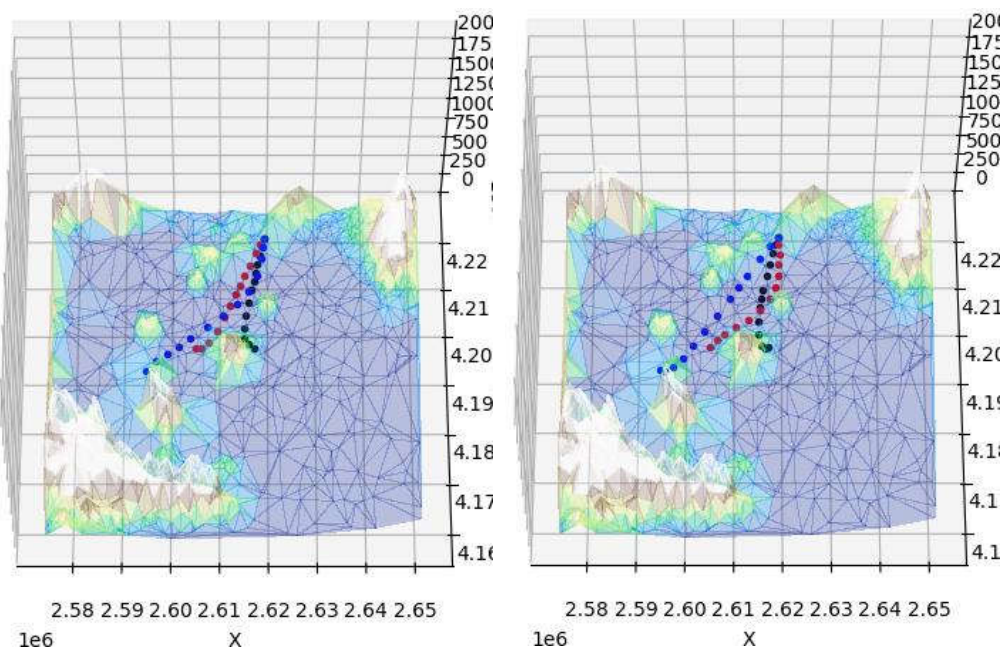
Η χρήση κοινού σημείου αφετηρίας και διαφορετικών τελικών στόχων στις δοκιμές αυτές προσομοιώνει και την εκτιμώμενη πραγματική περίπτωση χρήσης, όπου τα οχήματα υπολογίζεται πως θα ξεκινούν από κάποια κοινή βάση προς τους διάφορους προορισμούς.



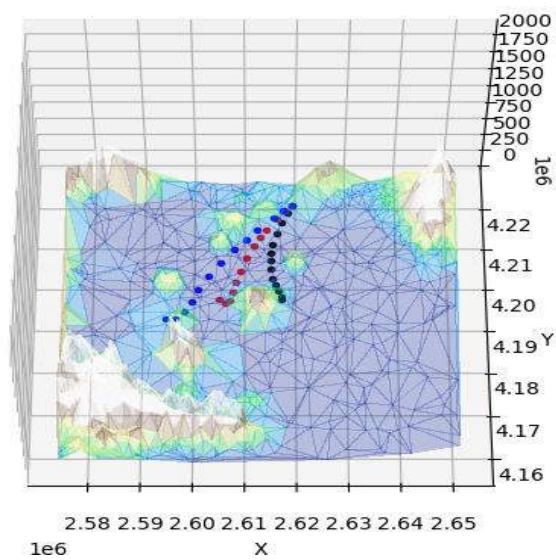
Εικόνα 26 - Η πρώτη γενιά



Εικόνα 27 - Ενδιάμεση γενιά

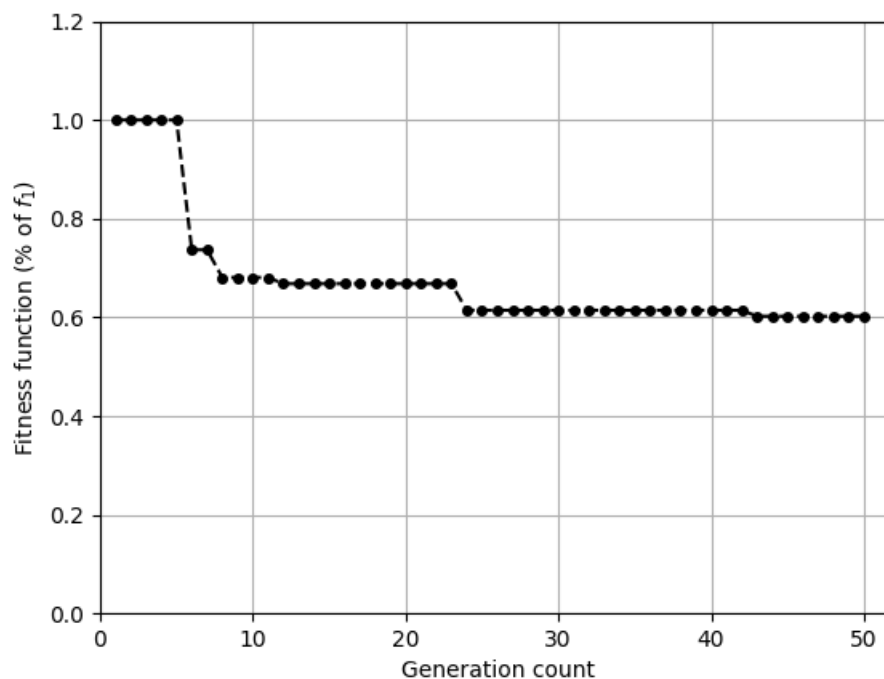


Εικόνα 28 - Περισσότερες ενδιάμεσες γενιές



Εικόνα 29 - Η τελική γενιά του γενετικού αλγορίθμου

Παρατηρούμε όπως και στην περίπτωση του ενός οχήματος πως οι αρχικές γενιές και οι λύσεις στις οποίες οδηγούν δεν είναι ικανοποιητικές, με βάση πάντα τα κριτήρια που έχουμε θέσει, αλλά με την πάροδο του χρόνου ο αλγόριθμος μέσω των βημάτων της διασταύρωσης και της μετάλλαξης καταλήγει σε πολύ διαφορετικές λύσεις οι οποίες πληρούν τις προϋποθέσεις που έχουν ζητηθεί. Αυτό αποτυπώνεται και στην συνάρτηση καταλληλότητας (Εικόνα 33) η οποία φαίνεται πως έχει ολοένα και μειούμενη τιμή, εμφανίζοντας ωστόσο και πάλι τα αναμενόμενα σκαλοπάτια στην γραφική παράσταση που αντιστοιχούν σε στασιμότητα της εξέλιξης από γενιά σε γενιά.



Εικόνα 30 - Η συνάρτηση καταλληλότητας

6.6 Σύνοψη

Στο κεφάλαιο αυτό παρουσιάστηκε η ανάγκη επέκτασης του αλγορίθμου προκειμένου να υλοποιηθεί σε μια ρεαλιστική εφαρμογή (AIROUS), καθώς και μια σύνοψη των βημάτων που ακολουθήθηκαν προς αυτό τον σκοπό. Εν κατακλείδι, παρουσιάστηκαν ενδεικτικά αποτελέσματα τα οποία δείχνουν ότι η προσέγγιση αυτή είναι λειτουργική, ξεκινώντας από δοκιμές των βασικών δυνατοτήτων καθώς ικανοποιεί τις απαιτήσεις και τους στόχους που είχαν τεθεί αρχικά.

Ως σενάριο για τις δοκιμές επιλέχθηκε ένα με βάση ρεαλιστικά κριτήρια το οποίο θα μπορούσε να αποτελεί μια πραγματική αποστολή για το οποίο κατευθύνσεις οι οποίες δόθηκαν από τον οργανισμό που έχει αναλάβει το project AIROUS, και τα αποτελέσματα αξιολογήθηκαν από τον ίδιο φορέα.

Κεφάλαιο 7^ο - Επίλογος

7.1 Σύνοψη αποτελεσμάτων

Με βάση τα παραπάνω αποτελέσματα καθώς και πολλές επιπλέον δοκιμές που δεν παρουσιάζονται σε αυτή την διατριβή για οικονομία χώρου, φαίνεται ότι ο αρχικός στόχος που είναι η επίλυση του προβλήματος του σχεδιασμού τροχιάς έχει επιτευχθεί. Ο βασικός αλγόριθμος δημιουργίας τροχιάς έχει αποδείξει πως μπορεί να δημιουργήσει μια τροχιά η οποία θα είναι ασφαλής για το σκάφος, τόσο αποφεύγοντας τις συγκρούσεις όσο ικανοποιώντας το φάκελο πτήσης του οχήματος, ενώ η γενική λύση με τον γενετικό αλγόριθμο φαίνεται να ικανοποιεί τον στόχο της παραγωγής βελτιστοποιημένων αποτελεσμάτων.

Πιο συγκεκριμένα, η επιλογή ενός γενετικού αλγορίθμου προκειμένου να εκμεταλλευτούμε την δυνατότητα βαθμονόμησης και μελέτης ως προς την απόδοση μεμονωμένων κριτηρίων αποδεικνύεται λειτουργική, ενώ η τροποποίηση των συντελεστών βαρύτητας στους επιμέρους όρους της συνάρτησης καταλληλότητας φαίνεται ότι είναι αρκετή και ικανή για να ωθήσει προς λύσεις με συγκεκριμένα χαρακτηριστικά. Όπως ήδη φάνηκε στην πράξη, η συνάρτηση βαθμονόμησης χρειάστηκε τελικά να επεκταθεί και η κίνηση αυτή απέδειξε την δυνατότητα πρόσθεσης επιπλέον όρων οι οποίοι θα αξιολογούν την απόδοση σε κάποιον άλλο τομέα πέρα από τα σημεία που έχουν ήδη οριστεί.

Όπως επίσης φάνηκε στην πράξη, ο αλγόριθμος δεν είναι σε καμία περίπτωση βελτιστοποιημένος υπολογιστικά καθώς ήδη έχει πραγματοποιηθεί μια ανακατασκευή με την οποία βελτιώθηκε κατά πολύ ο χρόνος απόκρισης αλλά σίγουρα θα επιδέχεται περεταίρω βελτιώσεων ειδικά σε αυτό το κομμάτι.

Κάτι το οποίο μπορεί σίγουρα να βελτιωθεί περεταίρω είναι το πρόβλημα με τη στασιμότητα μεταξύ των γενεών κατά τις επαναλήψεις του γενετικού αλγορίθμου, το οποίο είναι και αναμενόμενο στην χρήση τέτοιων γενετικών αλγορίθμων. Σε αυτές τις περιπτώσεις ο αλγόριθμος εγκλωβίζεται σε τοπικά ελάχιστα, και ο μόνος τρόπος να ξεφύγει από αυτές τις λύσεις είναι μέσω των τυχαίων μεταλλάξεων στα χρωμοσώματα. Στην ελαχιστοποίηση αυτού του προβλήματος μπορεί να συμβάλλει η επιλογή διαφορετικών προσεγγίσεων στην διαδικασία επιλογής των χρωμοσωμάτων για την διασταύρωση που γίνεται στο τέλος κάθε επανάληψης, ή ακόμα και ο τρόπος και η πιθανότητα μετάλλαξης των χρωμοσωμάτων. Σε κάθε περίπτωση η πραγματικά βέλτιστη λύση του αρχικού προβλήματος του σχεδιασμού τροχιάς θα παραμένει άγνωστη, αλλά

μειώνοντας την στασιμότητα θα μπορούσαν να βρεθούν πιο γρήγορα, έστω και κατ' ελάχιστο, καλύτερες λύσεις με βάση πάντα τα εκάστοτε τα κριτήρια.

7.2 Μελλοντικές εργασίες

Τα αποτελέσματα που παρουσιάστηκαν σε αυτή τη διατριβή αποτελούν μια πρώτη υλοποίηση των αλγορίθμων που συζητήθηκαν πρώτα κεφάλαια αυτής. Με βάση τις υλοποιήσεις, και ιδιαίτερα την εφαρμογή στα πλαίσια του project AIROUS, έχουν εντοπιστεί αρκετά σημεία τα οποία χρήζουν περαιτέρω διερεύνησης όπως τα εξής:

- Προσθήκη δυνατότητας αποκλεισμού αυστηρά εναέριων ζωνών (και όχι όγκων που ξεκινάνε από την επιφάνεια του ανάγλυφου/της θάλασσας).
- Ενσωμάτωση της δυνατότητας εκτίμησης των τροχιών με βάση κινηματικά μοντέλα για τα οχήματα, με τέτοιο τρόπο ώστε να μπορούν να αξιολογηθούν οι τροχιές ως προς τις απαραίτητες εισόδους προκειμένου να ακολουθηθούν.
- Προσθήκη της έννοιας του χρονοπρογραμματισμού των πτήσεων και συμπερίληψη της διάστασης του χρόνου στην διαδικασία αποφυγής συγκρούσεων μεταξύ οχημάτων σε περιπτώσεις πολλαπλών τροχιών.
- Προκειμένου να μπορεί να χρησιμοποιηθεί ένα μεγάλο τμήμα χάρτη, με πυκνές μετρήσεις μεγάλης ακρίβειας, θα πρέπει να επεκταθεί η διαδικασία της δειγματοληψίας των δεδομένων με στόχο τη μείωση του απαιτούμενου υπολογιστικού χρόνου.
- Πιθανή αναδιάρθρωση του γενικού τρόπου κλήσης και εκτέλεσης του αλγορίθμου ώστε να μπορεί να χρησιμοποιηθεί σε πραγματικό χρόνο, λύνοντας συνεχόμενα, τοπικά περιορισμένα, μικρού εύρους προβλήματα εύρεσης τροχιάς με βάση την τρέχουσα θέση ενός οχήματος και περιορισμένη εικόνα του περιβάλλοντος από αισθητήρες.

Βιβλιογραφία

- [1] K. S. Platanitis, G. P. Kladis and N. C. Tsourveloudis, "Safe flyable and energy efficient UAV missions via biologically inspired methods," in *International Conference on Unmanned Aircraft Systems (ICUAS)*, Athens, 2020.
- [2] K. S. Platanitis, G. P. Kladis, and N. C. Tsourveloudis, "UAV path planning for offshore swarm-based missions," in *International Conference on Unmanned Aircraft Systems*, Dubrovnik, 2022.
- [3] C. Ntakolia, K. S. Platanitis, G. P. Kladis and A. D. Zagorianos, "A Genetic Algorithm enhanced with Fuzzy-Logic for multi-objective Unmanned Aircraft Vehicle path planning missions," in *International Conference on Unmanned Aircraft Systems*, Dubrovnik, 2022.
- [4] V. Kunchev, L. Jain, V. Ivancevic and R. Finn, "Path Planning and Obstacle Avoidance for Autonomous Mobile Robots: A Review," in *Knowledge-Based Intelligent Information and Engineering Systems*, Bournemouth, 2006.
- [5] C. C, Q. S., B. H. and G. L., "Path planning and obstacle avoidance for AUV: A review," *Ocean Engineering*, vol. 235, 2021.
- [6] J. Minguez, F. Lamiroux and J.-P. Laumond, "Motion Planning and Obstacle Avoidance," *Springer Handbook of Robotics*, pp. 827-852.
- [7] H. Niu, Z. Ji, A. Savvaris and A. Tsourdos, "Energy efficient path planning for Unmanned Surface Vehicle in spatially-temporally variant environment," *Ocean Engineering*, vol. 196, 2020.
- [8] S. K. Debnath, R. Omar and N. B. A. Latip, "A review on energy efficient path planning algorithms for unmanned air vehicles.," *Computational Science and Technology*, pp. 523-532, 2019.
- [9] D. F., Z. Zhang, M. Fu, Y. Wang and C. Wang, "Energy-efficient Path Planning and Control Approach of USV Based on Particle Swarm optimization," in *OCEANS*, Charleston, 2018.
- [10] K. Zakhrarov, A. Saveliev and O. Sivchenko, "Energy-Efficient Path Planning Algorithm on Three-Dimensional Large-Scale Terrain Maps for Mobile Robots," in *International Conference on Interactive Collaborative Robotics*, 2020.

- [11] B. Patle, B. L. P. A. Ganesh, P. D.R.K and A. Jagadeesh, "A review: On path planning strategies or navigation of mobile robot," *Defence Technology*, vol. 15, no. 4, pp. 582-606, 2019.
- [12] X. Yu and Z. Youmin, "Sense and avoid technologies with applications to unmanned aircraft," *Progress in Aerospace Sciences*, vol. 74, pp. 152-166, 2015.
- [13] I. K. Nikolos, E. S. Zografos and A. N. Brintaki, "UAV path planning using evolutionary algorithms.," *Innovations in intelligent machines*, vol. 1, pp. 77-111, 2007.
- [14] S. Mittal and D. Kalyanmoy, "Three-dimensional offline path planning for UAVs using multiobjective evolutionary algorithms.," *IEEE congress on evolutionary computation*, pp. 3195-3202, 2007.
- [15] H. Bharadwaj and K. E. Vinodh, "Comparative study of neural networks in path planning for catering robots," *Procedia Computer Science*, vol. 133, pp. 417-423, 2018.
- [16] L. Dubins, "On Curves of Minimal Length with a Constraint on Average Curvature, and with Prescribed Initial and Terminal Positions and Tangents.," *American Journal of Mathematics*, vol. 79, no. 3, pp. 497-516, 1957.
- [17] I. K. Nikolos, N. C. Tsourveloudis and K. P. Valavanis, "Evolutionary algorithm based off-line path planner for UAV navigation.," *Automatika-Zagreb*, vol. 42, pp. 134-150, 2001.
- [18] I. K. Nikolos et al., "Evolutionary algorithm based offline/online path planner for UAV navigation," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 33, no. 6, pp. 898-912, 2003.
- [19] C. Harmening and H. Neuner, "Choosing the Optimal Number of B-spline Control Points," *Journal of Applied Geodesy*, vol. 10, no. 3, 2016.
- [20] W. Ding, W. Gao, K. Wang and S. Shen, "An Efficient B-Spline-Based Kinodynamic Replanning Framework for Quadrotors," *IEEE Transactions on Robotics*, vol. 35, no. 6, pp. 1287-1306, 2019.
- [21] H. Choset, *Principles of Robot Motion: Theory, Algorithms, and Implementations*, MIT Press, 2005.
- [22] J. Kim and P. Khosla, "Real-Time Obstacle Avoidance Using Harmonic Potential Functions," *IEEE Transactions on Robotics and Automation*, pp. 338-349, 1992.

- [23] M. Guerra, D. Efimov, G. Zheng and W. Perruquetti, "Avoiding Local Minima in the Potential Field Method using Input-to-State Stability," *Control Engineering Practice*, vol. 55, pp. 174-184, 2016.
- [24] P. Vadakkepat, K. Tan and W. Ming-Liang, "Evolutionary artificial potential fields and their application in real time robot path planning," *Proceedings of the 2000 Congress on Evolutionary Computation*, vol. 1, pp. 256-263, 2000.
- [25] R. Palm and A. Bouguerra, "Particle Swarm Optimization of Potential Fields for Obstacle Avoidance," in *RARM*, Istanbul, 2013.
- [26] D. Lee and B. Schachter, "Two Algorithms for Constructing a Delaunay Triangulation," *International Journal of Computer and Information Sciences*, vol. 9, no. 3, 1980.
- [27] P. Kora and P. Yadlapalli, "Crossover Operators in Genetic Algorithms: A Review," *International Journal of Computer Applications*, vol. 162, 2017.
- [28] S. Katoch, S. Chauhan and V. Kumar, "A review on genetic algorithm: past, present, and future," *Multimed Tools Appl*, vol. 80, pp. 8091-8126, 2021.
- [29] A. Hassanat, K. Almohammadi, E. Alkafaween, E. Abunawas, A. Hammouri and V. Prasath, "Choosing Mutation and Crossover Ratios for Genetic Algorithms—A Review with a New Dynamic Approach," *Information*, vol. 10, no. 12, 2019.
- [30] M. Mitchell, *An Introduction to Genetic Algorithms*, MIT Press, 1998.