

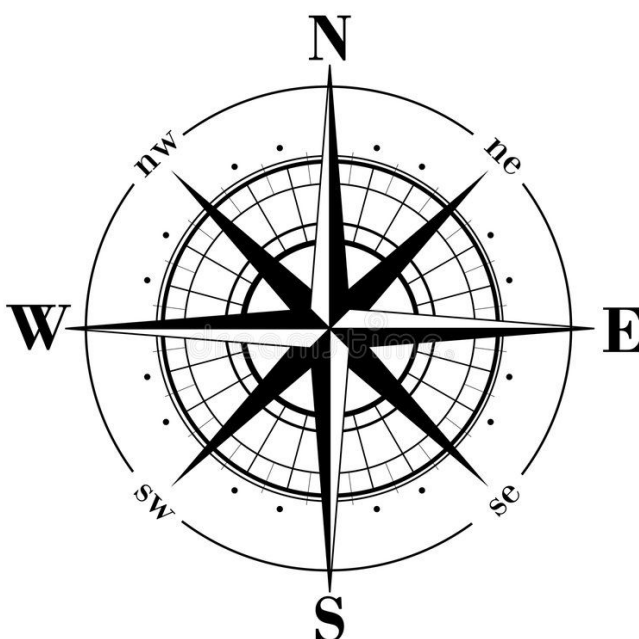


ΧΑΡΟΚΟΠΕΙΟ ΠΑΝΕΠΙΣΤΗΜΙΟ

ΣΧΟΛΗ *Χαροκόπειο Πανεπιστήμιο*
ΤΜΗΜΑ *Πληροφορική και Τηλεματική*
ΠΡΟΓΡΑΜΜΑ ΜΕΤΑΠΤΥΧΙΑΚΩΝ ΣΠΟΥΔΩΝ
Πληροφορική και Τηλεματική
ΚΑΤΕΥΘΥΝΣΗ *Τεχνολογίες και Εφαρμογές Ιστού*

Σύστημα συστάσεων για τη συντομότερη πλεύση ιστιοφόρων
Διπλωματική Εργασία

Χρήστος Πιτσικάλης



Αθήνα, 2017



ΧΑΡΟΚΟΠΕΙΟ ΠΑΝΕΠΙΣΤΗΜΙΟ

ΣΧΟΛΗ *Χαροκόπειο Πανεπιστήμιο*
ΤΜΗΜΑ *Πληροφορική και Τηλεματική*
ΠΡΟΓΡΑΜΜΑ ΜΕΤΑΠΤΥΧΙΑΚΩΝ ΣΠΟΥΔΩΝ
Πληροφορική και Τηλεματική
ΚΑΤΕΥΘΥΝΣΗ *Τεχνολογίες και Εφαρμογές Ιστού*

Τριμελής Εξεταστική Επιτροπή

Κωνσταντίνος Τσερπές
Επίκουρος Καθηγητής, Πληροφορικής και Τηλεματικής,
Χαροκόπειο

Μάρα Νικολαΐδου
Καθηγήτρια, Πληροφορικής και Τηλεματικής, Χαροκόπειο

Δημήτριος Μιχαήλ
Επίκουρος Καθηγητής, Πληροφορικής και Τηλεματικής,
Χαροκόπειο

Ο/Η Χρήστος Πιτσικάλης

δηλώνω υπεύθυνα ότι:

- 1) Είμαι ο κάτοχος των πνευματικών δικαιωμάτων της πρωτότυπης αυτής εργασίας και από όσο γνωρίζω η εργασία μου δε συκοφαντεί πρόσωπα, ούτε προσβάλλει τα πνευματικά δικαιώματα τρίτων.
- 2) Αποδέχομαι ότι η ΒΚΠ μπορεί, χωρίς να αλλάξει το περιεχόμενο της εργασίας μου, να τη διαθέσει σε ηλεκτρονική μορφή μέσα από τη ψηφιακή Βιβλιοθήκη της, να την αντιγράψει σε οποιοδήποτε μέσο ή/και σε οποιοδήποτε μορφότυπο καθώς και να κρατά περισσότερα από ένα αντίγραφα για λόγους συντήρησης και ασφάλειας.

ΕΥΧΑΡΙΣΤΙΕΣ

Αρχικά θα ήθελα να ευχαριστήσω όλους όσους συνέβαλαν στην ολοκλήρωση της διπλωματικής εργασίας μου και γενικότερα των σπουδών μου στο Τμήμα Πληροφορικής και Τηλεματικής του Χαροκόπειου Πανεπιστημίου.

Θα ήθελα να ευχαριστήσω τον επιβλέποντα καθηγητή της πτυχιακής μου εργασίας κ. Κωνσταντίνο Τσερπέ, για την καθοδήγησή του και την πολύτιμη συμβολή του σε κάθε φάση της δημιουργίας της.

Τέλος, θα ήθελα να ευχαριστήσω θερμά την οικογένειά μου για την ηθική και οικονομική συμπαράσταση όχι μόνο κατά τη διάρκεια της εκπόνησης της πτυχιακής μου εργασίας αλλά και καθ' όλη τη διάρκεια των σπουδών μου.

ΠΙΝΑΚΑΣ ΠΕΡΙΕΧΟΜΕΝΩΝ

1. ΕΙΣΑΓΩΓΗ	10
1.1 Πλεύσεις ιστιοφόρου σκάφους	10
1.2 Φαινόμενος άνεμος	11
1.3 Περιγραφή του προβλήματος	12
1.4 Πρόκληση	13
1.5 Σύνοψη Κεφαλαίων	13
2. ΥΠΟΒΑΘΡΟ	14
2.1 Σχετικές δουλειές/εργασίες	14
2.1.1 Υπολογισμός ταχύτητας	14
2.1.2 Συμφέρουσα πορεία	15
2.1.3 Προγράμματα βέλτιστης διαδρομής(Weather Routing)	17
2.1.4 Βιβλιογραφικό κενό	17
2.2 Εργαλεία	18
2.2.1 OpenWeatherMap API	18
2.2.2 Google Maps API	18
2.2.3 Hipster4j	19
2.2.4 GeoTools και ShapeFile	19
3. ΥΛΟΠΟΙΗΣΗ	20
3.1 Σχεδιασμός	20
3.1.1 Back-End	20
3.1.2 Front-End	21
3.2 Υλοποίηση	22
3.2.1 Εντοπισμός σημείων και διαθέσιμων διαδρομών	22
3.2.2 Ο Καιρός	24
3.2.3 Ο Γράφος	25
3.3 Επικοινωνία	26
3.3.1 validatePoint	26
3.3.2 sail	27
3.4 Απεικόνιση	28
4. ΑΠΟΤΕΛΕΣΜΑΤΑ	28
4.1 Παρουσίαση της εφαρμογής	28
5. ΣΥΜΠΕΡΑΣΜΑΤΑ	32
5.1 Τελικός Απολογισμός	32
5.2 Βελτιώσεις	33
6. ΒΙΒΛΙΟΓΡΑΦΙΑ	35

Περίληψη

Σκοπός της παρούσας εργασίας είναι η κατασκευή ενός εργαλείου το οποίο θα υποδεικνύει την συντομότερη πλεύση για ιστιοφόρα σκάφη. Για την υλοποίηση του εργαλείου αυτού υπολογίστηκαν σημαντικοί παράγοντες της κίνησης των ιστιοφόρων. Ιδιαίτερη έμφαση δόθηκε στις διαρκώς μεταβαλλόμενες παραμέτρους, όπως η κατεύθυνση και η ένταση του ανέμου. Το σύστημα που προκύπτει, κάνει χρήση των εκάστοτε καιρικών δεδομένων και βάσει αυτών υπολογίζει την πιθανή ταχύτητα του σκάφους στο εκάστοτε σημείο, προτείνοντας τελικά τη βέλτιστη διαδρομή από ένα καθορισμένο σημείο σε ένα άλλο. Τελικό αποτέλεσμα της εφαρμογής αυτής, είναι η γραφική απεικόνιση της βέλτιστης πορείας του σκάφους στο χάρτη, που αποτελεί χρήσιμο συμβουλευτικό εργαλείο για οποιοδήποτε ταξίδι με ιστιοπλοϊκό και ιδιαίτερα για τους ιστιοπλοϊκούς αγώνες.

Λέξεις κλειδιά: όρτσα, τακ, πραγματικός άνεμος, φαινόμενος άνεμος, νεκρός τομέας

Abstract

The purpose of this work is to build a tool that will indicate the shortest path for sailing boats. In order to implement this tool, important factors of the sailing movement were calculated. Particular emphasis was placed on ever-changing parameters, such as the direction and intensity of the wind. The resulting system makes use of the weather data, and based on it, it calculates the possible speed of the boat at each point, ultimately proposing the optimal route from one specified point to another. The final result of this application is the graphic depiction of the optimal course of the vessel on the map, which is a useful advisory tool for any sailing trip and especially for sailing races.

Keywords: close hauled, tack, true wind, apparent wind, dead area

ΚΑΤΑΛΟΓΟΣ ΕΙΚΟΝΩΝ

Εικ.1. Πλεύσεις ιστιοφόρου.....	σ.11
Εικ.2. Φαινόμενος άνεμος.....	σ.12
Εικ.3. Παράδειγμα validatePoint.....	σ.29
Εικ.4. Παράδειγμα πορείας με πολλαπλά ΤΑΚ.....	σ.30
Εικ.5. Κατεύθυνση ίδια με αυτήν του ανέμου.....	σ.31
Εικ.6. Στενό πέρασμα.....	σ.32

ΚΑΤΑΛΟΓΟΣ ΣΧΗΜΑΤΩΝ

Σχ.1: Σχεδιασμός λειτουργίας.....	σ.20
Σχ.2: Επιλογή σημείων.....	σ.22
Σχ.3: Μετρήσεις ανέμου.....	σ.24

1. ΕΙΣΑΓΩΓΗ

1.1 Πλεύσεις ιστιοφόρου σκάφους

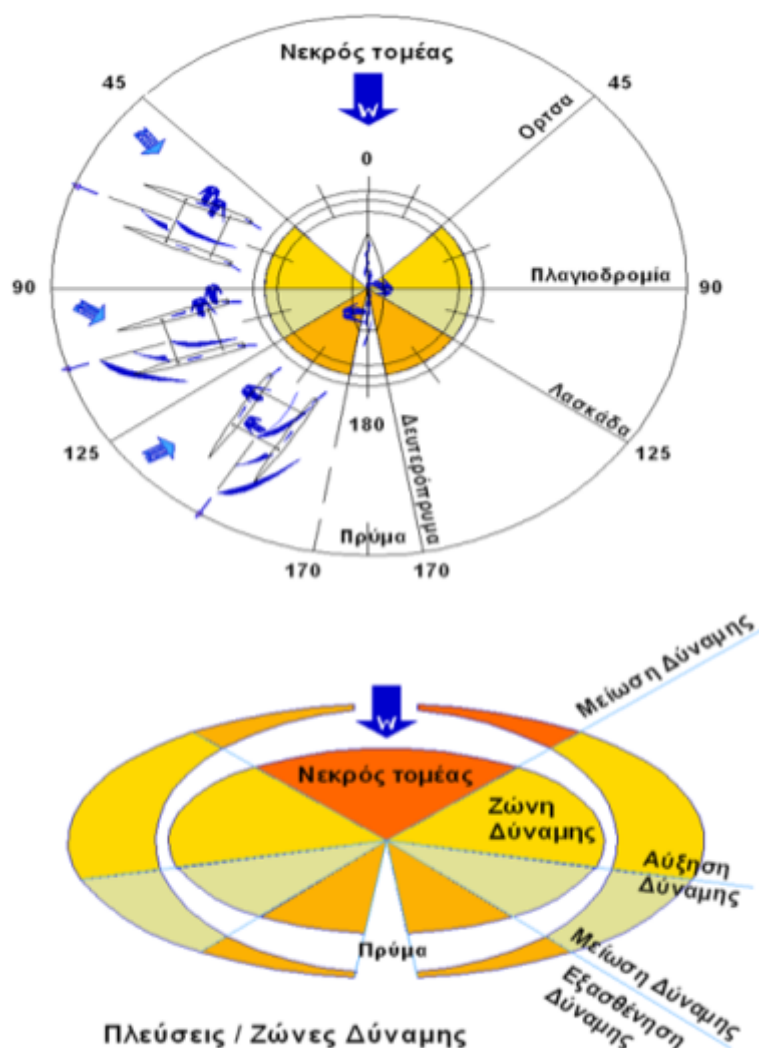
Η κίνηση ενός ιστιοφόρου σκάφους είναι πολύ διαφορετική από αυτήν ενός μηχανοκίνητου. Το ιστιοφόρο έχει ως κύριο μέσο πρόωσης τα πανιά, δηλαδή αξιοποιεί την δύναμη του αέρα, ενώ έχει συνήθως και κινητήρια μονάδα που χρησιμοποιείται ως βοηθητική. Η κίνηση με τα πανιά έχει ένα τεράστιο πλεονέκτημα έναντι του κινητήρα και αυτό είναι η ανεξάντλητη πηγή ενέργειας που είναι ο αέρας. Έχει όμως και κάποια μειονεκτήματα.

Όταν το ιστιοφόρο σκάφος ιστιοπλοεί, εξαρτάται από τον αέρα όχι μόνο η ταχύτητά του αλλά και οι κατευθύνσεις που μπορεί να πάρει (Marchaj, 2000). Ο βασικότερος περιορισμός του είναι ότι δεν μπορεί να ταξιδέψει σε πορεία αντίθετη από αυτήν του ανέμου αλλά σε τουλάχιστον 40 μοίρες από αυτήν (αναλόγως των προδιαγραφών του κάθε σκάφους). Επίσης, η ταχύτητα που μπορεί να αναπτύξει είναι ανάλογη και των δυο συνιστωσών.

Η κοντινότερη προς τον άνεμο πλεύση λέγεται όρτσα και συνήθως είναι στις 40 μοίρες από τον άνεμο (Καλαθάκης, 2017). Όσο ανοίγεται η πορεία, η κίνηση μεταβάλλεται σε ανοιχτά όρτσα και κλειστή πλαγιοδρομία μέχρι πλαγιοδρομία (στις 90 μοίρες) και από εκεί σε ανοιχτή πλαγιοδρομία μέχρι λασκάδα, μετά δευτερόπρυμα και πρύμα ή κατάπρυμα (Καλαθάκης, 2017).

Γρηγορότερη πλεύση θεωρείται η πλαγιοδρομία, όταν δηλαδή ο πραγματικός άνεμος είναι κάθετος προς την πορεία του σκάφους (Garrett, 1996), ενώ στα όρτσα είναι μεγαλύτερη η αίσθηση της ταχύτητας λόγω του φαινόμενου ανέμου (όπως θα αναλυθεί παρακάτω). Στα πρύμα, ενώ κάποιος θα περίμενε ότι η ταχύτητα θα είναι η μεγαλύτερη, είναι τελικά συνήθως η μικρότερη και αυτό λόγω της φύσης των ιστιοφόρων, επειδή δεν είναι σχεδιασμένα για να τα σπρώχνει ο άνεμος αλλά να χρησιμοποιούν κανόνες αεροδυναμικής με βάση το θεώρημα Bernoulli (Clancy, 1975).

Το παρακάτω σχήμα εξηγεί τις πλεύσεις και τις αντίστοιχες δυνάμεις από τον αέρα σε κάθε μία από αυτές:



Εικ. 1. Πλεύσεις ιστιοφόρου

Η μεταβολή της πορείας από όρτσα σε όρτσα διασχίζοντας τον νεκρό τομέα λέγεται ΤΑΚ ή αναστροφή ενώ η αντίστοιχη μεταβολή στα δευτερόπρυμα ονομάζεται πόντζα ή υποστροφή (Καλαθάκης, 2016).

1.2 Φαινόμενος άνεμος

Ο άνεμος τον οποίο αντιλαμβάνεται ένας ακίνητος παρατηρητής ή ένα ακίνητο ανεμόμετρο λέγεται πραγματικός άνεμος γιατί τα χαρακτηριστικά του δεν διαφέρουν από αυτά του ανέμου ο οποίος φυσά την στιγμή της παρατήρησης. Αν

όμως ο παρατηρητής ή το ανεμόμετρο κινούνται, είναι φυσικό η κίνησή τους να επηρεάζει τα χαρακτηριστικά του ανέμου που γίνονται αντιληπτά (Marchaj, 2000). Τα χαρακτηριστικά του ανέμου που αντιλαμβάνεται ο κινούμενος παρατηρητής είναι συνδυασμός της κίνησής του και των χαρακτηριστικών του πραγματικού ανέμου. Έτσι ο άνεμος που αντιλαμβάνεται λέγεται φαινόμενος άνεμος.

Ο φαινόμενος άνεμος διαφέρει από τον πραγματικό σε ένταση και διεύθυνση (Marchaj, 2000). Αν η κίνηση του παρατηρητή είναι αντίθετης κατεύθυνσης από αυτή του πραγματικού ανέμου, η ένταση του φαινόμενου ανέμου είναι μεγαλύτερη, ενώ όταν είναι μαζί με τον άνεμο, η ένταση του φαινόμενου ανέμου είναι μικρότερη.



Εικ. 2. Φαινόμενος άνεμος

1.3 Περιγραφή του προβλήματος

Το πρόβλημα λοιπόν είναι ότι η γρηγορότερη διαδρομή μεταξύ δύο σημείων δεν είναι πάντα η ίδια και το βασικότερο, δεν είναι αυτή που φαίνεται. Εφόσον κάποιες πορείες θα είναι απαγορευτικές, όπως η πορεία με γωνία προς τον άνεμο < 40 μοιρών, αυτομάτως θα πρέπει να επιλεγθεί διαφορετικός δρόμος ή πολλαπλά τακ (Καλαθάκης, 2016).

Το ερώτημα που προκύπτει, είναι τι τελικά συμφέρει περισσότερο από θέμα χρόνου. Μία επιλογή είναι να κάνει το σκάφος συνεχόμενα τακ ώστε να παραμένει στην πορεία του έχοντας όμως μικρότερη ταχύτητα. Ειδάλλως, θα πρέπει να επιλέξει διαφορετική διαδρομή. Το εν λόγω δίλημμα ενδεχομένως να αποτελεί λεπτομέρεια

όταν πρόκειται για ταξίδι αναψυχής. Ωστόσο, μπορεί να φέρει καλύτερα αποτελέσματα όταν πρόκειται για ιστιοπλοϊκό αγώνα.

1.4 Πρόκληση

Η πρόκληση λοιπόν που προκύπτει, είναι να δημιουργηθεί μια εφαρμογή η οποία θα μπορεί να εντοπίζει τη βέλτιστη διαδρομή και να δίνει στον χρήστη τις συντεταγμένες που θα πρέπει να ακολουθήσει ώστε να την επιτύχει. Αποτελεί πρόκληση, διότι ο άνεμος αλλάζει συνεχώς και αντίστοιχα τα γρήγορα κομμάτια δεν παραμένουν ίδια. Θα πρέπει, δηλαδή, η εφαρμογή να λαμβάνει υπόψιν πάντα τον άνεμο της εκάστοτε στιγμής και περιοχής, ώστε να προτείνει την πιο συμφέρουσα επιλογή.

1.5 Σύνοψη Κεφαλαίων

Στο **πρώτο κεφάλαιο** παρουσιάστηκαν κάποια εισαγωγικά στοιχεία για την ιστιοπλοΐα, ώστε να γίνει κατανοητή η λειτουργία του ιστιοφόρου σκάφους, καθώς και η πρόκληση την οποία καλείται να λύσει η εργασία αυτή.

Στο **δεύτερο κεφάλαιο** παρουσιάζονται οι μαθηματικοί τύποι και τα εργαλεία που χρησιμοποιήθηκαν για την επίλυση του εν λόγω προβλήματος. Επίσης, γίνεται αναφορά σε παρόμοια προγράμματα που εντοπίστηκαν κατά τη βιβλιογραφική ανασκόπηση.

Στο **τρίτο κεφάλαιο** φαίνεται ο σχεδιασμός του συστήματος καθώς και κάποιες λεπτομέρειες της υλοποίησής του.

Στο **τέταρτο κεφάλαιο** παρουσιάζεται η λειτουργία της εφαρμογής, ένα demo.

Στο **πέμπτο κεφάλαιο** γίνεται ένας απολογισμός όσον αφορά τα ακριβή αποτελέσματα της εργασίας και κατά πόσο μπορεί κάποιος να βασίζεται σε αυτά. Τονίζονται επίσης κάποια σημεία που χρήζουν βελτίωσης.

Στο έκτο κεφάλαιο αναφέρονται οι βιβλιογραφικές παραπομπές στις οποίες βασίζεται η συγγραφή της διπλωματικής αυτής εργασίας.

2. ΥΠΟΒΑΘΡΟ

2.1 Σχετικές δουλειές/εργασίες

2.1.1 Υπολογισμός ταχύτητας

Ο υπολογισμός της ταχύτητας ενός ιστιοφόρου σκάφους, σε σχέση με μια δοθείσα ένταση και κατεύθυνση ανέμου, δεν είναι καθόλου εύκολος, ίσως και αδύνατος (Garrett,1996). Το κάθε σκάφος έχει τα δικά του χαρακτηριστικά και χειρίζεται διαφορετικά τον άνεμο. Επίσης, παίζει πολύ σημαντικό ρόλο το τριμάρισμα των πανιών αλλά στην παρούσα εργασία θα πρέπει να θεωρηθεί ότι έχουν την βέλτιστη ρύθμιση.

Παρακάτω φαίνεται η μαθηματική σχέση που ενώνει τον πραγματικό άνεμο με την κατεύθυνση του φαινόμενου ανέμου (Garrett,1996):

$$\sin(a_0) \cdot \sin(a) \cdot \left(\frac{\sin\left(\frac{a}{2}\right)}{\sin(a_0 - a)} \right)^2 = VW \cdot \eta$$

όπου:

$a_0 \rightarrow$ η γωνία του φαινόμενου ανέμου με την κατεύθυνση του σκάφους

$a \rightarrow$ η γωνία πραγματικού ανέμου με την κατεύθυνση του σκάφους

$VW \rightarrow$ η ένταση του πραγματικού ανέμου σε μίλια

$\eta \rightarrow$ ένας συντελεστής για τις φυσικές ιδιότητες του κάθε σκάφους όπως αντίσταση στο νερό. Το κάθε σκάφος έχει διαφορετικό και το μικρότερο σημαίνει μεγαλύτερη ταχύτητα.

Έχοντας λύσει τον παραπάνω τύπο ως προς a_0 , πλέον μπορεί να υπολογιστεί η ταχύτητα με τον παρακάτω τύπο:

$$VB := VW \cdot \frac{\sin(a_0 - a)}{\sin(a)}$$

2.1.2 Συμφέρουσα πορεία

Υπάρχει ένας δείκτης που καθορίζει κατά πόσο μια πορεία είναι αποδοτική, αναφορικά με το σημείο στο οποίο θέλει κάποιος να φτάσει. Ο δείκτης αυτός είναι ο VMG (Velocity Made Good) που υπολογίζεται από τον παρακάτω τύπο¹:

$$VMG, \hat{y} = V_o \cos \theta$$

όπου:

$V_o \rightarrow$ η ταχύτητα του σκάφους

$\theta \rightarrow$ η γωνία μεταξύ πορείας του σκάφους και τελικού προορισμού

Ο τύπος αυτός χρησιμοποιείται κυρίως όταν η πορεία του ταξιδιού είναι τελείως αντίθετη από την κατεύθυνση του ανέμου, οπότε και θα πρέπει να κάνει το σκάφος πολλαπλά TAK. Δείχνει, δηλαδή, την αποδοτικότερη κλίση που θα πρέπει να έχει το σκάφος προς τον άνεμο. Για παράδειγμα, έστω ότι φυσάει βοριάς και ο προορισμός είναι βόρεια. Ο παρακάτω πίνακας δείχνει τιμές κλίσης/ταχύτητας:

Κλίση	Ταχύτητα α	VM G
55	4	2.3
60	5	2.5
65	5.2	2.2

Από τα παραπάνω προκύπτουν τα εξής συμπεράσματα:

- Στις 55 μοίρες, ταξίδι πιο κοντά στον επιθυμητό προορισμό, αλλά η ταχύτητα είναι μικρή
- Στις 65 μοίρες, η ταχύτητα είναι μεγαλύτερη σε σχέση με τις άλλες δυο πλευσεις, αλλά το σκάφος απομακρύνεται πιο πολύ από τον προορισμό
- Στις 60 μοίρες, είναι ο καλύτερος συνδυασμός γωνίας πλευσης και ταχύτητας, που σημαίνει πως η ταχύτητα του σκάφους είναι κατά πολύ μεγαλύτερη,

¹ https://en.wikipedia.org/wiki/Velocity_made_good

παρόλο που υπάρχει πιο μεγάλη απόκλιση σε σχέση με το ταξίδι στις 55 μοίρες. Οπότε η πορεία αυτή δίνει την γρηγορότερη πλεύση.

Ο δείκτης VMG είναι πολύ χρήσιμος, αλλά στον παραπάνω θα πρέπει να είναι γνωστή η ταχύτητα. Είναι ένας τύπος που μπορεί να χρησιμοποιηθεί κατά την διάρκεια του ταξιδιού, όπου η ταχύτητα θα είναι γνωστή από τα όργανα του σκάφους.

Υπάρχει ακόμη ένας τύπος ο οποίος χρειάζεται μόνο την ένταση του ανέμου, την κατεύθυνσή του καθώς και κάποια χαρακτηριστικά του σκάφους. Δεν είναι ακριβείας, ωστόσο πλησιάζει κατά πολύ τις πραγματικές συνθήκες.

- κλειστή πλαγιοδρομία:

$$VMG_{D,\hat{y}} = \frac{V_w}{\cos\theta_s} \left(1 + \frac{\beta}{100}\right)^{\frac{|\theta_o - \theta_s|}{i}} \cos\theta_\gamma$$

- ανοιχτή πλαγιοδρομία:

$$VMG_{D,\hat{y}} = \frac{V_w}{\cos\theta_s} \left(1 + \frac{\beta}{100}\right)^{\frac{|180^\circ - \theta_s - \theta_o|}{i}} \cos\theta_\gamma$$

Όπου:

$V_w \rightarrow$ Ένταση ανέμου

$\theta_s \rightarrow$ Νεκρή ζώνη(όταν η πλεύση είναι αδύνατη πχ 30 μοίρες από τον άνεμο)

$\theta_o \rightarrow$ Γωνία μεταξύ ανέμου και πορείας του σκάφους

$\theta_\gamma \rightarrow$ Γωνία μεταξύ πορείας του σκάφους και τελικού προορισμού

$i \rightarrow$ Διάστημα διαβαθμίσεων(πχ x% ανά 5 μοίρες $\Rightarrow i = 5$)

$\beta \rightarrow$ Σταθερή αύξησης ταχύτητας(πχ 10% ανά μοίρα $\Rightarrow \beta = 10$)

Από τον παραπάνω τύπο, μπορεί να βρεθεί βέβαια και μια σχέση για την ταχύτητα του σκάφους σε σχέση με τον άνεμο, σβήνοντας το τελευταίο κομμάτι ($\cos\theta_\gamma$). Το αποτέλεσμα αυτής της σχέσης δεν είναι η ταχύτητα του σκάφους, αλλά μια τιμή που δείχνει την απόδοση του σκάφους για κάθε γωνία ως προς τον άνεμο που είναι ανάλογη της ταχύτητας και είναι η τιμή που χρησιμοποιήθηκε στην εφαρμογή.

2.1.3 Προγράμματα βέλτιστης διαδρομής(Weather Routing)

Κατά τη βιβλιογραφική ανασκόπηση εντοπίστηκαν ορισμένα προγράμματα που βρίσκουν την βέλτιστη πορεία ιστιοφόρων, αξιοποιώντας τα στοιχεία του καιρού. Πρόκειται για προγράμματα που είναι όλα επί πληρωμή και συνήθως απαιτούν εγκατάσταση. Αυτά τα προγράμματα, έχουν την δυνατότητα να παίρνουν τον καιρό για πολλές μέρες και ώρες και να υπολογίζουν διαδρομές δυναμικά. Το πώς ακριβώς δουλεύουν δεν είναι γνωστό μιας και δεν είναι ανοιχτού κώδικα. Έχουν ρυθμίσεις για πολλούς τύπους σκαφών και κάποια από αυτά έχουν και επιλογή αποφυγής μεγάλων κυμάτων, αν και η πρόβλεψη των κυμάτων δεν μπορεί να είναι πολύ ακριβής, και χρήση μηχανής σε περιοχές με λίγο αέρα.

Κάποια από αυτά είναι:

- PredictWind
- FastSeas - δεν απαιτεί εγκατάσταση και παρέχει μερικά δωρεάν request ανά μήνα
- SailGrib
- Expedition

2.1.4 Βιβλιογραφικό κενό

Με βάση τη βιβλιογραφική ανασκόπηση που προηγήθηκε, προκύπτει ένα βιβλιογραφικό κενό, το οποίο καλείται ο ερευνητής να καλύψει με την εν λόγω εργασία. Το κενό αυτό αφορά στην έλλειψη εφαρμογής που να μπορεί να χρησιμοποιείται από οποιαδήποτε συσκευή, χωρίς να απαιτείται εγκατάσταση, καθώς επίσης και μια τεκμηριωμένη υπολογιστική λύση στο πρόβλημα. Η λύση του προβλήματος αυτού, περιγράφεται αναλυτικά στα επόμενα κεφάλαια.

2.2 Εργαλεία

2.2.1 OpenWeatherMap API

Το OpenWeatherMap είναι ένας πάροχος πρόβλεψης καιρού χρησιμοποιώντας περισσότερους από 40.000 μετεωρολογικούς σταθμούς παγκοσμίως. Στο επίσημο site μπορεί κάποιος να βρει από απλά γραφήματα καιρικών συνθηκών ανά περιοχή μέχρι και χάρτες με διάφορα layers για καιρικά φαινόμενα όπως θερμοκρασία, σύννεφα, βαρομετρικό και άνεμο (openweathermap.org).

Επίσης, η εταιρεία αυτή παρέχει ένα API ώστε να μπορεί οποιοσδήποτε να έχει πρόσβαση σε αυτά τα δεδομένα για την ανάπτυξη εφαρμογών. Κατά κύριο λόγο αυτό το API προσφέρεται δωρεάν εκτός εάν πρόκειται για επαγγελματική χρήση. Το μόνο που χρειάζεται είναι ο χρήστης να δημιουργήσει έναν λογαριασμό και να λάβει ένα appId με το οποίο θα μπορεί να χρησιμοποιεί το API ελεύθερα.

2.2.2 Google Maps API

Το Google Maps API, είναι ένα εργαλείο το οποίο επιτρέπει στον εκάστοτε χρήστη να χρησιμοποιεί τους χάρτες της Google μαζί με διάφορες βιβλιοθήκες μέσω είτε JavaScript είτε Android εφαρμογής κτλ. Αυτό επιτρέπει την απεικόνιση διαφόρων πραγμάτων που χρειάζονται στον χάρτη και την κατάλληλη διαχείρισή τους (developers.google.com/maps).

Όπως και με το OpenWeatherMap API, η Google παρέχει δωρεάν λογαριασμούς για Developers ώστε να βοηθήσει στην ανάπτυξη εφαρμογών με χάρτες και τοποθεσίες. Πρέπει δηλαδή να δημιουργηθεί ένας λογαριασμός και ένα api_key το οποίο θα χρησιμοποιείται για την επικοινωνία με το API της Google.

2.2.3 Hipster4j

Το Hipster4j είναι μια εύχρηστη, δυνατή και ευέλικτη βιβλιοθήκη ευρετικής αναζήτησης και είναι γραμμένη σε Java. Είναι λογισμικό ανοιχτού κώδικα το οποίο διανέμεται με άδεια από την Apache και προσφέρεται για εμπορική και μη χρήση (www.hipster4j.org).

Με την χρήση της βιβλιοθήκης αυτής, μπορεί πολύ εύκολα να δημιουργηθεί γράφος, είτε κατευθυνόμενος είτε όχι, και έπειτα να χρησιμοποιηθεί κάποιος αλγόριθμος βέλτιστης διαδρομής (shortest path) όπως ο αλγόριθμος του Dijkstra ή ακόμα καλύτερα ο A*(βελτιωμένη έκδοση του Dijkstra).

2.2.4 GeoTools και ShapeFile

Το GeoTools είναι μια βιβλιοθήκη ανοιχτού λογισμικού, γραμμένη σε Java, που επιτρέπει την διαχείριση γεωχωρικών δεδομένων. Η βιβλιοθήκη αυτή υλοποιεί τις προδιαγραφές Open Geospatial Consortium (OGC) όπως αυτές έχουν αναπτυχθεί (geotools.org).

Το ShapeFile είναι αρχείο(συνδυασμός αρχείων) το οποίο διαβάζεται από την βιβλιοθήκη GeoTools και παρέχει γεωγραφικές πληροφορίες για κάποιες συγκεκριμένες συντεταγμένες. Συνήθως παρέχονται ανά χώρα (www.gadm.org).

Ο συνδυασμός των παραπάνω επιτρέπει στον κώδικα να δημιουργήσει έναν χάρτη και να βρίσκει πληροφορίες για το κάθε σημείο σε αυτόν. Το κυριότερο είναι ότι μπορεί να διαχωρίσει εάν ένα σημείο είναι θάλασσα ή όχι.

3. ΥΛΟΠΟΙΗΣΗ

3.1 Σχεδιασμός

Για την επίλυση του προβλήματος, έπρεπε να δημιουργηθεί μια εφαρμογή η οποία ανά πάσα στιγμή να μπορεί να επιστρέφει την βέλτιστη διαδρομή μεταξύ δυο σημείων. Προτιμήθηκε να είναι κάτι γενικό και άμεσο στην χρήση του όπως η πλοήγηση με τους χάρτες της Google (Google Maps).

Ο σχεδιασμός του συστήματος φαίνεται στην παρακάτω εικόνα:



σχ. 1. Σχεδιασμός λειτουργίας

3.1.1 Back-End

Οι απαραίτητες πληροφορίες θα πρέπει να κρατούνται σε ένα κεντρικό σύστημα και να είναι διαθέσιμες στον χρήστη στιγμιαία. Για τον λόγο αυτόν, κρίθηκε αναγκαία η χρήση ενός Web Service το οποίο θα μπορεί να καλείται από οποιαδήποτε συσκευή, είτε Desktop είτε Mobile, Android ή iOS. Αυτό κάνει την εφαρμογή πιο εύχρηστη και ευέλικτη αφού με τον τρόπο αυτόν δεν χρειάζεται να δημιουργηθεί καινούρια εφαρμογή για την κάθε πλατφόρμα.

Στην αρχικοποίηση του το Web Service βρίσκει όλα τα σημεία που είναι διαθέσιμα να τα χρησιμοποιήσει η εφαρμογή. Έπειτα, γίνεται λήψη πληροφοριών για τον καιρό σε προκαθορισμένα σημεία. Η διαδικασία για τις καιρικές συνθήκες

επαναλαμβάνεται ανά τακτά χρονικά διαστήματα, αφού ο άνεμος αλλάζει συχνά ως προς την ένταση και την κατεύθυνσή του οπότε το σύστημα θα πρέπει να είναι ενημερωμένο. Πλέον το Web Service έχει όλη την πληροφορία που χρειάζεται ώστε να μπορέσει υπολογίσει την βέλτιστη διαδρομή μεταξύ δυο σημείων και να την επιστρέψει στον χρήστη.

Για την επίτευξη καλύτερης εμπειρίας χρήσης, υπάρχουν δυο λειτουργίες διαθέσιμες προς τον χρήστη:

1. **validatePoint**: Η λειτουργία αυτή επιτρέπει στην χρήστη να δει εάν το σημείο που έχει επιλέξει υπάρχει στο σύστημα οπότε και να μπορεί να το χρησιμοποιήσει
2. **sail**: Αφού έχουν επιλεγεί δυο σημεία που υπάρχουν στο σύστημα και είναι διαθέσιμα προς χρήση, η λειτουργία sail επιστρέφει το μονοπάτι που πρέπει να ακολουθηθεί ώστε να βελτιστοποιήσουμε το ταξίδι μας

3.1.2 Front-End

Μετά την δημιουργία του Web Service όπου και θα γίνονται όλοι οι υπολογισμοί, θα πρέπει να απεικονισθούν τα αποτελέσματα σε έναν χάρτη για λόγους ευχρηστίας. Το front-end θα επικοινωνεί με το back-end μέσω Ajax Request και όλα είναι άμεσα.

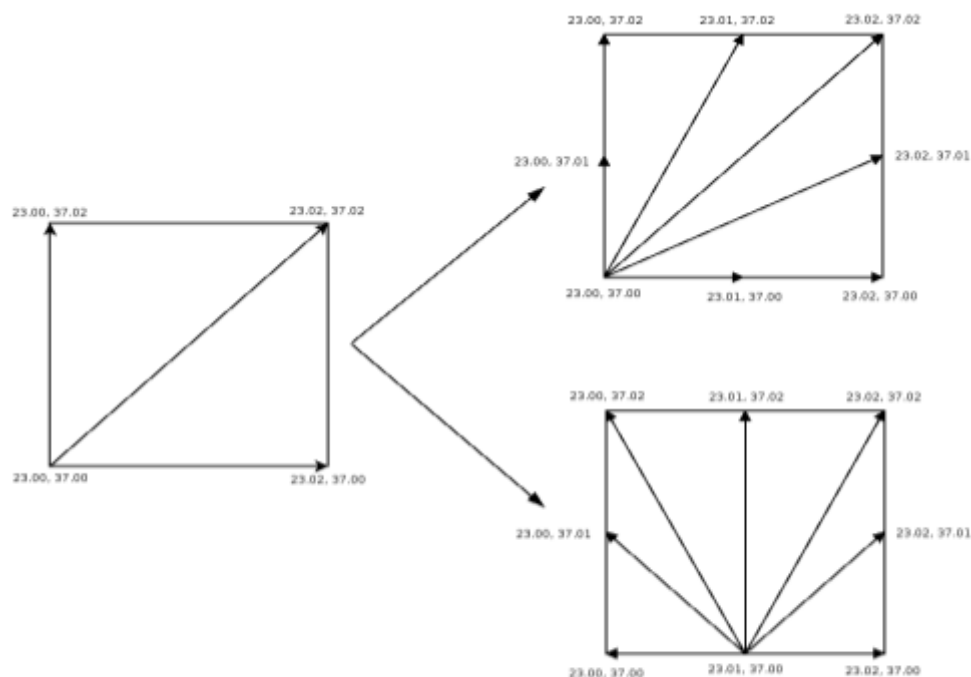
Αρχικά, επιλέγει ο χρήστης τα δυο σημεία που θέλει και αφού γίνει το validation και μπορεί τελικά να τα χρησιμοποιήσει, στέλνει ένα ακόμα request για να δει την συντομότερη διαδρομή μεταξύ των επιλεγθέντων σημείων. Η απάντηση θα περιέχει το μονοπάτι οπότε μένει μόνο να απεικονισθεί στον χάρτη.

3.2 Υλοποίηση

3.2.1 Εντοπισμός σημείων και διαθέσιμων διαδρομών

Κατά την αρχικοποίηση, όπως φάνηκε παραπάνω, πρέπει να γνωστοποιηθούν ποια σημεία είναι διαθέσιμα, δηλαδή τα σημεία που βρίσκονται στην θάλασσα, και τι διαδρομές μπορεί να κάνει το σκάφος από κάθε σημείο. Λόγω του ότι το σύστημα το διαχειρίζεται ο υπολογιστής, πρέπει να έχει προκαθορισμένα σημεία. Για παράδειγμα, για την λειτουργία του προγράμματος για τον Σαρωνικό, θα πρέπει να του υποδειχθούν τα σημεία γεωγραφικού μήκους 23.00 μέχρι 24.00 και γεωγραφικού πλάτους 37.00 μέχρι 38.00. Πρέπει να καθοριστεί και ένα βήμα με το οποίο θα ελέγχονται τα σημεία, για παράδειγμα 0.02 που σημαίνει ότι θα ελεγχθούν όλα τα σημεία μεταξύ των παραπάνω συντεταγμένων με απόσταση μεταξύ τους 0.02 μοίρες.

Με αυτόν τον τρόπο, περιορίζεται ο αριθμός των σημείων που θα χρησιμοποιηθούν αλλά επίσης περιορίζονται και οι πορείες που θα μπορεί να πάρει το σκάφος. Έτσι λοιπόν, επιλέχθηκε να χρησιμοποιηθούν και ενδιάμεσα σημεία έτσι ώστε να έχουμε περισσότερες επιλογές. Εξηγείται στο παρακάτω σχήμα:



σχ. 2. Επιλογή σημείων

Όπως φαίνεται λοιπόν, με την χρήση των ενδιάμεσων σημείων, οι πορείες που μπορεί να πάρει το σκάφος έχουν διαφορά 22.5 μοιρών έναντι των 45 μοιρών, τις επιλογές που θα υπήρχαν χρησιμοποιώντας μόνο τα κύρια σημεία.

Αφού αποφασίστηκε λοιπόν η διάταξη των διαθέσιμων σημείων, θα πρέπει να γίνει έλεγχος για το κάθε σημείο εάν είναι γη ή θάλασσα και έπειτα, για τον κάθε δυνατό προορισμό, σύμφωνα με το παραπάνω σχήμα, θα πρέπει να ελεγχθεί αν το αντίστοιχο μονοπάτι είναι θαλάσσιο ή αν υπάρχει στεριά ενδιάμεσα.

Εδώ έρχεται η σειρά της βιβλιοθήκης GeoTools και ενός ShapeFile για την Ελλάδα. Για κάθε σημείο, πρώτα ελέγχεται αν το σημείο αυτό υπάρχει στο ShapeFile. Εάν υπάρχει, τότε πρόκειται για στεριά. Σε αυτήν την περίπτωση δεν χρειάζεται να ελεγχθούν οι διαθέσιμες διαδρομές από το σημείο αυτό οπότε παρακάμπτεται. Αλλιώς, εντοπίζονται οι πιθανοί γείτονές του και πλέον, για κάθε γείτονα, ελέγχεται αν ο δρόμος προς αυτόν είναι θαλάσσιος ή αν περιέχονται στεριανά σημεία. Για να γίνει αυτό, το μονοπάτι χωρίζεται σε N (πχ 20) ίσα κομμάτια, άρα υπάρχουν $N - 1$ σημεία να ελεγχθούν όπως έγινε προηγουμένως.

Χρησιμοποιώντας το παραπάνω configuration με κύρια σημεία ανά 0.02 μοίρες με ενδιάμεσα, η μεγαλύτερη απόσταση μεταξύ δυο σημείων θα είναι 3.14km άρα ο έλεγχος των σημείων θα γίνεται ανά 157 μέτρα. Αυτά τα νούμερα προκύπτουν από τις παρακάτω παραδοχές:

- Η μοίρα έχει 60 λεπτά
- Η απόσταση μεταξύ δυο λεπτών είναι 1 ναυτικό μίλι(περίπου 1852 μέτρα)
- Το κάθε λεπτό έχει 60 δευτερόλεπτα

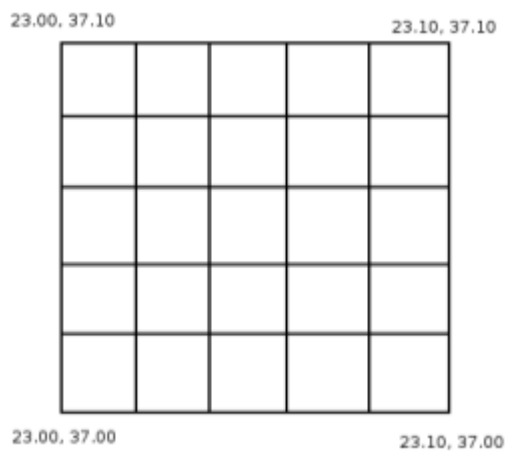
Σύμφωνα με τα παραπάνω, το βήμα 0.02 μοίρες ισούται με 1 λεπτό και 12 δευτερόλεπτα δηλαδή περίπου 2.22km. Από το ισοσκελές ορθογώνιο νοητό τρίγωνο που δημιουργείται(βλ σημεία [23.00, 37.00] [23.02, 37.00] [23.02, 37.02]), η υποτεινούσα υπολογίζεται περίπου στα 3.14km.

Όλα τα παραπάνω κρατούνται σε λίστα για την ευκολότερη διαχείρισή τους. Υπάρχει, δηλαδή, μια λίστα από σημεία τα οποία περιέχουν από μια λίστα με τους πιθανούς προορισμούς.

3.2.2 Ο Καιρός

Για να λαμβάνει το πρόγραμμα πληροφορίες για τον καιρό, χρησιμοποιείται το API του OpenWeatherMap που αναφέρθηκε στο προηγούμενο κεφάλαιο. Δεν είναι όμως δυνατόν να γνωρίζει τον καιρό για όλα τα σημεία του χάρτη, κάτι τέτοιο ίσως να ήταν και άσκοπο εκτός από κοστοβόρο. Έτσι, αποφασίστηκε να μην γίνεται λήψη του καιρού για κάθε ένα από τα σημεία της λίστας αλλά ανά κάποιες περιοχές. Αυτό σημαίνει ότι πολλά σημεία μπορεί να έχουν τον ίδιο καιρό.

Το σύστημα λαμβάνει τον καιρό με την ίδια λογική που πήρε και τα σημεία. Λόγω του ότι υπάρχει πλάνο για την κάλυψη ευρύτερης περιοχής, εδώ χρησιμοποιείται βήμα μεγαλύτερο από αυτό των σημείων. Για την κάθε περιοχή, ορίζεται η ταχύτητα και ένταση του ανέμου ως την μέση τιμή των τεσσάρων σημείων που την καθορίζουν.



σχ. 3. Μετρήσεις ανέμου

Στο παραπάνω σχήμα φαίνεται μία περιοχή καιρού όπου έχει χρησιμοποιηθεί βήμα 0.10 μοίρες και μέσα φαίνονται τα σημεία του χάρτη με βήμα 0.02 μοίρες όπως έγινε γνωστό στην προηγούμενη ενότητα. Για όλα τα σημεία που εμπεριέχονται στην

περιοχή αυτή, θεωρείται πως έχουν τον ίδιο καιρό ως το μέσο όρο των τεσσάρων ορίων της περιοχής.

Αφού είναι πλέον γνωστός ο καιρός, δημιουργείται ένας γράφος όπως φαίνεται στην επόμενη ενότητα. Λόγω του ότι ο καιρός δεν είναι σταθερός, η διαδικασία αυτή επαναλαμβάνεται κάθε 3 ή 5 ώρες.

3.2.3 Ο Γράφος

Ο γράφος είναι το κυριότερο κομμάτι της εφαρμογής. Για την δημιουργία του και την διαχείρισή του χρησιμοποιήθηκε η βιβλιοθήκη Hipster4j που αναφέρεται στο κεφάλαιο 2. Αφού έχουν δημιουργηθεί και κρατηθεί στην λίστα τα έγκυρα σημεία (θαλάσσια) και ο καιρός είναι γνωστός, μπορούν πλέον να τοποθετηθούν όλα αυτά σε έναν γράφο.

Περνώντας λοιπόν ένα ένα τα σημεία της λίστας, για την μετάβαση προς κάθε γειτονικό σημείο υπολογίζεται ένα βάρος. Υπάρχουν δυο επιλογές για την υλοποίηση:

1. Χρησιμοποιώντας τους τύπους για την ταχύτητα του σκάφους αναλόγως της έντασης και κατεύθυνσης του ανέμου, βρίσκεται η ταχύτητα και έπειτα σαν βάρος υπολογίζεται ο χρόνος που χρειάζεται για να διανυθεί η εκάστοτε απόσταση
2. Χρησιμοποιώντας τον δείκτη VMG (Velocity Made Good) μπορούν να εντοπιστούν οι πιο συμφέρουσες διαδρομές. Η καλύτερη διαδρομή έχει και μεγαλύτερο δείκτη. Ο δείκτης είναι ανάλογος της ταχύτητας και της πορείας ως προς τον τελικό προορισμό

Τώρα, έχοντας δημιουργήσει τον γράφο, πρέπει να βρεθεί η βέλτιστη διαδρομή μεταξύ δυο σημείων. Το πρόβλημα, πλέον, ανάγεται σε κλασσικό shortest path πρόβλημα το οποίο μπορεί να λυθεί χρησιμοποιώντας τον αλγόριθμο του Dijkstra ή ακόμα καλύτερα τον A star (βελτιωμένη έκδοση του Dijkstra).

Οι shortest path αλγόριθμοι χρησιμοποιούν επιστρέφουν ως αποτέλεσμα τον συνδυασμό κόμβων με ελάχιστο κόστος. Στην περίπτωση λοιπόν που έχει

χρησιμοποιηθεί ο πρώτος τρόπος κατανομής βαρών, το συνολικό κόστος θα είναι και ο χρόνος που θα χρειαστεί για την ολοκλήρωση της διαδρομής.

Εάν όμως επιλεγθεί ο δεύτερος, θα πρέπει το βάρος να υπολογιστεί σαν 1 / VMG ώστε η μικρότερη τιμή να είναι και η πιο συμφέρουσα. Επίσης, θα πρέπει ο γράφος να δημιουργείται δυναμικά κάθε φορά αφού ο προορισμός θα αλλάζει. Μπορεί όμως να χρησιμοποιηθεί ο τύπος για την ταχύτητα (μια σχέση αυτής) όπως επισημάνθηκε στο δεύτερο κεφάλαιο. Σε αυτήν την περίπτωση, δεν μπορεί να υπολογισθεί το συνολικό κόστος (χρόνος) της διαδρομής, ίσως χρησιμοποιώντας κάποια επιπλέον παράμετρο να ήταν δυνατός αυτός ο υπολογισμός.

3.3 Επικοινωνία

3.3.1 *validatePoint*

Η πρώτη λειτουργία του Web Service που θα χρησιμοποιήσει ο χρήστης είναι η *validatePoint*. Με αυτήν την λειτουργία, ο χρήστης στέλνει ένα σημείο του χάρτη στο Web Service για να δει εάν το σημείο αυτό υπάρχει στον γράφο, αν δηλαδή υποστηρίζεται για πλοήγηση από την εφαρμογή.

Αφού, όπως εξηγήθηκε παραπάνω, τα σημεία που έχει το σύστημα είναι συγκεκριμένα, ο χρήστης είναι αδύνατο να τα πετύχει. Έτσι λοιπόν, ζητώντας επιβεβαίωση για ένα σημείο, το σύστημα θα ψάξει το κοντινότερο σε αυτό σημείο (εντός μιας μικρής εμβέλειας) και θα το επιστρέψει. Με αυτόν τον τρόπο, ο χρήστης δεν χρειάζεται να γνωρίζει από πριν ποια σημεία υποστηρίζονται από την εφαρμογή.

Παρακάτω φαίνεται ένα παράδειγμα από Request και Response για το *validatePoint*:

```
<?xml version="1.0" encoding="utf-8"?>
<sailService>
  <request name="validatePoint">
    <point>23.00,37.00</point>
  </request>
</sailService>
```

Με Response:

```
<?xml version="1.0" encoding="utf-8"?>
<validatePoint>
  <response>
    <lat>37.00</lat>
    <lon>23.00</lon>
    <speed>12.30</speed>
    <direction>165</direction>
  </response>
</validatePoint>
```

3.3.2 sail

Η λειτουργία αυτή είναι η βασική λειτουργία της εργασίας. Εδώ, ο χρήστης στέλνει δυο σημεία, που ήδη έχουν επαληθευτεί από το σύστημα ώστε να είναι σίγουρος ότι υπάρχουν στον γράφο, και ζητάει την συντομότερη διαδρομή. Ο γράφος είναι έτοιμος οπότε η απάντηση από το Service έρχεται άμεσα.

Παρακάτω φαίνονται τα αντίστοιχα Request-Response:

```
<?xml version="1.0" encoding="utf-8"?>
<sailService>
  <request name="sail">
    <from>23.00,37.00</from>
    <to>23.06,37.04</to>
  </request>
</sailService>
```

Εάν βρέθηκε διαδρομή, επιστρέφεται η παρακάτω απάντηση:

```
<?xml version="1.0" encoding="utf-8"?>
<sail>
  <response>
    <from>23.00,37.00</from>
    <to>23.06,37.04</to>
    <route>23.02,37.02|23.04,37.04|23.06,37.04</route>
  </response>
```

```
</sail>
```

Για την βέλτιστη διαδρομή δηλαδή, επιστρέφονται όλα τα ενδιάμεσα σημεία από τα οποία θα πρέπει να περάσει το σκάφος, χωρισμένα με το σύμβολο “|”.

Εάν όμως δεν βρεθεί δρόμος, η απάντηση έρχεται ως εξής:

```
<?xml version="1.0" encoding="utf-8"?>
<sail>
  <response>
    <from>23.00,37.00</from>
    <to>23.06,37.04</to>
    <route>No Sail</route>
  </response>
</sail>
```

3.4 Απεικόνιση

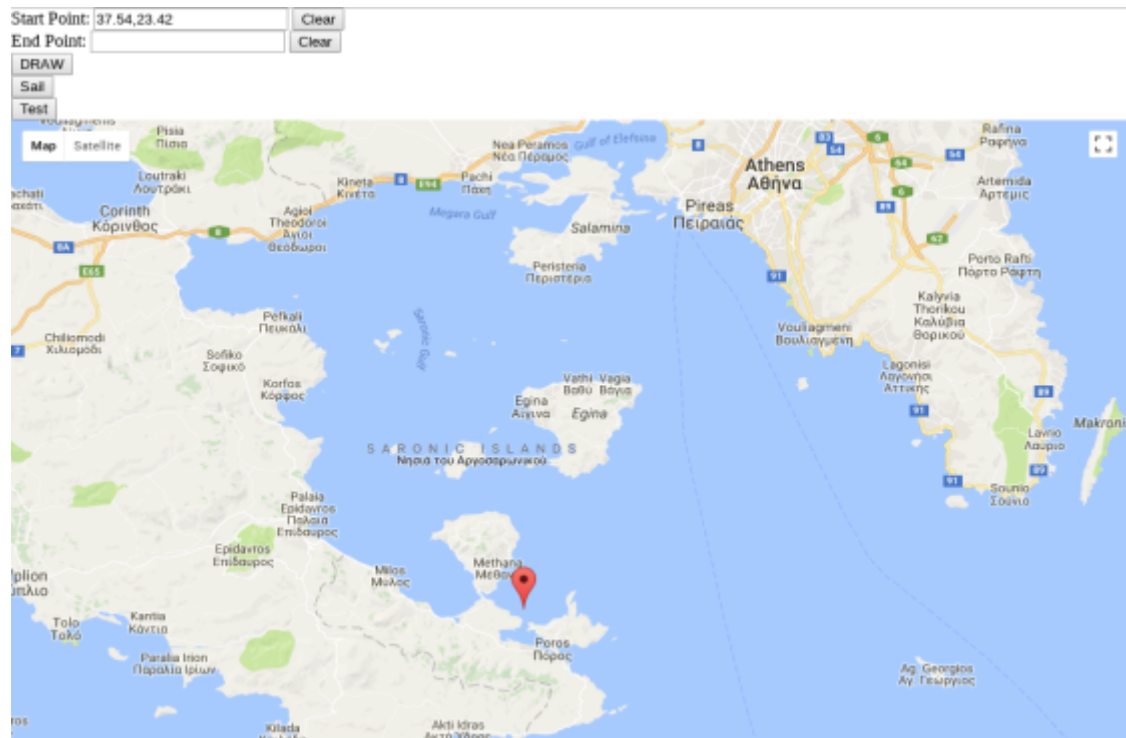
Για την απεικόνιση των παραπάνω, έχει χρησιμοποιηθεί το API του Google Maps έτσι ώστε να μπορέσουν να περαστούν τα αποτελέσματα του συστήματος στον χάρτη. Στην περίπτωση βέβαια της εφαρμογής αυτής, δεν χρειάστηκε κάποια ιδιαίτερη λειτουργία μιας και η απεικόνιση της διαδρομής υλοποιήθηκε με την λειτουργία draw, δηλαδή δημιουργία γραμμών για την ένωση σημείων. Η απεικόνιση φαίνεται στο επόμενο κεφάλαιο όπου υπάρχουν και κάποιες εικόνες από την εφαρμογή και εξηγείται πώς αυτή δουλεύει.

4. ΑΠΟΤΕΛΕΣΜΑΤΑ

4.1 Παρουσίαση της εφαρμογής

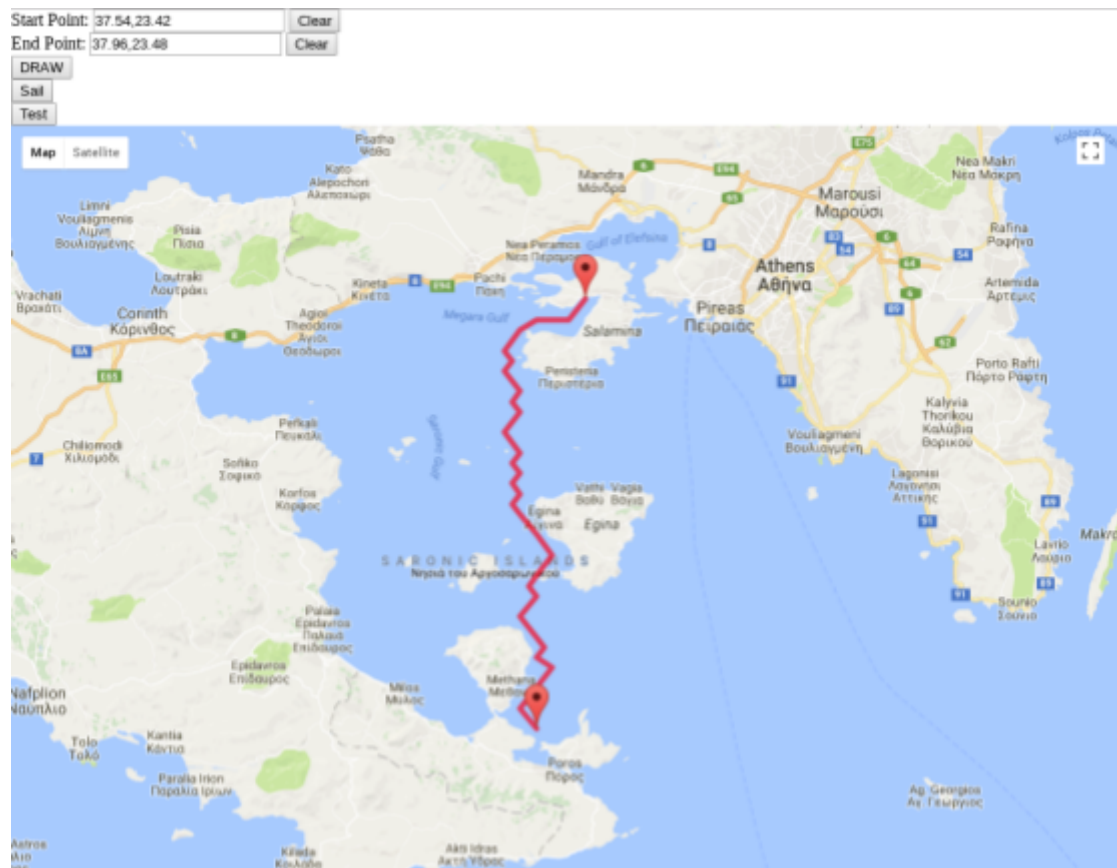
Η χρήση της εφαρμογής είναι πολύ απλή. Αρχικά, πρέπει να βρεθούν δυο σημεία στον χάρτη που να υπάρχουν στο σύστημα ώστε να μπορούν να χρησιμοποιηθούν για πλοήγηση. Αφού τοποθετηθούν στα αντίστοιχα κουτιά, στέλνεται το request **sail** και επιστρέφει την βέλτιστη διαδρομή.

Στην παρακάτω οθόνη, με δεξί κλικ σε κάποιο σημείο, γίνεται αναζήτηση για το ίδιο και τα πολύ κοντινά σε αυτό σημεία εάν υπάρχουν στο σύστημα. Εάν υπάρχει, τοποθετούνται οι συντεταγμένες του στο αντίστοιχο κουτάκι.



Εικ. 3. Παράδειγμα validatePoint

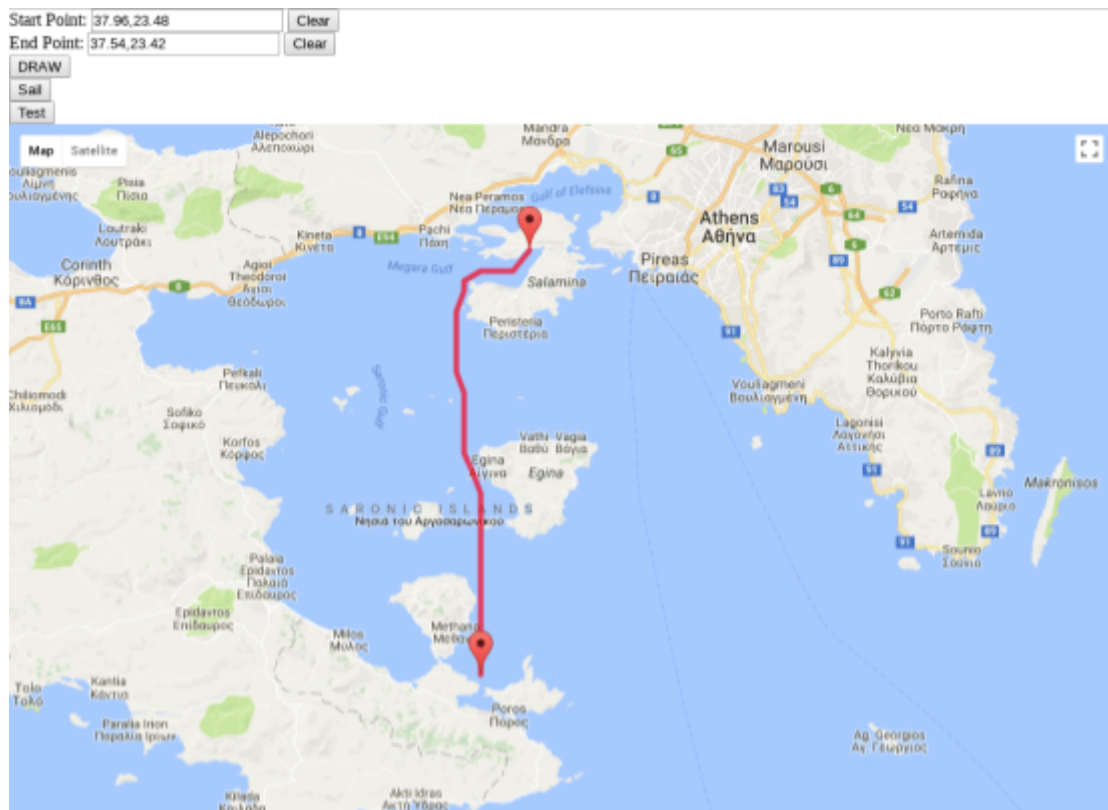
Έπειτα, αφού επιλεγθεί και το δεύτερο σημείο, χρησιμοποιείται το κουμπί Sail. Τότε, στέλνεται ένα Ajax Request στο σύστημα με τα δυο σημεία και αυτό υπολογίζει την βέλτιστη διαδρομή που θα πρέπει να ακολουθηθεί για την ταχύτερη πλεύση σύμφωνα με τις τρέχουσες καιρικές συνθήκες. **Στο παράδειγμα, ο καιρός έχει υπολογιστεί χειροκίνητα με ένταση 18.5 μίλια και κατεύθυνση 355 μοίρες, δηλαδή Βοριάς.**



Εικ. 4. Παράδειγμα πορείας με πολλαπλά TAK

Στην εικόνα φαίνονται τα πολλαπλά TAK που θα πρέπει να πραγματοποιηθούν μιας και ο προορισμός είναι βόρεια, δηλαδή αντίθετα από την κατεύθυνση του ανέμου.

Αν αντιστραφούν τα σημεία εκκίνησης και τερματισμού, το αποτέλεσμα θα πρέπει να είναι μια πολύ πιο ομαλή πορεία χωρίς κανένα TAK, αφού αυτήν την φορά η πορεία του σκάφους είναι σχεδόν παράλληλη με την κατεύθυνση του ανέμου. Η περίπτωση αυτή φαίνεται στην παρακάτω εικόνα:

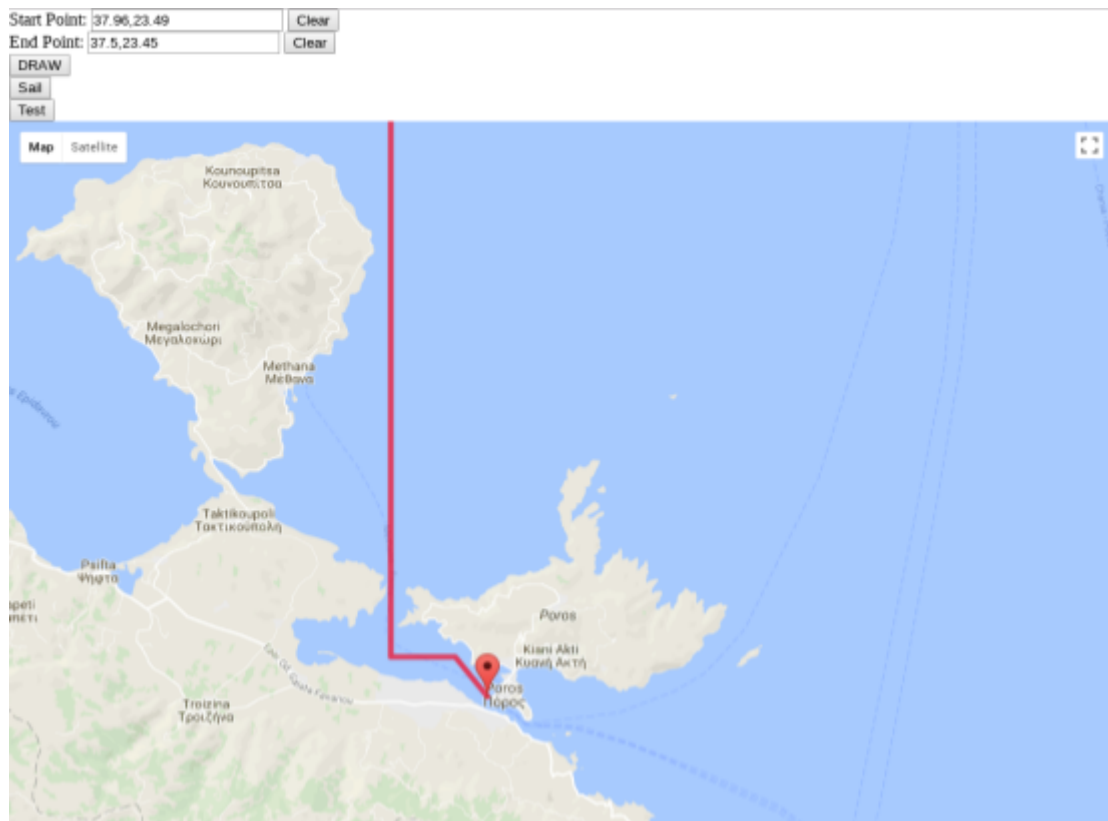


Εικ. 5. Κατεύθυνση ίδια με αυτήν του ανέμου

Υπάρχουν βέβαια και περιπτώσεις όπου είναι δυνατή μόνο η μία διαδρομή με πανιά ενώ η αντίθετη όχι. Παράδειγμα, η διαδρομή από την Σαλαμίνα προς το λιμάνι του Πόρου με τον ίδιο άνεμο με πριν.

Όπως φαίνεται στην παρακάτω εικόνα, μπαίνοντας στο λιμάνι του Πόρου με Βοριά, δεν αντιμετωπίζεται κανένα πρόβλημα αφού δεν χρειάζεται να κάνει το σκάφος κανένα ΤΑΚ μιας και η πλεύση είναι δευτερόπρυμα ή κατάπρυμα. Εάν όμως θέλει κάποιος να βγει από το λιμάνι του Πόρου προς οποιαδήποτε άλλη κατεύθυνση, δεν είναι δυνατή καθαρά η ιστιοπλοΐα οπότε το σύστημα θα επιστρέφει “Δεν βρέθηκε διαδρομή!”. Ο λόγος είναι πως το πέρασμα είναι πολύ στενό, οπότε η έξοδος από το λιμάνι θα πρέπει να γίνει με αντίθετη κίνηση του σκάφους στην κατεύθυνση του ανέμου.

Πρακτικά γίνεται και πολλαπλά και γρήγορα ΤΑΚ σε μικρό χώρο αλλά το σύστημα έχει προκαθορισμένα σημεία και διαδρομές οπότε περιπτώσεις σαν αυτήν θα επιστρέφουν ότι η ιστιοπλοΐα σε αυτό το σημείο δεν είναι δυνατή.



Εικ. 6. Στενό πέρασμα

5. ΣΥΜΠΕΡΑΣΜΑΤΑ

5.1 Τελικός Απολογισμός

Ο υπολογισμός βέλτιστης διαδρομής ενός ιστιοπλοϊκού είναι μια πολύ πολύπλοκη υπόθεση, ίσως και αδύνατη στο να είναι ακριβής. Αυτό διότι η φύση της ιστιοπλοΐας και ο τρόπος με τον οποίο τα πανιά χρησιμοποιούν τον άνεμο για την κίνηση του σκάφους είναι πολύ δύσκολο να προβλεφθεί από οποιονδήποτε μαθηματικό τύπο. Πάντα παίζουν ρόλο τα φυσικά χαρακτηριστικά του σκάφους και στον υπολογισμό θα πρέπει να χρησιμοποιείται και η παραμικρή λεπτομέρεια. Εκτός αυτού, ακόμα και οι συνθήκες που επικρατούν στην θάλασσα μπορούν να αλλάξουν κατά πολύ την ταχύτητα. Για παράδειγμα, αν σε μια περιοχή έχει πολύ κύμα, σίγουρα

αυτό θα είναι ένα επιπλέον εμπόδιο για το σκάφος και το κάθε σκάφος θα το αντιμετωπίζει με διαφορετικό τρόπο.

Ακόμα και ο υπολογισμός του καιρού δεν μπορεί να είναι ακριβής κατά την πρόγνωση για τις επόμενες ώρες ή και μέρες (Kalnay, 2003). Όσοι έχουν εμπειρία στην θάλασσα, παίρνουν τον καιρό λίγο πριν ξεκινήσουν το ταξίδι γιατί οι προβλέψεις από την προηγούμενη ημέρα είναι δύσκολο να επαληθευτούν με ακρίβεια. Γι' αυτόν τον λόγο δεν θεωρήθηκε σκόπιμο να δημιουργείται ο γράφος δυναμικά με τον καιρό. Ίσως σε κάποια βελτιωμένη έκδοση να υπάρχει αυτή η λειτουργία.

Το πρόγραμμα που δημιουργήθηκε, είναι στην ουσία ένα Web Service που μπορεί να χρησιμοποιηθεί από πολλές πλατφόρμες και δεν θα χρειάζεται ούτε συνδρομή ούτε εγκατάσταση. Η χρήση του θα είναι απλή, όπως και ο σκοπός του.

Τέλος, σε καμία περίπτωση δεν θα πρέπει να βασίζεται κάποιος σε εφαρμογές τέτοιου είδους γιατί, όπως αναφέρθηκε πιο πάνω, καμία δεν μπορεί να είναι τελείως ακριβής. Εξαρτάται κατά πολύ από το σκάφος αλλά μερικές φορές ίσως και από το ίδιο το πλήρωμα μιας και αν βγάλει το πρόγραμμα ότι πρέπει να γίνουν πολλαπλά ΤΑΚ, το πλήρωμα μπορεί να αποφασίσει να μην το ακολουθήσει επειδή θεωρεί ότι είναι κουραστικό και έχει κάποιο κόστος. Η χρήση της εφαρμογής αυτής θα πρέπει να είναι καθαρά ως βοήθημα.

5.2 Βελτιώσεις

Υπάρχουν πολλά σημεία στο πρόγραμμα που χρειάζονται επανασχεδίαση καθώς και ελλείψεις αρκετών λειτουργιών που θα έπρεπε να ενσωματωθούν ώστε να προσφέρει καλύτερα αποτελέσματα. Παρακάτω προτείνονται ορισμένες βελτιώσεις ως πεδίο για μελλοντική έρευνα.

- Ομαλοποίηση διαδρομής: Επειδή τα σημεία είναι ήδη καταχωρημένα σε έναν γράφο, είναι συγκεκριμένα. Μερικές φορές, ενώ υπάρχει η δυνατότητα για

μετάβαση από ένα σημείο σε ένα άλλο απ' ευθείας, το πρόγραμμα επιστρέφει μερικές μεταβάσεις σε άλλα σημεία, επειδή ο αρχικός και ο τελικός προορισμός δεν ενώνονται μεταξύ τους. Έτσι, παρατηρούνται κάποιες αλλαγές πορείας χωρίς όμως να υπάρχει λόγος

- Ένωση σημείων χειροκίνητα: Κάποια σημεία και κάποια περάσματα δεν υπάρχουν στο πρόγραμμα διότι δεν μπορεί να βρεθεί ένωση μεταξύ τους, οπότε μια απλή διαδρομή θα πρέπει να περάσει από εναλλακτική. Ένα χαρακτηριστικό παράδειγμα είναι ο Ισθμός της Κορίνθου, όπου για να μεταβεί κάποιος από τον Κορινθιακό στον Σαρωνικό κόλπο, μπορεί πολύ απλά να περάσει από τον Ισθμό και να γλιτώσει τον γύρο της Πελοποννήσου
- Καιρός για ολόκληρη περιοχή: Προς το παρόν, η πληροφορία για τον καιρό λαμβάνεται ανά σημείο. Αυτό καθυστερεί πολύ και ίσως να είναι σπατάλη χρόνου. Χρειάζεται, λοιπόν, μια αποδοτικότερη λύση
- Υπολογισμός συνολικού χρόνου: Για την καλύτερη εμπειρία χρήσης, θα ήταν καλό να επιστρέφεται και ο συνολικός χρόνος που θα χρειαστεί για την διαδρομή
- Προσθήκη ενδιάμεσων σημείων: Η προσθήκη ενδιάμεσων σημείων θα ήταν μια πολύ χρήσιμη λειτουργία μιας και ο χρήστης θα μπορεί να επιλέξει την διαδρομή που θέλει να ακολουθήσει. Μπορεί για παράδειγμα να μην θέλει να περάσει ανάμεσα από την Αίγινα και το Αγκίστρι. Επίσης, με την λειτουργία αυτή, θα μπορεί να προγραμματιστεί και ολόκληρη εκδρομή με στάσεις σε διάφορα σημεία. Η λειτουργία αυτή, ακόμη, θα ήταν χρήσιμη σε αγώνες αφού συνήθως πρέπει τα σκάφη να περάσουν από συγκεκριμένα σημεία
- Δυναμικός καιρός: Καλό θα ήταν, για να είναι πιο πραγματικό, να υπάρχει δυνατότητα να γίνει λήψη του καιρού για τις επόμενες 3 και 6 ώρες, οτιδήποτε παραπάνω ίσως είναι άσκοπο
- Λιμάνια: Πολύ καλό επίσης θα ήταν να σημειωθούν τα διαθέσιμα λιμάνια ώστε να μπορεί ο χρήστης να επιλέξει διαδρομή από λιμάνι σε λιμάνι χωρίς να το ψάχνει στον χάρτη για να το βάλει χειροκίνητα

- Δυνατότητα μεικτής μετακίνησης: Επιλογή ώστε να χρησιμοποιείται μηχανή σε περιοχές με πολύ μικρής έντασης άνεμο, κάτι που θα ήταν πιο κοντά σε πραγματικές συνθήκες ταξιδιού(όχι αγώνα)

6. ΒΙΒΛΙΟΓΡΑΦΙΑ

Clancy, L.J. (1975). *Aerodynamics*. New Jersey: John Wiley & Sons

Garrett, R. (1996). *The symmetry of sailing: The physics of sailing for yachtsmen*.

Adlard Coles Nautical.

Καλαθάκης, Μ., Μάνθος, Κ., Καραπέτης, Γ. (2016). *Η ιστιοπλοία χωρίς μυστικά*.

Αθήνα: ΕΜΠΝΕΥΣΗ.

Kalnay, E. (2003). *Atmospheric modeling, data assimilation and predictability*,

Cambridge University Press, Cambridge

Marchaj, C. A. (2000). *Aero-hydrodynamics of sailing*. Adlard Coles Nautical.

Google Maps API. (n.d). Retrieved from <https://developers.google.com/maps/>

GeoTools. (n.d.). Retrieved from <http://geotools.org/>

Global Administrative Areas. (n.d.). Retrieved from <http://www.gadm.org/>

OpenWeatherMap. (n.d.). Current weather and forecast. Retrieved from

<https://openweathermap.org/>

Hipster4j. (n.d.). Retrieved from <http://www.hipster4j.org/>

Velocity made good. (2017, August 04). Retrieved from

https://en.wikipedia.org/wiki/Velocity_made_good