

ΔΗΜΟΚΡΙΤΕΙΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΘΡΑΚΗΣ
ΠΟΛΥΤΕΧΝΙΚΗ ΣΧΟΛΗ ΞΑΝΘΗΣ
ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΑΡΑΓΩΓΗΣ ΚΑΙ ΔΙΟΙΚΗΣΗΣ



Μάθημα: Ευφυή Συστήματα Βιομηχανικής Παραγωγής
Διδάσκοντες: κ. Κουλουριώτης Δημήτριος, κ. Μπαλαφούτης
Αθανάσιος

**ΕΡΓΑΣΙΑ ΣΤΟ SIMULINK: ΑΥΤΟΜΑΤΟΣ ΚΑΙ ΕΥΦΥΗΣ
ΕΛΕΓΧΟΣ ΓΩΝΙΑΣ ΠΛΕΥΣΗΣ ΠΛΟΙΟΥ ΜΕΣΩ ΡΥΘΜΙΣΗΣ
ΤΗΣ ΓΩΝΙΑΣ ΠΗΔΑΛΙΟΥ**

Λούκας Πέτρος (92237)
Τσουλφαΐδης Αναστάσιος (92249)

Ε' Χειμερινό Εξάμηνο
Ακαδημαϊκό Έτος 2023-2024
Ξάνθη 2024

I. Περιεχόμενα

1. Εισαγωγή.....	5
2. Θεωρητικό Υπόβαθρο	6
2.1 Το πεδίο ενός πλοίου	6
3. Το Μοντέλο	7
4. Υλοποίηση στο SIMULINK.....	8
4.1 Υλοποίηση μέσω αυτόματου ελεγκτή PID.....	8
4.2 Υλοποίηση μέσω ευφυούς ελεγκτή FUZZY	12
4.3 Σύγκριση των 2 ελεγκτών	21
5. Συμπεράσματα.....	23
6. Βιβλιογραφικές Αναφορές	24

II. Κατάλογος Εικόνων

Εικόνα 1: Αναπαράσταση του πλοίου που μελετάται σε σύστημα συντεταγμένων X-Y .	6
Εικόνα 2: Παράμετροι PID κατά την 1η δοκιμή.....	9
Εικόνα 3: Αποτελέσματα PID κατά την 1η δοκιμή.....	9
Εικόνα 4: Αυτόματη ρύθμιση PID μέσω Tune.....	10
Εικόνα 5: Παράμετροι PID που προέκυψαν μέσω Tune (2η προσπάθεια)	10
Εικόνα 6: Αποτελέσματα PID κατά τη 2η δοκιμή	11
Εικόνα 7: Αποτελέσματα PID κατά την 3η δοκιμή.....	11
Εικόνα 8: Μεταβλητές εισόδου και εξόδου του ελεγκτή FUZZY	13
Εικόνα 9: Εύρος τιμών και συναρτήσεις συμμετοχής γωνίας πεδαλίου (εξόδου).....	14
Εικόνα 10: Εύρος τιμών και συναρτήσεις συμμετοχής σφάλματος (1ης εισόδου).....	14
Εικόνα 11: Εύρος τιμών και συναρτήσεις συμμετοχής ρυθμού μεταβολής σφάλματος (2ης εισόδου).....	14
Εικόνα 12: Οι πρώτοι 17 κανόνες	15
Εικόνα 13: Οι τελευταίοι 8 κανόνες.....	16
Εικόνα 14: Αποτελέσματα FUZZY κατά την 1η δοκιμή	16
Εικόνα 15: Εύρος τιμών και συναρτήσεις συμμετοχής γωνίας πεδαλίου (εξόδου) κατά την 2η προσπάθεια	17
Εικόνα 16: Εύρος τιμών και συναρτήσεις συμμετοχής σφάλματος (1ης εισόδου) κατά την 2η προσπάθεια	17
Εικόνα 17: Εύρος τιμών και συναρτήσεις συμμετοχής ρυθμού μεταβολής σφάλματος (2ης εισόδου) κατά την 2η προσπάθεια.....	17
Εικόνα 18: Αποτελέσματα FUZZY κατά την 2η δοκιμή	18
Εικόνα 19: Αποτελέσματα FUZZY κατά την 3η δοκιμή	18
Εικόνα 20: Αποτελέσματα FUZZY κατά την 4η δοκιμή	19
Εικόνα 21: Αποτελέσματα FUZZY κατά την 5η δοκιμή	20

Εικόνα 22: Αποτελέσματα FUZZY κατά την 6η δοκιμή	21
Εικόνα 23: Αποτελέσματα μοντέλου με τους 2 ελεγκτές.....	22

III. Κατάλογος Σχημάτων

Σχήμα 1: Σχηματικό Διάγραμμα Μοντέλου Ελέγχου	7
Σχήμα 2: Μοντέλο Ελέγχου με PID στο SIMULINK.....	8
Σχήμα 3: Μοντέλο ελέγχου με FUZZY LOGIC στο SIMULINK	12
Σχήμα 4: Τελικό μοντέλο ελέγχου FUZZY στο SIMULINK	20
Σχήμα 5: Μοντέλο που περιέχει τους ελεγκτές PID και FUZZY.....	21

IV. Κατάλογος Πινάκων

Πίνακας 1: Πίνακας κανόνων.....	15
---------------------------------	----

1. Εισαγωγή

Σήμερα, η μηχανική ελέγχου (Control Engineering) διαδραματίζει θεμελιώδη ρόλο στα τεχνολογικά συστήματα. Για να σχεδιαστούν με ορθό και αποδοτικό τρόπο τα συστήματα ελέγχου (Control Systems), απαιτούνται οι κατάλληλες γνώσεις της θεωρίας ελέγχου (Control Theory), καθώς ο σχεδιασμός τους είναι μίας υψηλής πολυπλοκότητας διαδικασία. Η ρύθμιση και η βελτίωση των συστημάτων ελέγχου επιφέρει τεράστια οφέλη στις πρακτικές εφαρμογές αυτών, όπως για παράδειγμα στη βιομηχανία (Morawski *et al.*, 2006).

Όσον αφορά τον αυτόματο έλεγχο στα πλοία, είναι ένας κλάδος ο οποίος μελετάται εδώ και έναν αιώνα. Το 1911, κατασκευάστηκε ο πρώτος μηχανισμός αυτόματου τιμονιού από τον Elmer Sperry. Τα τελευταία χρόνια, η έρευνα έχει επεκταθεί από τον αυτόματο έλεγχο επανδρωμένων σκαφών μέχρι και μη-επανδρωμένα σκάφη, με τα Αυτόνομα Επιφανειακά Σκάφη (ASV) ή Μη-Επανδρωμένα Οχήματα Επιφάνειας (USV) να χρησιμοποιούνται σε ευρεία γκάμα εφαρμογών (Larrazabal & Peñas, 2016). Επιπλέον, το φάσμα θαλάσσιων σκαφών διευρύνεται και σε σκάφη όπως Τηλεχειριζόμενα Οχήματα (ROV) και ημιβυθισμένες εξέδρες. Συστήματα αυτομάτου ελέγχου όπως έλεγχος πλοήγησης, έλεγχος παρακολούθησης σημείου διαδρομής, έλεγχος απόσβεσης πηδαλίου και αγκυροβόλησης είναι πλέον εμπορικά προϊόντα (Lindegaard & Fossen, 2003). Οι κινητήριες δυνάμεις για την ανάπτυξη των παραπάνω συστημάτων αυτομάτου ελέγχου αποτελούν η αυξανόμενη ανάγκη για υπηρεσίες μεταφοράς και η απαίτηση υψηλότερων επιπέδων ασφαλείας.

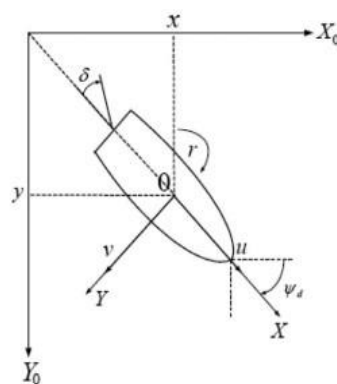
Προκειμένου να επιτευχθεί πιο αποδοτικός έλεγχος, καθώς και για να μειωθεί η ανθρώπινη παρέμβαση σε επικίνδυνα περιβάλλοντα, οι ελεγκτές πλοίων έχουν περάσει από μία μεγάλη τεχνολογική εξέλιξη. Ειδικότερα, από τους κλασσικούς PID ελεγκτές, έχει γίνει μετάβαση σε έναν πιο εξελιγμένο, ισχυρό και προσαρμοστικό έλεγχο με την τεχνητή νοημοσύνη να παίζει σημαντικό ρόλο. Οι ελεγκτές που αξιοποιούν τη τεχνητή νοημοσύνη, αποδεικνύονται πολύ χρήσιμοι για την αυτόματη πλοήγηση πλοίων, καθώς μπορούν να αντιμετωπίσουν τη μη-γραμμικότητα και την αβεβαιότητα που εμφανίζονται σε θαλάσσιες εφαρμογές. Μία περίπτωση αυτών των ελεγκτών είναι οι ασαφούς λογικής (Fuzzy Logic Controllers), οι οποίοι είναι μη-γραμμικοί ελεγκτές, και ενσωματώνοντας την τεχνογνωσία ενός χειριστή, καθίστανται αποτελεσματικοί για εφαρμογές όπως αυτές των Μη-Επανδρωμένων Οχημάτων Επιφάνειας (Larrazabal & Peñas, 2016).

2. Θεωρητικό Υπόβαθρο

Στο συγκεκριμένο κεφάλαιο, γίνεται επεξήγηση συγκεκριμένων όρων που συναντώνται κατά την έκταση της εργασίας. Ειδικότερα, ορίζεται και αναλύεται το πηδάλιο, καθώς και το πώς επηρεάζει την πορεία ενός πλοίου. Οι συγκεκριμένες πληροφορίες είναι αρκετά σημαντικές, καθώς αποτελούν βάση τόσο για τη μοντελοποίηση του προβλήματος στο κατάλληλο λογισμικό, όσο και για την ανάπτυξη των κανόνων στον ελεγκτή FUZZY.

2.1 Το πηδάλιο ενός πλοίου

Το πηδάλιο είναι ένα μέρος του σκάφους, το οποίο επιτρέπει τον έλεγχο της πορείας σε οριζόντιο επίπεδο, της διεύθυνσης, καθώς και του στριψίματος. Ο ρόλος του είναι να δημιουργήσει μία ροπή, η οποία στρέφει το σκάφος σε μία γωνία, εκτρέποντας το από την αρχική του πορεία. Αξίζει να σημειωθεί το γεγονός ότι το ίδιο το πηδάλιο είναι αυτό που θα στρέψει το σκάφος, λειτουργώντας ως σερβομηχανισμός, και όχι οι δυνάμεις που ασκούνται πάνω σε αυτό. Οι απαραίτητες δυνάμεις και ροπές δημιουργούνται υδροδυναμικά με τη ροή του νερού γύρω από το σκάφος, γεγονός που προκαλεί την συνολική ευστάθεια του σκάφους να μην εξαρτάται αποκλειστικά και μόνο από το πηδάλιο, αλλά και από το σχήμα της γάστρας. Όταν κατά την πορεία ο καπετάνιος στρίψει το τιμόνι, το φτερό του πηδαλίου παίρνει μία θέση, η οποία σχηματίζει κάποια γωνία με τη ροή του νερού γύρω από τη γάστρα (Ortsa, 2023). Είναι πολύ σημαντικό να αναφερθεί ότι η γωνία αυτή, είναι αντίθετη από την γωνία στριψίματος. Συγκεκριμένα, όταν η γωνία πλεύσης (πορείας) του πλοίου φτάσει στη καθορισμένη γωνία κατεύθυνσης, το πηδάλιο στρέφεται με τον μέγιστο ρυθμό προς την αντίθετη κατεύθυνση μέχρι να φτάσει την καθορισμένη γωνία πηδαλίου (Mofidi & Carrica, 2014). Αυτό, φαίνεται στην Εικόνα 1, όπου γίνεται αναπαράσταση της πορείας του υπό μελέτη πλοίου και οι γωνίες πλεύσης και πηδαλίου συμβολίζονται με ψ_d και δ , αντίστοιχα.



Εικόνα 1: Αναπαράσταση του πλοίου που μελετάται σε σύστημα συντεταγμένων X-Y

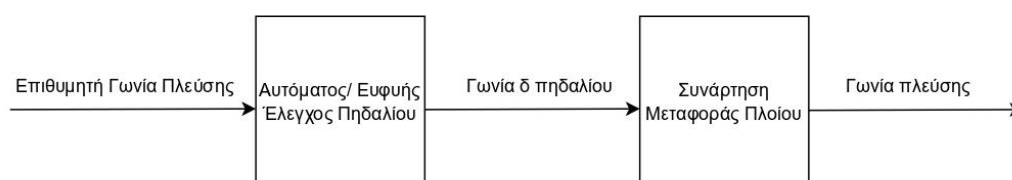
3. Το Μοντέλο

Προκειμένου να γίνει προσομοίωση και πρόβλεψη συμπεριφοράς ενός πραγματικού συστήματος, απαιτείται ένα επαρκές και αξιόπιστο μαθηματικό μοντέλο. Οι μαθηματικές αναπαραστάσεις, εκτός από υψηλές σε ακρίβεια, πρέπει να διεξαχθούν και με χαμηλό κόστος. Κατά συνέπεια, με τη χρήση των κατάλληλων συναρτήσεων που περιγράφουν τα υπό μελέτη συστήματα, οι ενέργειες ελέγχου μπορούν να προγραμματιστούν αποφεύγοντας τις δαπανηρές δοκιμές σε χρόνο και χρήμα. Στις περιπτώσεις οχημάτων και μεταφορικών μέσων όπως πλοία, χρησιμοποιούνται απλά μοντέλα, τα οποία αναπαριστούν την πραγματική συμπεριφορά τους, καθώς και τα δυναμικά χαρακτηριστικά τους, στα περισσότερα πρακτικά σενάρια (Larrazabal & Peñas, 2016).

Αναφορικά με το πλοίο που χρησιμοποιείται στη παρούσα εργασία, λαμβάνεται υπόψη ένα γραμμικό μοντέλο Wit Oppe του ναυτιλιακού πλοίου M/S Compass Island. Η συνάρτηση μεταφοράς η οποία συσχετίζει τη γωνία πηδαλίου δ (είσοδος) με την γωνία πλεύσης ψ (έξοδος), είναι η εξής (Nikończuk & Łozowicki, 2006):

$$T(s) = \frac{\psi}{\delta}(s) = \frac{2,0002}{s(1 + 0,563s)} = \frac{2,0002}{0,563s^2 + s}$$

Επιπροσθέτως, το ολοκληρωμένο μοντέλο υλοποιείται ως εξής: Ο χρήστης εισάγει μία είσοδο $\psi_1 \in [-10, 10]$ μοίρες η οποία αναπαριστά την επιθυμητή γωνία πλεύσης. Στη συνέχεια, μέσω ενός ελεγκτή, ρυθμίζεται η κατάλληλη γωνία πηδαλίου όπου εισάγεται ως είσοδος στην συνάρτηση μεταφοράς που αναφέρθηκε παραπάνω. Τέλος, ως έξοδος του ολικού συστήματος, προκύπτει η πραγματική γωνία πλεύσης του πλοίου ψ_2 . Μέσω ανάδρασης ρυθμίζεται κατάλληλα ο ελεγκτής, ο οποίος με βάση τη διαφορά της επιθυμητής και πραγματικής γωνίας πλεύσης $e = \psi_1 - \psi_2$ ρυθμίζει αυτόματα τη γωνία που θα στρίβει το πηδάλιο. Το παραπάνω σύστημα, περιγράφεται σχηματικά μέσω του διαγράμματος στο Σχήμα 1:



Σχήμα 1: Σχηματικό Διάγραμμα Μοντέλου Ελέγχου

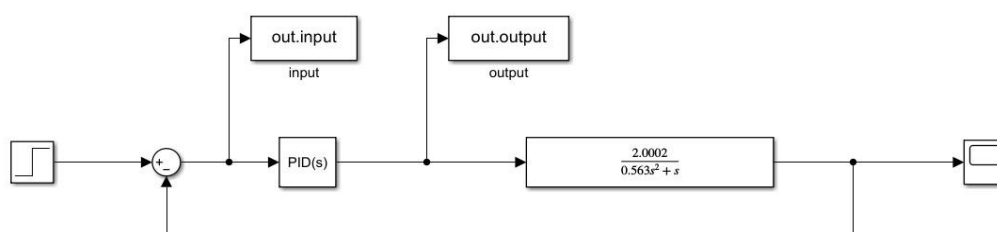
4. Υλοποίηση στο SIMULINK

Στο συγκεκριμένο κεφάλαιο, γίνεται η υλοποίηση του προβλήματος στο λογισμικό του SIMULINK. Ειδικότερα, θα υλοποιηθούν δύο μοντέλα, με το ένα εξ'αυτών να πραγματοποιεί έλεγχο μέσω ενός αυτόματου ελεγκτή PID, ενώ το άλλο, με τη χρήση ενός ευφυούς ελεγκτή FUZZY. Αφού γίνει η μοντελοποίηση, πραγματοποιείται ρύθμιση και προσαρμογή των παραμέτρων του εκάστοτε ελεγκτή, προκειμένου να βελτιωθούν και να συγκριθούν μεταξύ τους τα αποτελέσματα.

4.1 Υλοποίηση μέσω αυτόματου ελεγκτή PID

Ο PID, είναι ένας ευρέως χρησιμοποιούμενος ελεγκτής ο οποίος έχει αναρίθμητες εφαρμογές τόσο σε βιομηχανικό περιβάλλον όσο και σε διάφορα προβλήματα ελέγχου. Συγκεκριμένα, λειτουργεί μετρώντας διαρκώς τη τιμή του σφάλματος και προσπαθώντας να την ελαχιστοποιήσει μέσω τριών όρων, του Proportional (P), του Integral (I) και του Derivative (D). Αναλυτικότερα, ο P είναι ανάλογος της τρέχουσας τιμής της καμπύλης σφάλματος, και για παράδειγμα, αν το σφάλμα είναι μεγάλο, ο P θα λάβει μεγάλη τιμή μέσω του παράγοντα ενίσχυσης K_p . Επιπλέον, ο I ολοκληρώνει τις τιμές της καμπύλης σφάλματος, προσπαθώντας να εξαλείψει το υπολειπόμενο από τον P σφάλμα, ενώ ο D παραγωγίζει τις τιμές της καμπύλης σφάλματος, προσπαθώντας να εκτιμήσει τις μελλοντικές τιμές του σφάλματος. Μέσω διαδικασίας Tuning, ρυθμίζονται κατάλληλα οι παραπάνω όροι ώστε να πραγματοποιηθεί ο κατάλληλος έλεγχος ανάλογα τις απαιτήσεις της εφαρμογής (Araki, 2009).

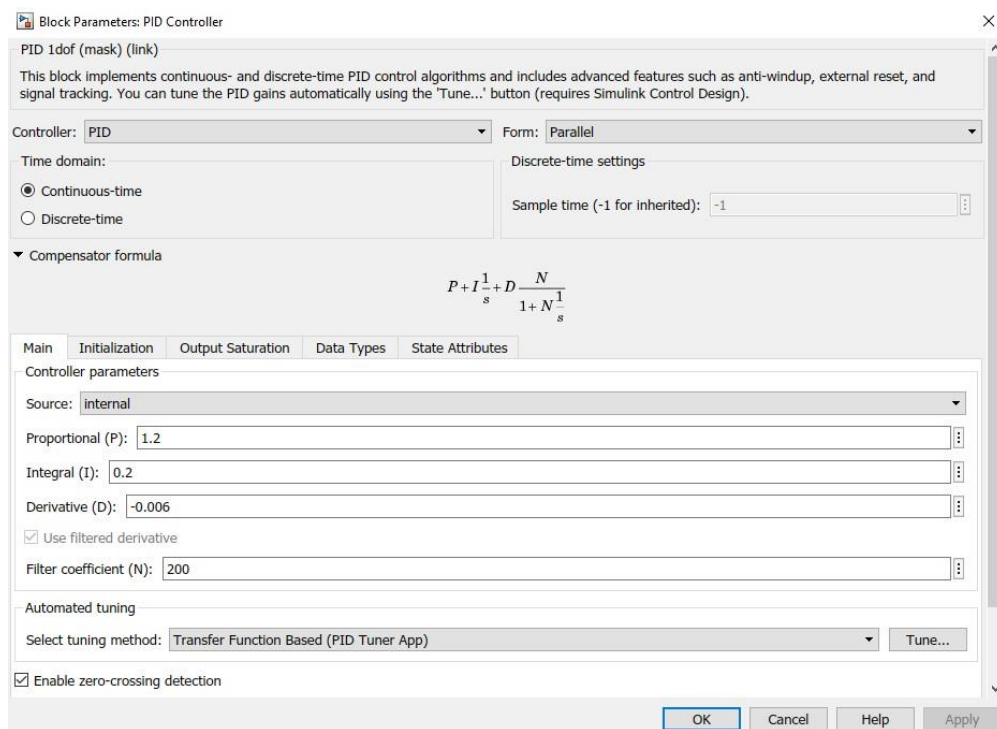
Για να υλοποιηθεί στο Simulink, ορίζεται είσοδος Step Function, ακολουθεί ο αφαιρετής εξόδου (για να υπολογιστεί το σφάλμα), ο ελεγκτής PID, η συνάρτηση μεταφοράς, η ανάδραση (δηλαδή η έξοδος) και ένα Scope για να αναπαρασταθεί γραφικά η έξοδος του συστήματος, όπως φαίνεται και στο σχήμα 2. Τα blocks out.input και out.output αφορούν τη μέτρηση τιμών στο συγκεκριμένο σημείο που έχουν τοποθετηθεί, προκειμένου να ληφθούν δεδομένα για ρύθμιση διαφορετικού ελεγκτή, σε επόμενη μοντελοποίηση.



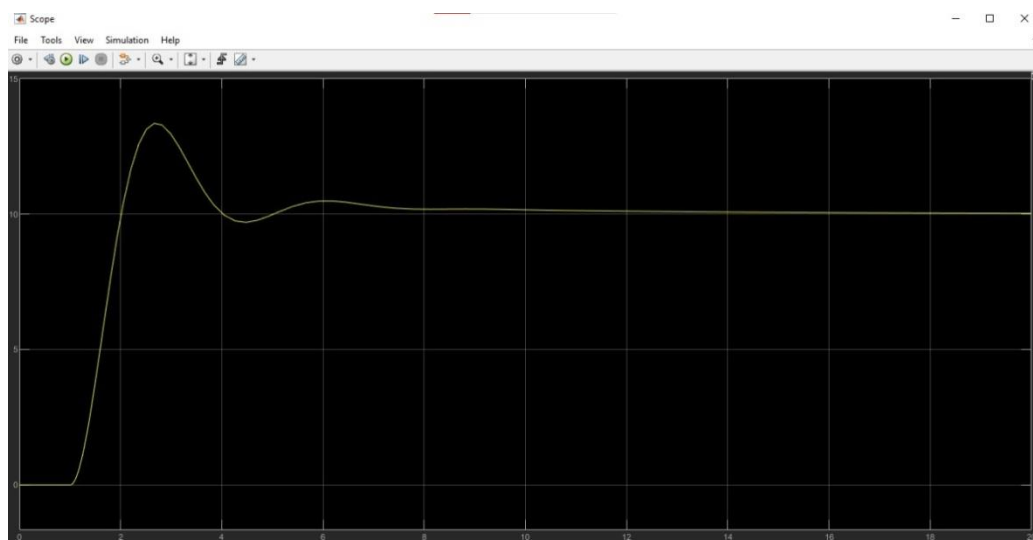
Σχήμα 2: Μοντέλο Ελέγχου με PID στο SIMULINK

Στόχος είναι η λήψη της αποδοτικότερης καμπύλης αποτελεσμάτων στο scope, αναφορικά με την ελαχιστοποίηση των ταλαντώσεων και την γρηγορότερη άφιξη στην επιθυμητή τιμή. Η επιθυμητή τιμή, ορίζεται στην είσοδο του συστήματος μέσω Step Function. Για τη συγκεκριμένη εφαρμογή, γίνονται δοκιμές για είσοδο $\psi_1 = 10^\circ$.

Αρχικά, πραγματοποιήθηκε δοκιμή για τιμές $P = 1.2$, $I = 0.2$, $D = -0.006$ και $N = 200$ (Εικόνα 2).



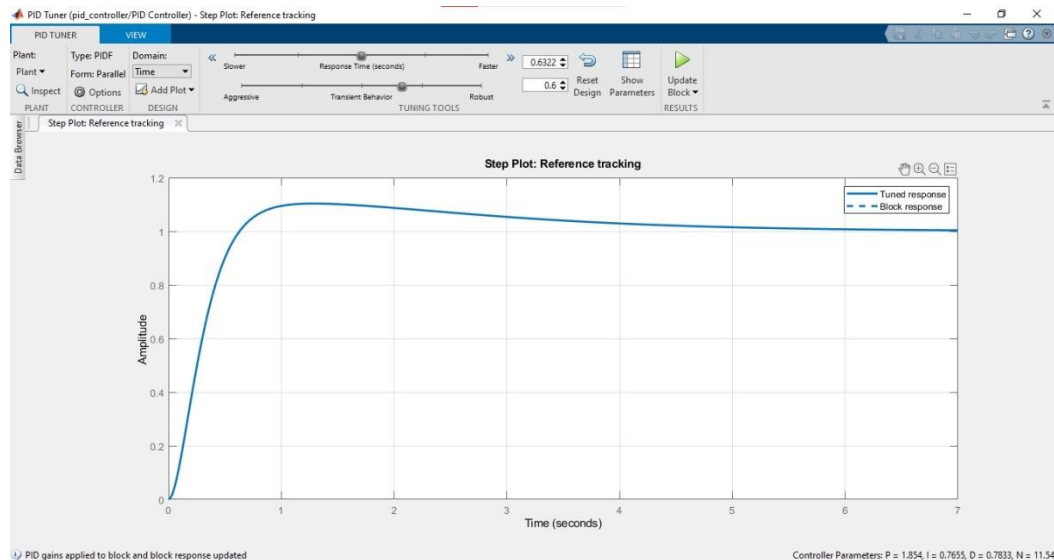
Εικόνα 2: Παράμετροι PID κατά την 1η δοκιμή



Εικόνα 3: Αποτελέσματα PID κατά την 1η δοκιμή

Παρατηρείται ότι τα συγκεκριμένα αποτελέσματα απέχουν από το ιδανικό, καθώς όχι μόνο υπάρχουν ταλαντώσεις, αλλά και η άφιξη στην επιθυμητή τιμή (10°) πραγματοποιείται σε 16 δευτερόλεπτα.

Σε αυτό το σημείο, γίνεται μετάβαση στην δεύτερη προσπάθεια βελτίωσης. Για τον ορισμό των κατάλληλων παραμέτρων του ελεγκτή PID, πραγματοποιείται αυτόματη ρύθμιση του μέσα από το εργαλείο Tune που παρέχεται από το λογισμικό SIMULINK, όπως φαίνεται και στην Εικόνα 4. Επιπλέον, οι παράμετροι που προέκυψαν φαίνονται στην Εικόνα 5, ενώ τα αποτελέσματα φαίνονται στην Εικόνα 6.



Εικόνα 4: Αυτόματη ρύθμιση PID μέσω Tune

Controller: PID Form: Parallel

Time domain:
☒ Continuous-time
☐ Discrete-time

Discrete-time settings
Sample time (-1 for inherited): -1

Compensator formula
$$P + I \frac{1}{s} + D \frac{N}{1 + N \frac{1}{s}}$$

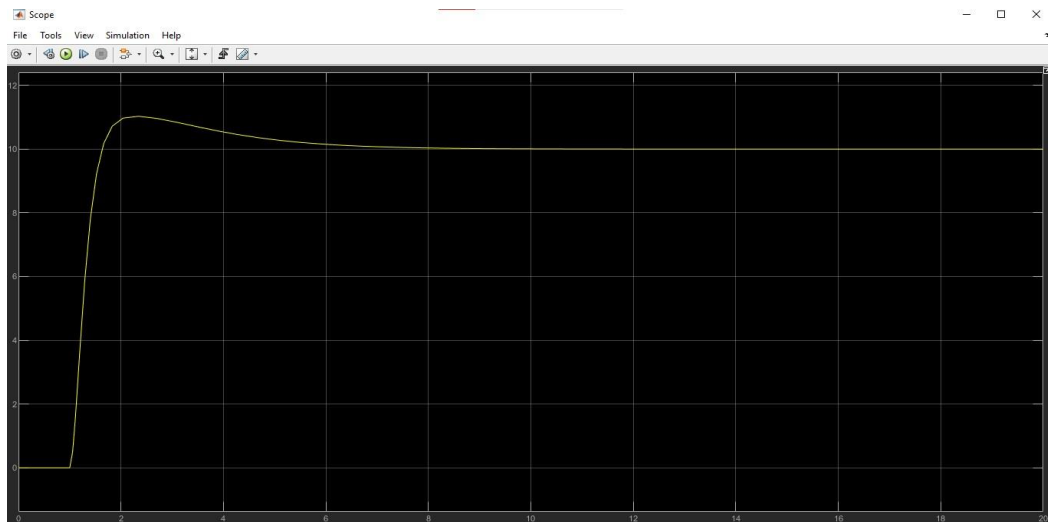
Main Initialization Output Saturation Data Types State Attributes

Controller parameters
Source: internal
Proportional (P): 1.85414532365193
Integral (I): 0.765545116444004
Derivative (D): 0.783327981990689
☒ Use filtered derivative
Filter coefficient (N): 11.5365013947656

Automated tuning
Select tuning method: Transfer Function Based (PID Tuner App) Tune...
☒ Enable zero-crossing detection

OK Cancel Help Apply

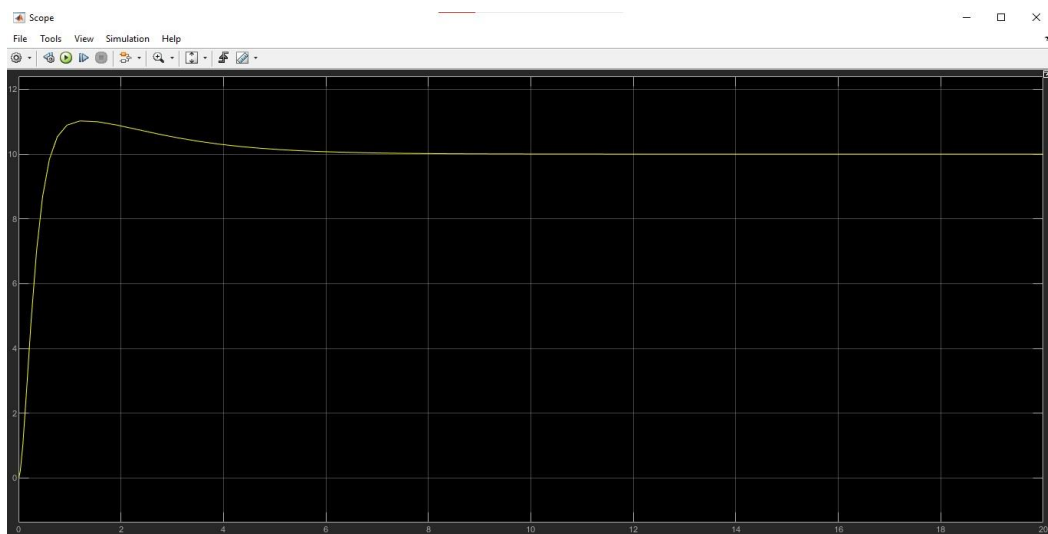
Εικόνα 5: Παράμετροι PID που προέκυψαν μέσω Tune (2η προσπάθεια)



Εικόνα 6: Αποτελέσματα PID κατά τη 2η δοκιμή

Παρατηρείται αξιοσημείωτη βελτίωση στην καμπύλη εξόδου του συστήματος, για τους ακόλουθους 2 λόγους. Πρώτον, οι ταλαντώσεις μειώθηκαν, γεγονός που καθιστά την καμπύλη πιο ομαλή, και δεύτερον, η άφιξη στην επιθυμητή τιμή (10°) πραγματοποιείται σε υποδιπλάσιο χρόνο (8 δευτερόλεπτα) συγκριτικά με την 1η δοκιμή. Συνεπώς, μετά από την αυτόματη ρύθμιση των παραμέτρων PID, το μοντέλο φτάνει στα επιθυμητά αποτελέσματα όχι μόνο πιο γρήγορα, αλλά και πιο ομαλά.

Μετά από διερεύνηση του μοντέλου στο SIMULINK, παρατηρήθηκε ότι υπάρχει μια επιπρόσθετη ρύθμιση προκειμένου να βελτιωθούν τα αποτελέσματα. Ειδικότερα, στην είσοδο (Step Function), η παράμετρος "Step Time" είναι ορισμένη στο 1, γεγονός που προκαλεί το ευθύγραμμο τμήμα της καμπύλης εξόδου κατά το 1ο δευτερόλεπτο της γραφικής αναπαράστασης. Οπότε, ορίζεται η παράμετρος "Step Time" σε 0, γεγονός που απαλείφει το συγκεκριμένο ευθύγραμμο τμήμα, μειώνοντας τον χρόνο άφιξης στην επιθυμητή τιμή στα 7 δευτερόλεπτα, όπως φαίνεται και στην Εικόνα 7.



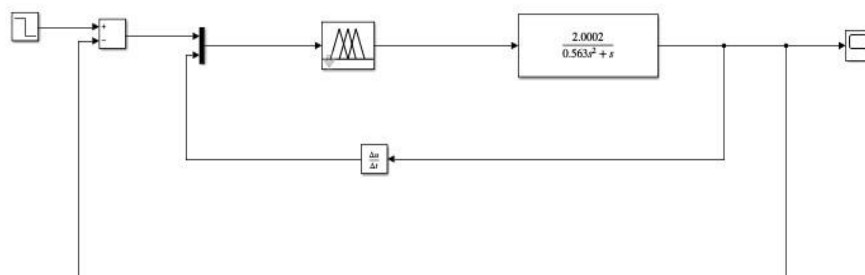
Εικόνα 7: Αποτελέσματα PID κατά την 3η δοκιμή

Συμπερασματικά, πραγματοποιήθηκε έλεγχος μέσω PID για τον αυτόματο πιλότο του πλοίου που εξετάζεται, με σταδιακή βελτίωση των αποτελεσμάτων. Η καλύτερη άφιξη στο επιθυμητό αποτέλεσμα καταγράφεται στα 7 δευτερόλεπτα, με τις ταλαντώσεις να έχουν μειωθεί σημαντικά συγκριτικά με την 1η προσπάθεια. Ακολουθεί μοντελοποίηση του συγκεκριμένου προβλήματος μέσω ευφυούς ελεγκτή FUZZY, προκειμένου να υπάρξει βελτίωση μεγαλύτερου επιπέδου στα αποτελέσματα.

4.2 Υλοποίηση μέσω ευφυούς ελεγκτή FUZZY

Ο ελεγκτής ασαφούς λογικής (Fuzzy Logic Controller), είναι ένας ελεγκτής βασισμένος στην τεχνητή νοημοσύνη. Λειτουργεί θέτοντας κανόνες, οι οποίοι βασίζονται σε αποφάσεις, αναλύοντας παράλληλα τη συμπεριφορά του συστήματος και τις γλωσσικές μεταβλητές (linguistic variables) στο πλαίσιο του συστήματος. Οι είσοδοι (inputs) του ελεγκτή ασαφούς λογικής επεξεργάζονται μέσω τριών σταδίων: Της ασαφοποίησης, της λήψης αποφάσεων και της αποασαφοποίησης πριν την έξοδο (output). Αναλυτικότερα, στο στάδιο ασαφοποίησης η μεταβλητή εισόδου μετατρέπεται σε γλωσσική μεταβλητή με τη βοήθεια προκαθορισμένων συναρτήσεων συμμετοχής (Membership Functions). Στη συνέχεια, μέσω κανόνων που έχουν οριστεί, πραγματοποιείται έξοδος από το στάδιο ασαφοποίησης και ξεκινάει το στάδιο της αποασαφοποίησης. Εκεί, η γίνεται η μετατροπή της ασαφοποιημένης εξόδου στην απαιτούμενη έξοδο που χρειάζεται για να ελεγχθεί το σύστημα. Συνεπώς, οι ελεγκτές ασαφούς λογικής έχουν ευρύ φάσμα εφαρμογών σε συστήματα που έχουν υψηλή αβεβαιότητα και μη-γραμμικότητα, γεγονός που καθιστά τους συγκεκριμένους ελεγκτές πολύπλοκους και με υψηλό κόστος υλοποίησης (Pandey *et al.*, 2022).

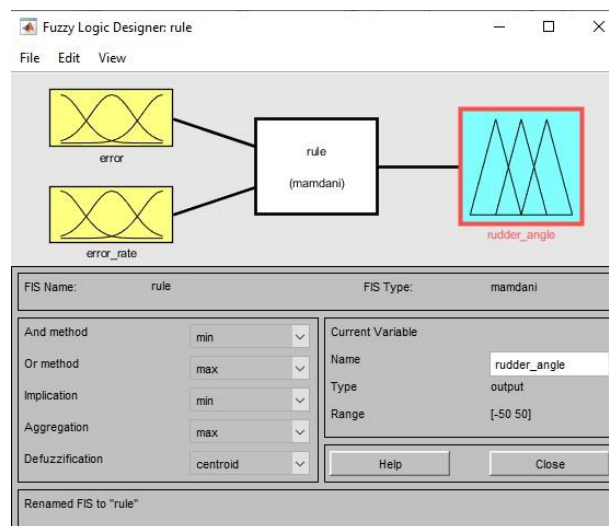
Για να υλοποιηθεί στο Simulink, ορίζεται είσοδος Step Function, ακολουθεί ο αφαιρετής εξόδου (για να υπολογιστεί το σφάλμα), ένας πολυπλέκτης MUX όπου ενώνει τα σήματα σφάλματος και μεταβολής σφάλματος (για να υπάρξουν 2 είσοδοι στην είσοδο του ελεγκτή), ο ελεγκτής FUZZY LOGIC, η συνάρτηση μεταφοράς, οι αναδράσεις σφάλματος και ρυθμού μεταβολής σφάλματος (έξοδοι) και ένα Scope για να αναπαρασταθεί γραφικά η έξοδος του συστήματος, όπως φαίνεται και στο Σχήμα 3.



Σχήμα 3: Μοντέλο ελέγχου με FUZZY LOGIC στο SIMULINK

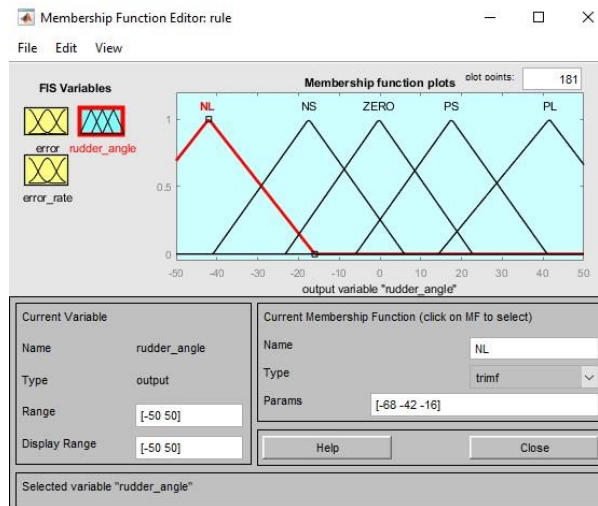
Στόχος είναι η λήψη αποδοτικότερης καμπύλης αποτελεσμάτων στο scope, συγκριτικά με τον PID. Η επιθυμητή τιμή, ορίζεται στην είσοδο του συστήματος μέσω Step Function. Για τη συγκεκριμένη εφαρμογή, γίνονται δοκιμές για είσοδο $\psi_1 = -10^\circ$. Ο λόγος για τον οποίο ορίζεται αρνητική τιμή εισόδου, είναι γιατί στα αποτελέσματα επιτυγχάνεται αντίθετη τιμή σε σχέση με την είσοδο. Ωστόσο, το γεγονός αυτό, βελτιώνεται σε μεταγενέστερο στάδιο κατά την υλοποίηση και τη βελτίωση του συστήματος.

Αρχικά, φορτώθηκε μέσω του MATLAB η εφαρμογή του FUZZY LOGIC. Ως προεπιλογή, υπάρχει μία είσοδος (input), το block κανόνων και η έξοδος (output). Για την υλοποίηση της εφαρμογής που πραγματεύεται η συγκεκριμένη εργασία, ο αριθμός των εισόδων αυξήθηκε σε 2, με τη μία είσοδο να αποτελεί το σφάλμα και η δεύτερη να αποτελεί τον ρυθμό μεταβολής του σφάλματος. Η έξοδος παρέμεινε μία, με αυτή να είναι η κατάλληλα ρυθμισμένη γωνία πηδαλίου (rudder angle), η οποία στη συνέχεια θα εισέλθει ως είσοδος στο block της συνάρτησης μεταφοράς. Ο ελεγκτής FUZZY, με τις μεταβλητές εισόδου και εξόδου του, φαίνεται στην Εικόνα 8.

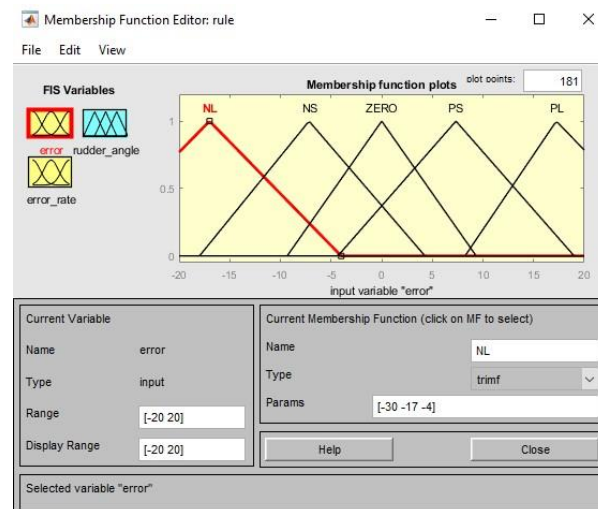


Εικόνα 8: Μεταβλητές εισόδου και εξόδου του ελεγκτή FUZZY

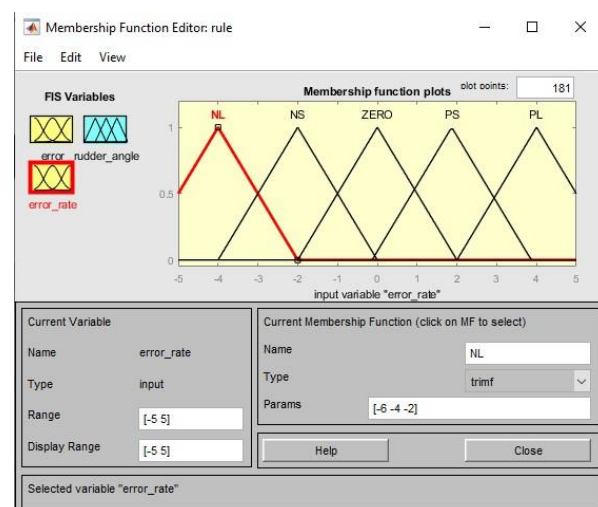
Κατά τη πρώτη δοκιμή, τα εύρη που τοποθετήθηκαν για την γωνία πηδαλίου (έξοδος ελεγκτή), το σφάλμα (1η είσοδος ελεγκτή) και του ρυθμού μεταβολής σφάλματος (2η είσοδος ελεγκτή) ήταν ίσα με $[-50, 50]$, $[-20, 20]$ και $[-5, 5]$, αντίστοιχα. Παράλληλα, το εύρος των εισόδων και της εξόδου απεικονίζεται σε ασαφή σύνολα επιθυμητών σχημάτων (συναρτήσεις συμμετοχής), τριγωνικής μορφής, όπως φαίνεται και στις Εικόνες 9,10 και 11. Επιπλέον, η έξοδος αποασαφοποιείται για να ληφθεί μια τελική τιμή του ελέγχου. Αυτή επιλέγεται με την μέθοδο του κεντροειδούς της περιοχής, διότι είναι η πιο συχνά χρησιμοποιούμενη μέθοδος αποασαφοποίησης. Η συγκεκριμένη μέθοδος, λειτουργεί δημιουργώντας το κέντρο βάρους της κατανομής για να δώσει την τελική τιμή της γωνίας πηδαλίου.



Εικόνα 9: Εύρος τιμών και συναρτήσεις συμμετοχής γωνίας πηδαλίου (εξόδου)



Εικόνα 10: Εύρος τιμών και συναρτήσεις συμμετοχής σφάλματος (1ης εισόδου)



Εικόνα 11: Εύρος τιμών και συναρτήσεις συμμετοχής ρυθμού μεταβολής σφάλματος (2ης εισόδου)

Επιπροσθέτως, ένα ασαφές σύστημα χαρακτηρίζεται από ένα σύνολο γλωσσικών δηλώσεων βασισμένων σε ειδική γνώση. Η γνώση αυτή, εκφράζεται συνήθως σε κανόνες τύπου "IF-THEN", οι οποίοι υλοποιούνται παίρνοντας την μορφή ασαφών συνθηκών. Όπως προαναφέρθηκε στο κεφάλαιο 2, όταν η γωνία πλεύσης του πλοίου φτάσει στη καθορισμένη γωνία κατεύθυνσης, το πηδάλιο στρέφεται με τον μέγιστο ρυθμό προς την αντίθετη κατεύθυνση μέχρι να φτάσει την καθορισμένη γωνία πηδαλίου. Συνεπώς, οι συγκεκριμένες γωνίες είναι αντίθετες. Το γεγονός αυτό, αποτέλεσε βάση δημιουργίας των κανόνων, οι οποίοι φαίνονται στις Εικόνες 12 & 13, και συνοψίζονται στον Πίνακα 1.

Αναλυτικότερα, για τον 1ο κανόνα, αν το σφάλμα (δηλαδή η διαφορά της επιθυμητής με την πραγματική γωνία πλεύσης) είναι μεγάλο και αρνητικό (π.χ. $\psi_1 - \psi_2 = 1 - 9 = -8$), αυτό σημαίνει ότι η επιθυμητή γωνία πλεύσης του πλοίου είναι πολύ πιο μικρή από αυτήν που όντως πλέει. Οπότε, για να έρθει το πλοίο στην επιθυμητή τροχιά πλεύσης, θα πρέπει να στρίψει το πηδάλιο κατά πολύ προς την αντίθετη μεριά από αυτήν που πλέει το πλοίο. Με ανάλογο τρόπο προέκυψαν και οι υπόλοιποι κανόνες.

Πίνακας 1: Πίνακας κανόνων

δ		e				
		NL	NS	ZO	PS	PL
de	PL	PL	NS	NL		NL
	PS		ZO	NS		
	ZO		PS	ZO	NS	
	NS		PS		ZO	
	NL		PL		PS	

1. If (error is NL) and (error_rate is NL) then (rudder_angle is PL) (1)
2. If (error is NL) and (error_rate is NS) then (rudder_angle is PL) (1)
3. If (error is NL) and (error_rate is ZERO) then (rudder_angle is PL) (1)
4. If (error is NL) and (error_rate is PS) then (rudder_angle is PL) (1)
5. If (error is NL) and (error_rate is PL) then (rudder_angle is PL) (1)
6. If (error is NS) and (error_rate is NL) then (rudder_angle is PL) (1)
7. If (error is NS) and (error_rate is NS) then (rudder_angle is PS) (1)
8. If (error is NS) and (error_rate is ZERO) then (rudder_angle is PS) (1)
9. If (error is NS) and (error_rate is PS) then (rudder_angle is ZERO) (1)
10. If (error is NS) and (error_rate is PL) then (rudder_angle is NS) (1)
11. If (error is ZERO) and (error_rate is NL) then (rudder_angle is PL) (1)
12. If (error is ZERO) and (error_rate is NS) then (rudder_angle is PS) (1)
13. If (error is ZERO) and (error_rate is ZERO) then (rudder_angle is ZERO) (1)
14. If (error is ZERO) and (error_rate is PS) then (rudder_angle is NS) (1)
15. If (error is ZERO) and (error_rate is PL) then (rudder_angle is NL) (1)
16. If (error is PS) and (error_rate is NL) then (rudder_angle is PS) (1)
17. If (error is PS) and (error_rate is NS) then (rudder_angle is ZERO) (1)

Εικόνα 12: Οι πρώτοι 17 κανόνες

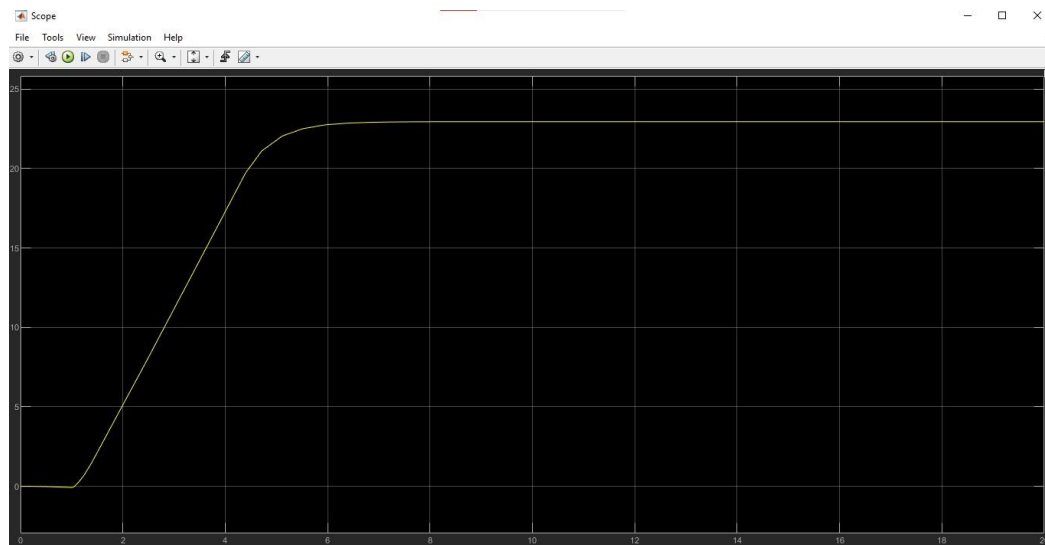

```

18. If (error is PS) and (error_rate is ZERO) then (rudder_angle is NS) (1)
19. If (error is PS) and (error_rate is PS) then (rudder_angle is NS) (1)
20. If (error is PS) and (error_rate is PL) then (rudder_angle is NL) (1)
21. If (error is PL) and (error_rate is NL) then (rudder_angle is NL) (1)
22. If (error is PL) and (error_rate is NS) then (rudder_angle is NL) (1)
23. If (error is PL) and (error_rate is ZERO) then (rudder_angle is NL) (1)
24. If (error is PL) and (error_rate is PS) then (rudder_angle is NL) (1)
25. If (error is PL) and (error_rate is PL) then (rudder_angle is NL) (1)

```

Εικόνα 13: Οι τελευταίοι 8 κανόνες

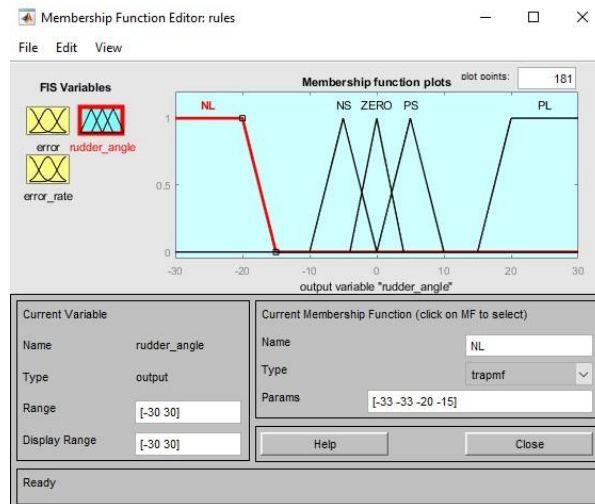
Σε αυτό το σημείο, πραγματοποιείται η 1η δοκιμή για τον συγκεκριμένο ελεγκτή, χρησιμοποιώντας τις εισόδους, εξόδους, συναρτήσεις συμμετοχής και κανόνες που φαίνονται στις Εικόνες 8-13, και τα αποτελέσματα που προέκυψαν φαίνονται στην Εικόνα 14.



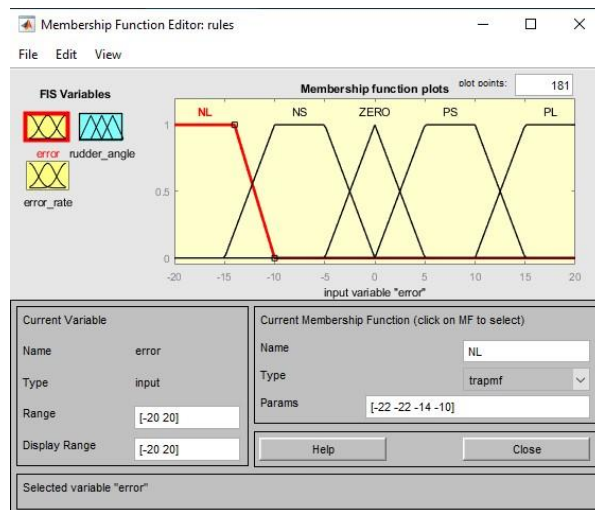
Εικόνα 14: Αποτελέσματα FUZZY κατά την 1η δοκιμή

Παρατηρείται ότι τα συγκεκριμένα αποτελέσματα απέχουν από το ιδανικό, καθώς δεν επιτυγχάνεται η επιθυμητή τιμή (10°), γεγονός που σημαίνει ότι απαιτείται βελτίωση.

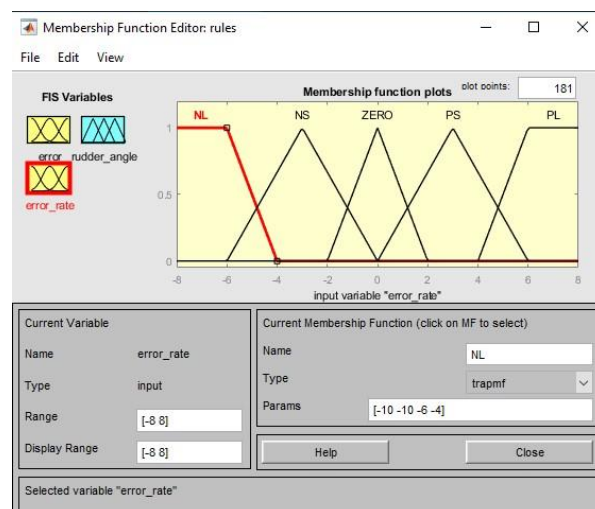
Κατά την 2η δοκιμή, πραγματοποιήθηκαν αλλαγές στις συναρτήσεις συμμετοχής, καθώς στα εύρη τους. Συγκεκριμένα, στην έξοδο και στο σφάλμα χρησιμοποιήθηκαν τραπεζοειδείς συναρτήσεις αντί για τριγωνικές στις ακραίες τιμές, ενώ στον ρυθμό μεταβολής χρησιμοποιήθηκαν τραπεζοειδείς για όλες τις τιμές, εκτός από αυτές κοντά στο μηδέν, στις οποίες παρέμεινε τριγωνική. Επιπλέον, το εύρος εξόδου μειώθηκε σε $[-30 \ 30]$, το εύρος ρυθμού μεταβολής σφάλματος αυξήθηκε σε $[-8 \ 8]$ και το εύρος του σφάλματος παρέμεινε το ίδιο. Οι προαναφερόμενες αλλαγές, φαίνονται στις Εικόνες 15-17. Τα εύρη τιμών των συναρτήσεων συμμετοχής για το σφάλμα και τον ρυθμό μεταβολής σφάλματος παρέμειναν ίδια, ενώ στην γωνία πηδαλίου "συμπυκνώθηκαν" οι συναρτήσεις συμμετοχής για τιμές γύρω από το 0.



Εικόνα 15: Εύρος τιμών και συναρτήσεις συμμετοχής γωνίας πηδαλίου (εξόδου) κατά την 2η προσπάθεια

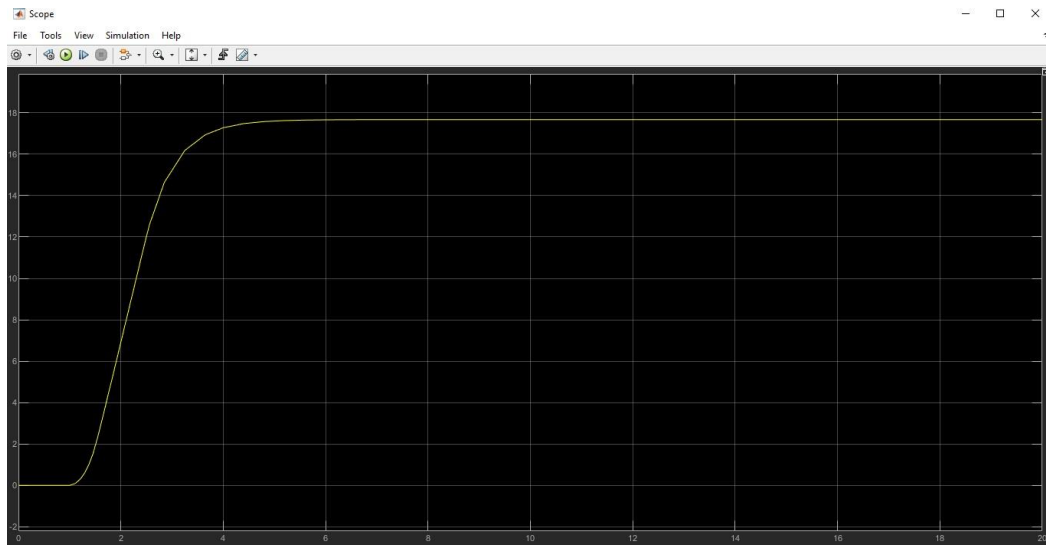


Εικόνα 16: Εύρος τιμών και συναρτήσεις συμμετοχής σφάλματος (1ης εισόδου) κατά την 2η προσπάθεια



Εικόνα 17: Εύρος τιμών και συναρτήσεις συμμετοχής ρυθμού μεταβολής σφάλματος (2ης εισόδου) κατά την 2η προσπάθεια

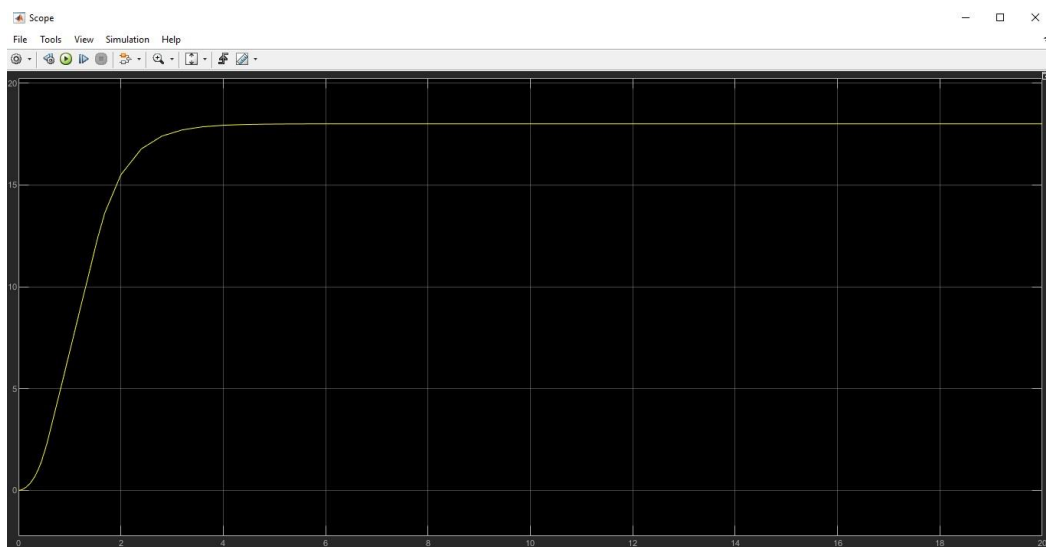
Συνεπώς, πραγματοποιείται η 2η δοκιμή για τον συγκεκριμένο ελεγκτή, χρησιμοποιώντας τις εισόδους, εξόδους, συναρτήσεις συμμετοχής και κανόνες που φαίνονται στις Εικόνες 15-17, και τα αποτελέσματα που προέκυψαν φαίνονται στην Εικόνα 18.



Εικόνα 18: Αποτελέσματα FUZZY κατά την 2η δοκιμή

Παρατηρείται αξιοσημείωτη βελτίωση στην καμπύλη εξόδου του συστήματος, καθώς όχι μόνο είναι πιο ομαλή η καμπύλη συγκριτικά με την 1η προσπάθεια, αλλά και η καμπύλη φτάνει σε τιμή πλησιέστερη στην επιθυμητή.

Κατά την 3η προσπάθεια, ορίζεται η παράμετρος "Step Time" σε 0, γεγονός που απαλείφει το συγκεκριμένο ευθύγραμμο τμήμα που προκαλείται λόγω του προκαθορισμένου Step Time = 1 μειώνοντας τον χρόνο άφιξης στην τελική τιμή κατά 1 δευτερόλεπτο, όπως φαίνεται και στην Εικόνα 19.



Εικόνα 19: Αποτελέσματα FUZZY κατά την 3η δοκιμή

Στην 4η προσπάθεια, πραγματοποιήθηκαν δοκιμές μέσω μικρών μεταβολών (της τάξεως των 1-2 μονάδων) σε όλες τις μεταβλητές. Παρατηρήθηκε ότι αλλάζοντας σε μικρό βαθμό τη γωνία πηδαλίου, τόσο επεκτείνοντας, όσο και μικραίνοντας το διάστημα, δεν υπήρξε κάποια αξιοσημείωτη βελτίωση. Ωστόσο, στο σφάλμα, παρατηρήθηκε ότι μικραίνοντας το, άρχισε να προσεγγίζει η καμπύλη την επιθυμητή τιμή, με την κλίση παράλληλα να γίνεται πιο απότομη (κατακόρυφη). Πρώτα πρώτα, μεταβλήθηκε από $[-20,20]$ σε $[-18, 18]$ και η καμπύλη έπεσε από τις 18° στο 16° . Μετά από επιπλέον δοκιμές, παρατηρήθηκε ότι το διάστημα $[-13, 13]$ ήταν το καλύτερο, αλλά όχι ιδανικό, καθώς η καμπύλη έπεσε ελαφρά κάτω από 10° (Εικόνα 20). Παρόλα αυτά, επιλέχθηκαν αυτές οι τιμές ως οι καλύτερες, καθώς ήταν οι πλησιέστερες στις 10° .



Εικόνα 20: Αποτελέσματα FUZZY κατά την 4η δοκιμή

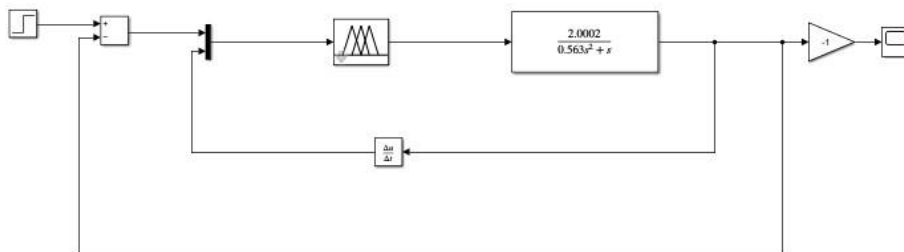
Κατά την 5η προσπάθεια, πραγματοποιήθηκε αλλαγή στον ρυθμό μεταβολής σφάλματος. Ο λόγος που έγινε αυτό, είναι ότι έχοντας πλέον μικρότερο σφάλμα, το σύστημα πλησιάζει πιο κοντά στην επιθυμητή κατάσταση, και άρα, ο ρυθμός μεταβολής του σφάλματος μπορεί επίσης να μειωθεί. Επειδή η καμπύλη είναι αρκετά κοντά στις 10° , η μείωση του εύρους του ρυθμού μεταβολής σφάλματος ήταν της τάξεως των 0.2 μονάδων δηλαδή από $[-8 \ 8]$ πήγε σε $[-7.8 \ 7.8]$, όπως φαίνεται και στην Εικόνα 21.

Έτσι, πραγματοποιήθηκε αξιοσημείωτη βελτίωση του ελεγκτή FUZZY, με το σύστημα πλέον να είναι πολύ καλύτερο συγκριτικά με την 1η δοκιμή ρύθμισης του ελεγκτή. Τα αποτελέσματα της 5ης προσπάθειας βελτίωσης φαίνονται στην Εικόνα 22, όπου η καμπύλη έχει ομαλή μορφή και καταφθάνει στις 10° σε χρόνο τεσσάρων δευτερολέπτων.



Εικόνα 21: Αποτελέσματα FUZZY κατά την 5η δοκιμή

Τέλος, γίνεται και μία επιπρόσθετη βελτίωση στο σύστημα, η οποία το καθιστά πιο ολοκληρωμένο και φιλικότερο προς τον χρήστη, μέσω ρύθμισης της εισόδου του. Όπως προαναφέρθηκε, είχε οριστεί αρνητική τιμή βηματικής εισόδου (Final Value = -10), καθώς στα αποτελέσματα επιτυγχάνεται αντίθετη τιμή σε σχέση με την είσοδο. Ωστόσο, το γεγονός αυτό μπορεί να αλλάξει προσθέτοντας ένα Block "Gain" στην έξοδο, το οποίο παίρνει τιμή -1. Έτσι, μπορεί να εισαχθεί πλέον η πραγματική επιθυμητή τιμή (10), και όχι η αντίθετη (-10). Το νέο μοντέλο φαίνεται στο Σχήμα 4.



Σχήμα 4: Τελικό μοντέλο ελέγχου FUZZY στο SIMULINK

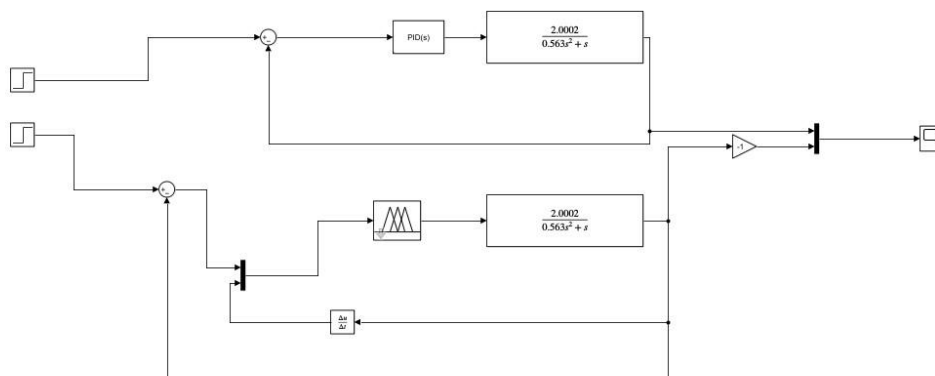
Με την προσθήκη του Gain, πραγματοποιήθηκε μεταβολή της καμπύλης του αποτελέσματος, λίγο πάνω από τις 10° . Επομένως, ξαναρυθμίστηκαν οι τιμές από τα εύρη του σφάλματος και του ρυθμού μεταβολής σφάλματος, σε $[-13.2, 13.2]$ και $[-7.7, 7.7]$, αντίστοιχα. Με αυτή την αλλαγή, επιτυγχάνονται ξανά οι 10° , και πραγματοποιείται μετάβαση στη σύγκριση του ελεγκτή FUZZY με τον ελεγκτή PID. Τα αποτελέσματα της τελευταίας βελτίωσης (6η προσπάθεια) φαίνονται στην Εικόνα 22.



Εικόνα 22: Αποτελέσματα FUZZY κατά την 6η δοκιμή

4.3 Σύγκριση των 2 ελεγκτών

Σε αυτό το σημείο, έχοντας υλοποιήσει τους ελεγκτές PID και FUZZY, μπορούμε να τους συγκρίνουμε μεταξύ τους. Για να πραγματοποιηθεί αυτό, τοποθετούνται σε κοινό περιβάλλον στο SIMULINK, όπου μπορούν να συγκριθούν οι καμπύλες αποτελεσμάτων μέσω κοινού Scope, όπως φαίνεται και στο μοντέλο στο Σχήμα 5.



Σχήμα 5: Μοντέλο που περιέχει τους ελεγκτές PID και FUZZY

Αξίζει να σημειωθεί το γεγονός ότι αφού πραγματοποιήθηκε η τοποθέτηση των ελεγκτών σε κοινό περιβάλλον, επηρεάστηκαν τα αποτελέσματα του ελεγκτή FUZZY. Οπότε, ρυθμίστηκε κατάλληλα, με τα τελικά εύρη τιμών της γωνίας πεδαλίου, του σφάλματος, και του ρυθμού μεταβολής σφάλματος να είναι $[-30, 30]$, $[-13.3, 13.3]$ και $[-8, 8]$, αντίστοιχα. Τα αποτελέσματα φαίνονται στην Εικόνα 23, με τη κίτρινη καμπύλη να είναι η έξοδος του μοντέλου PID, ενώ η μπλε είναι η έξοδος του μοντέλου FUZZY.



Εικόνα 23: Αποτελέσματα μοντέλου με τους 2 ελεγκτές

Παρατηρείται ότι τα αποτελέσματα του ελεγκτή FUZZY είναι καλύτερα από τον ελεγκτή PID, τόσο στο κομμάτι της ομαλότητας της καμπύλης, όσο και στον χρόνο άφιξης στην επιθυμητή τιμή. Ειδικότερα, η καμπύλη αποτελεσμάτων του μοντέλου που περιέχει τον ελεγκτή PID φτάνει στις 10° μέσα σε 7 δευτερόλεπτα, ενώ η καμπύλη αποτελεσμάτων του μοντέλου που περιέχει τον ελεγκτή FUZZY επιτυγχάνει αυτό το αποτέλεσμα μέσα σε 2.8 δευτερόλεπτα.

5. Συμπεράσματα

Συμπερασματικά, η Μηχανική Ελέγχου είναι ένας ταχέως αυξανόμενος κλάδος, ο οποίος βρίσκει εφαρμογή σε πληθώρα τεχνολογικών συστημάτων. Από τους κλασσικούς αυτόματους ελεγκτές, έχει γίνει μετάβαση σε ευφυείς ελεγκτές, οι οποίοι χρησιμοποιούν τεχνητή νοημοσύνη, προκειμένου γίνει διαχείριση πολύπλοκων συστημάτων με αποδοτικότερο τρόπο. Αναφορικά με τα συστήματα ελέγχου στα πλοία, σημαντικό αποτελεί το γεγονός κατανόησης της λειτουργίας των παραμέτρων που επηρεάζουν την πορεία του. Για την μοντελοποίηση του προβλήματος που πραγματεύεται η συγκεκριμένη εργασία, χρησιμοποιήθηκε το λογισμικό SIMULINK που παρέχεται από το MATLAB. Εκεί, έγινε προσομοίωση ελέγχου της γωνίας πλεύσης πλοίου μέσω της ρύθμισης της γωνίας πηδαλίου, τόσο μέσω ελεγκτή PID, όσο και μέσω ελεγκτή FUZZY. Παρατηρήθηκε ότι με την ρύθμιση και την βελτίωση των κατάλληλων παραμέτρων μέσω αλλαγών, ο FUZZY επιφέρει καλύτερα αποτελέσματα σε ομαλότητα και χρόνο επίτευξης επιθυμητής τιμής, γεγονός λογικό, αφού είναι ένας πιο εξελιγμένος ελεγκτής συγκριτικά με τον PID. Τέλος, μπορεί να πραγματοποιηθεί μελλοντική έρευνα πάνω στο συγκεκριμένο πρόβλημα. Συγκεκριμένα, μπορούν να διερευνηθούν και να εφαρμοστούν επιπρόσθετες τεχνικές που αξιοποιούν την τεχνητή νοημοσύνη για τις ρυθμίσεις των παραμέτρων των ελεγκτών, έτσι ώστε να επιτευχθούν αποδοτικότερα οι ζητούμενες έξοδοι των συστημάτων.

6. Βιβλιογραφικές Αναφορές

Araki, M. (2009). "Control Systems, Robotics and Automation-Volume VII-PID Control." Kyoto University, Japan.

Larrazabal, J. M., & Peñas, M. S. (2016). "Intelligent rudder control of an unmanned surface vessel." *Expert Systems with Applications*, Vol. 55, pp. 106-117.

Lindegaard, K. P., & Fossen, T. I. (2003). "Fuel-efficient rudder and propeller control allocation for marine craft: Experiments with a model ship." *IEEE Transactions on Control Systems Technology*, Vol. 11, No. 6, pp. 850-862.

Mofidi, A., & Carrica, P. M. (2014). "Simulations of zigzag maneuvers for a container ship with direct moving rudder and propeller." *Computers & fluids*, Vol. 96, pp. 191-203.

Morawski, L., Pomirski, J., & Rak, A. (2006). "Design of the ship course control system." DS 36: Proceedings DESIGN 2006, the 9th International Design Conference, Dubrovnik, Croatia.

Nikończuk, P., & Łozowicki, A. (2006). "Evolutionary algorithms application in a ship autopilot system with optimal controller." 7th IFAC Conference on Manoeuvring and Control of Marine Craft Lisbon, Portugal.

Ortsa.gr (2023), "Το πηδάλιο του σκάφους και πως λειτουργεί". <https://rb.gy/1dhr7o> (Ημερ. θέασης 09/01/2024).

Pandey, A. K., Singh, V., & Jain, S. (2022). "Study and comparative analysis of perturb and observe (P&O) and fuzzy logic based PV-MPPT algorithms." *Applications of AI and IOT in Renewable Energy*, pp. 193-209.