Modeling and Control of an Autonomous Sailboat

Ben Bruick
Robotics Department
University of Michigan
Ann Arbor, MI, USA
bruick@umich.edu

Kavin M. Govindarajan
Robotics Department
University of Michigan
Ann Arbor, MI, USA
kmgovind@umich.edu

Matt Prince
Robotics Department
University of Michigan
Ann Arbor, MI, USA
mattpri@umich.edu

***Abstract*—This paper presents the modeling and control of an autonomous sailboat designed for long-duration missions. The proposed model incorporates aerodynamic forces using a NACA airfoil for the sail and accounts for time-varying wind conditions. Additionally, the dynamic model includes the influence of ocean currents. We present a blended LQR-Proportional control algorithm to navigate the sailboat to a set of waypoints, which is coupled with a replanning algorithm to avoid unreachable waypoints. This is compared with a controller learned from a genetic algorithm. The effectiveness of the model and control approach is validated through simulations. [Code] [1]**

I. INTRODUCTION

The field of autonomous maritime vehicles has seen significant advancements in recent years, driven by the need for innovative solutions in transportation, environmental monitoring, and oceanographic research [1]. Among the various forms of autonomous vessels, sailboats present a unique set of challenges and opportunities because of their reliance on wind propulsion and complex navigation requirements [2]. Autonomous sailboats, which use the power of wind, offer a sustainable and energy-efficient alternative that can operate over long durations without fuel dependency [3]. This paper focuses on the modeling and control of autonomous sailboats, aiming to contribute to the growing body of knowledge in nautical autonomy and control systems.

The development of an autonomous sailboat involves intricate challenges in accurately modeling the dynamic interactions between the vessel and its environment, including wind, water currents, and waves [4] [5]. These interactions must be captured through robust mathematical models that can accommodate the inherently nonlinear and time-varying nature of marine environments [4]. In addition, sophisticated control strategies are essential to ensure optimal sailboat performance, allowing it to navigate complex routes while adapting to changing environmental conditions. The interplay between predictive modeling and adaptive control constitutes a successful autonomous sailboat system [2].

This paper seeks to address these challenges by presenting comprehensive models that describe the dynamic behavior of sailboats and by proposing advanced control algorithms designed to optimize navigational efficiency and route stability. We draw on interdisciplinary methodologies that incorporate principles from fluid dynamics, control theory, and artificial intelligence. Through simulation and real-world testing, we demonstrate the efficacy of the proposed models and control strategies, paving the way for broader applications and future enhancements in autonomous maritime technology.

By exploring the intricacies of autonomous sailboat modeling and control, this work contributes to the vision of sustainable and intelligent marine operations, offering a pathway towards reducing the carbon footprint of marine activities and enhancing the autonomy of global maritime systems.

[1]Code: https://github.com/kmgovind/auto-sailboat

Fig. 1: Åland sailing boat used as reference [6]

II. MODELING

We consider a vehicle similar to that shown in Fig. 1. We follow the axis definitions shown in Fig. 2. This vehicle has the attributes described in Table I.

Parameter	Estimated Value
Length (LOA)	4.5 – 5.5 m
Beam (B)	1.4 – 1.7 m
Draft (D)	0.6 – 1.0 m
Sail Area (S)	5 – 8 m²
Displacement	300 – 350 kg
Waterline Length (LWL)	4.0 – 4.5 m
Sail/Rudder Foil	NACA 0015

TABLE I: Estimated Dimensions of the Sailboat

We make the following simplifying assumptions:

- Motion is restricted to a horizontal plane at sea level. Thus, the vehicle is limited to surge, sway, and yaw (reference Fig.2).
- The influence of waves shall be neglected. The only external forces that shall act upon the vehicle are forces from the wind and surface currents.

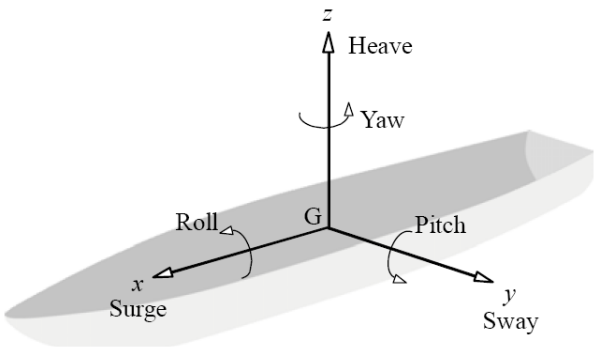


Fig. 2: Boat-fixed frame definition and motions

- We assume the velocity of the vehicle to be sufficiently small. Therefore, Coriolis and centripetal forces are neglected.

A. Equations of Motion

We base our model upon the model presented by Melin [5]. We define two frames: an earth-fixed North-East-Up frame that is assumed to be inertial with an origin defined by the boat's start point and a boat-fixed frame such that the origin lies at the boat's center of gravity.

We define a five-state model given by $\mathbf{x} = \begin{bmatrix} x & y & \theta & v & \omega \end{bmatrix}^\intercal$ where x, y represent the sailboat position, θ represents the heading, and ω represents the boat's rotational speed in the North-East-Up frame. v represents the boat's velocity in the boat-fixed frame. We define the control signals of the boat to be $\mathbf{u} = \begin{bmatrix} \delta_r & \delta_s \end{bmatrix}^\intercal$ where δ_r is the angle of the rudder and δ_s is the angle of the sail.

The system dynamics are given by:

$$\dot{x} = \begin{bmatrix} v\cos(\theta) + p_1 a_{tw}\cos(\psi_{tw}) + v_{cx} \\ v\sin(\theta) + p_1 a_{tw}\sin(\psi_{tw}) + v_{cy} \\ 0.5\omega \\ \frac{g_s - g_r p_{11} - p_2 v^2}{p_9} \\ \frac{g_s(p_6 - p_7\cos(\delta_s)) - g_r p_8\cos(\delta_r) - p_3\omega v}{p_{10}} \end{bmatrix} \tag{1}$$

where p_i is defined in Table II and where the sail and rudder force contributions are defined as:

$$g_s = (F_{x\_lift} - F_{x\_drag})\sin(\delta_s - \psi_{aw}) \tag{2}$$

$$g_r = D\sin(\delta_r) \tag{3}$$

with the following variables:

- v_{cx}, v_{cy} are the ocean current components.
- F_{x_lift}, F_{x_drag} are the boat-aligned lift and drag components from the sail.
- D is the drag force from the apparent wind.

B. Wind Model

To accurately simulate the environmental forces acting on the sailboat, we implement a high-fidelity wind model that accounts for gusts, turbulence, and long-term wind drift [7] [8]. The wind velocity is represented as a time-varying vector, incorporating both magnitude and direction changes.

Parameter	Value	Description
p_1	0.03	Drift coefficient
p_2	40 kg/s	Tangential friction
p_3	6000 kg·m	Angular friction
p_4	200 kg/s	Sail lift
p_5	1500 kg/s	Rudder lift
p_6	0.5 m	Distance to sail CoE
p_7	0.5 m	Distance to mast
p_8	2 m	Distance to rudder
p_9	300 kg	Mass of boat
p_{10}	400 kg·m^2	Moment of inertia
p_{11}	0.2	Rudder brake coefficient

TABLE II: Parameter values and descriptions

1) Wind Speed Dynamics: The wind speed $W(t)$ at time t is modeled as the sum of a base wind strength W_0 and three dynamic components: gusts, turbulence, and long-term drift:

$$W(t) = W_0 + W_g(t) + W_{\text{turb}}(t) + W_d(t). \tag{4}$$

The gust component $W_g(t)$ is a sinusoidal fluctuation:

$$W_g(t) = A_g\sin(2\pi f_g t), \tag{5}$$

where A_g is the maximum gust amplitude and f_g is the gust frequency.

The turbulence component $W_{\text{turb}}(t)$ introduces small stochastic variations:

$$W_{\text{turb}}(t) = A_{\text{turb}}(\xi - 0.5), \tag{6}$$

where A_{turb} is the turbulence amplitude and ξ is a normally distributed random variable.

The long-term wind drift $W_d(t)$ represents slow variations in wind speed:

$$W_d(t) = A_d\sin(2\pi f_d t), \tag{7}$$

where A_d is the drift amplitude and f_d is the drift frequency.

2) Wind Direction Dynamics: The wind direction $\theta_W(t)$, expressed in radians, evolves as:

$$\theta_W(t) = \theta_0 + \theta_d(t) + \theta_{\text{turb}}(t). \tag{8}$$

Here, θ_0 is the initial wind direction, $\theta_d(t)$ models slow directional drift, and $\theta_{\text{turb}}(t)$ introduces small random fluctuations. The drift component is given by:

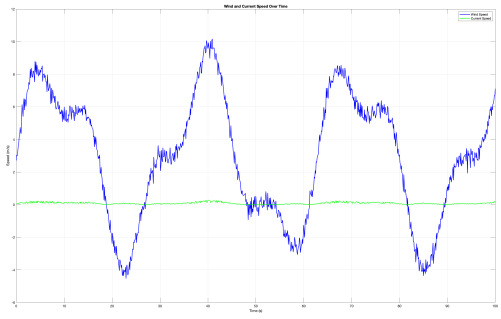
$$\theta_d(t) = A_\theta\sin(2\pi f_\theta t), \tag{9}$$

where A_θ is the maximum deviation and f_θ is the frequency of directional drift.

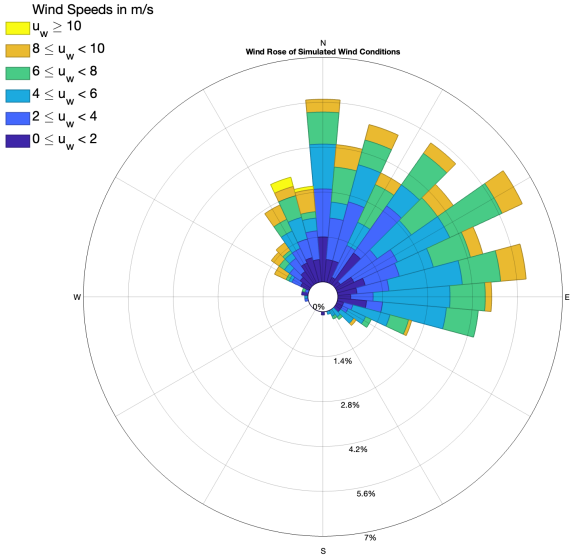
The turbulence component $\theta_{\text{turb}}(t)$ is a small random perturbation:

$$\theta_{\text{turb}}(t) = A_{\theta,\text{turb}}(\xi - 0.5), \tag{10}$$

where $A_{\theta,\text{turb}}$ controls the amplitude of random direction changes.



(a) Wind Speed & Current Speed



(b) Wind Heading

Fig. 3: Wind Speed and Heading during Simulation

3) Wind Components: The wind velocity vector $\mathbf{W}(t) = (W_x, W_y)$ is computed as:

$$W_x = W(t)\cos\theta_W(t), \tag{11}$$
$$W_y = W(t)\sin\theta_W(t). \tag{12}$$

This formulation ensures that both the magnitude and direction of the wind vary realistically over time, capturing key atmospheric phenomena that influence sailboat dynamics.

4) Surface Current Modeling Using the Ekman Spiral: The wind model is also used to estimate the speed and direction of surface currents based on principles of the Ekman Spiral [9]. Due to the Coriolis effect, the movement of surface water does not align perfectly with the wind direction. Instead, in the Northern Hemisphere, surface currents are deflected 20–45° to the right of the wind direction [10], [11]. The angle of deflection depends on various factors, including wind speed, ocean stratification, and depth.

In our model, we compute the surface current velocity components C_x and C_y as:

$$C_x = \gamma W(t)\cos(\theta_W + \theta_E), \tag{13}$$
$$C_y = \gamma W(t)\sin(\theta_W + \theta_E), \tag{14}$$

where γ is the empirical wind-current speed ratio (between 1–3% of the wind speed) [12], and θ_E is the Ekman deflection angle, which is set within the range of 20–45°.

This method ensures that surface currents realistically represent wind-driven drift, accounting for both magnitude and directional shift due to the Coriolis force.

C. Linearization

As is evident from prior sections, the dynamics associated with this system are highly nonlinear. To aid in controller design, we consider the linearized dynamics given by:

$$\dot{\mathbf{x}} = A\mathbf{x} + B\mathbf{u} \tag{15}$$

where A is the state Jacobian matrix and B is the control Jacobian matrix:

$$A = \frac{\partial \dot{x}}{\partial \mathbf{x}} \tag{16}$$

$$= \begin{bmatrix} 0 & 0 & -v\sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & v\cos(\theta) & \sin(\theta) & 0 \\ 0 & 0 & 0 & 0 & 0.5 \\ 0 & 0 & 0 & -\frac{2p_2 v}{p_9} & 0 \\ 0 & 0 & 0 & -\frac{p_3\omega}{p_{10}} & -\frac{p_3 v}{p_{10}} \end{bmatrix} \tag{17}$$

$$B = \frac{\partial \dot{x}}{\mathbf{u}} \tag{18}$$

$$= \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ -\frac{p_{11} g_r'}{p_9} & \frac{g_s'}{p_9} \\ -\frac{p_8 g_r' \cos(\delta_r)}{p_{10}} & \frac{(p_6 - p_7\cos(\delta_s))g_s' - p_7 g_s \sin(\delta_s)}{p_{10}} \end{bmatrix} \tag{19}$$

where

$$g_s' = \frac{\partial g_s}{\partial \delta_s} = (F_{x\_lift} - F_{x\_drag})\cos(\delta_s - \psi_{aw}) \tag{20}$$
$$g_r' = \frac{\partial g_r}{\partial \delta_r} = D\cos(\delta_r) \tag{21}$$

We perform the linearization of the system dynamics about some equilibrium point (x^*, u^*):

$$\mathbf{x}^* = \begin{bmatrix} x^* & y^* & \theta^* & v^* & \omega^* \end{bmatrix}^\intercal \tag{22}$$
$$\mathbf{u}^* = \begin{bmatrix} \delta_r^* & \delta_s^* \end{bmatrix}^\intercal \tag{23}$$

We pick a scenario where the sailboat is moving forward at a constant velocity in a straight line with no rotational motion. We pick the following values, that correspond to this scenario, to linearize the system:

$$\dot{\mathbf{x}} = \begin{bmatrix} 0 & 0 & 0 & 2 & 0 \end{bmatrix}^\intercal \tag{24}$$
$$\dot{\mathbf{u}} = \begin{bmatrix} 0° & 30° \end{bmatrix}^\intercal \tag{25}$$

We further assume no ocean currents, and constant wind speed, sail forces, and drag force from the rudder.

This yields the following Jacobian matrix values:

$$A = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.5 \\ 0 & 0 & 0 & -\frac{4p_2}{p_9} & 0 \\ 0 & 0 & 0 & 0 & -\frac{2p_3}{p_{10}} \end{bmatrix} \tag{26}$$

$$B = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ -\frac{10p_{11}}{p_9} & \frac{48.3}{p_9} \\ -\frac{10p_8}{p_{10}} & \frac{(p_6 - p_7\cos 30^\circ)(48.3) - p_7 g_s \sin 30^\circ}{p_{10}} \end{bmatrix} \tag{27}$$

D. Stability Analysis

For our system, we calculate the following eigenvalues for the boat:

$$\begin{bmatrix} 0 \\ 0 \\ 0 \\ -2.6616 \\ -150 \end{bmatrix} \tag{28}$$

From these eigenvalues, we can see that the system is marginally stable. The system is stable in velocity and angular velocity. This can be attributed to the fact that the drag from the water will stabilize the boat's velocity and angular velocity. The position and heading are marginally stable.

III. CONTROL

A. Hierarchical Control

In this section, we explore the high-level path planner, which determines how to get to desired waypoints. We assume that there is a list of waypoints known a priori. To plan the heading, we use a simplified model of the vehicle, modeled off a Dubins' car. This system has the dynamics:

$$\dot{x} = v\cos(\theta) \tag{29}$$
$$\dot{y} = v\sin(\theta) \tag{30}$$
$$\dot{\theta} = u \tag{31}$$

where (x, y) is the position of the vehicle, θ is the heading angle, v is the constant forward speed, and u is the control input representing the turning rate.

1) Blended LQR-Proportional Path Planner: To control the vehicle to the next waypoint, we introduce a "blended LQR-proportional controller". We seek to solve the optimal control problem of minimizing the cost function:

$$J = \int_0^{T_f} \left(x^T Q x + u^T R u\right) dt \tag{32}$$

where Q and R are positive definite matrices that weight the state and control input, respectively. We define:

$$Q = \begin{bmatrix} 10 & 0 & 0 & 0 \\ 0 & 10 & 0 & 0 \\ 0 & 0 & 5 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \tag{33}$$

$$R = 1 \tag{34}$$

so as to penalize position and heading errors. The cost function is minimized subject to the system dynamics. The solution to this problem is given by the Linear Quadratic Regulator (LQR) equations. The LQR controller computes the optimal control input u as a linear function of the state x:

$$u = -Kx \tag{35}$$

where K is the feedback gain matrix computed from the solution of the algebraic Riccati equation. The LQR controller is designed to minimize the cost function while ensuring that the system dynamics are satisfied.

However, for certain conditions, the system dynamics are rendered uncontrollable. This occurs when the controllability matrix $\mathcal{C} = \begin{bmatrix} B & AB & A^2B & \cdots & A^{n-1}B \end{bmatrix}$ is not full rank. For states where the linearized system dynamics are uncontrollable, we switch to a proportional controller on the heading error defined in Eqn. 38. The heading error is defined as the angle between the current heading and the heading to the next waypoint. The control input is then the heading error multiplied by a proportional gain. The control input is then used to determine the turning rate of the vehicle. In this work, we use $k_p = 1$. This gain is chosen as it sets the angular rate of the vehicle equal to the error in heading (in units of radians), and is associated with a settling time of less than 4 seconds (which is sufficient for our system). We calculate k_p by:

$$T_s = \frac{\ln(0.02)}{-k_p} \tag{36}$$
$$k_p = \frac{-\ln(0.02)}{4} \approx 1 \tag{37}$$

$$u = k_p(\theta_{waypoint} - \theta_{boat}) \tag{38}$$

2) Lower-Level Angle-Based Controller: From Sec. III-A1, we now have a desired path to follow, which can be passed as a desired heading θ_{des} for the lower-level controller. From the current state of the boat and the environmental state, we can compute the effective velocity of the boat by:

$$V_{\text{eff}_x} = v_b\cos(\theta_b) + V_c\cos(\psi_c) \tag{39}$$
$$V_{\text{eff}_y} = v_b\sin(\theta_b) + V_c\sin(\psi_c) \tag{40}$$
$$V_{\text{eff}} = \begin{bmatrix} V_{\text{eff}_x} \\ V_{\text{eff}_y} \end{bmatrix} \tag{41}$$

The apparent wind is given by:

$$V_{\text{aw}_x} = V_w\cos(\psi_w) - V_{\text{eff}_x} \tag{42}$$
$$V_{\text{aw}_y} = V_w\sin(\psi_w) - V_{\text{eff}_y} \tag{43}$$
$$V_{\text{aw}} = \begin{bmatrix} V_{\text{aw}_x} \\ V_{\text{aw}_y} \end{bmatrix} \tag{44}$$

with associated apparent wind angle

$$\psi_{\text{aw}} = \arctan(V_{\text{aw}_y}, V_{\text{aw}_x}) \tag{45}$$

where $\psi_{\text{aw}} \in [0, 2\pi]$.

We want to trim the sail such that the resultant force vector will push us towards our waypoint. Thus, we compute:

$$\delta_s = 0.5 \times (\psi_{\text{aw}} - \theta_b) \tag{46}$$

where δ_s is clamped to the range $[\delta_{s_{\min}}, \delta_{s_{\max}}]$.

We compute the heading error of the boat as:

$$e_\theta = \text{mod}(\theta_{\text{des}} - \theta_b + \pi, 2\pi) - \pi \tag{47}$$

We then use a proportional controller to command the rudder deflection:

$$\delta_r = K_p \times e_\theta \tag{48}$$

where δ_r is clamped to the range $[\delta_{r_{\min}}, \delta_{r_{\max}}]$.

The command inputs are normalized by:

$$\delta_r = \text{mod}(\delta_r + \pi, 2\pi) - \pi \tag{49}$$

$$\delta_s = \text{mod}(\delta_s + \pi, 2\pi) - \pi \tag{50}$$

B. Replanning

There exist certain waypoints that cannot be reached (due to the fact that sailboats cannot sail upwind). While upwind progress can be achieved via a process called "tacking" [13], we instead introduce a waypoint replanning algorithm that avoids waypoints for which tacking would be necessary to achieve. This method creates a "No-Sail-Zone" from $45°$ +/- the oncoming wind direction, as can be seen in Fig. 4. If a waypoint is within this zone, the controller will not attempt to travel towards it, and will instead proceed to the next available waypoint.

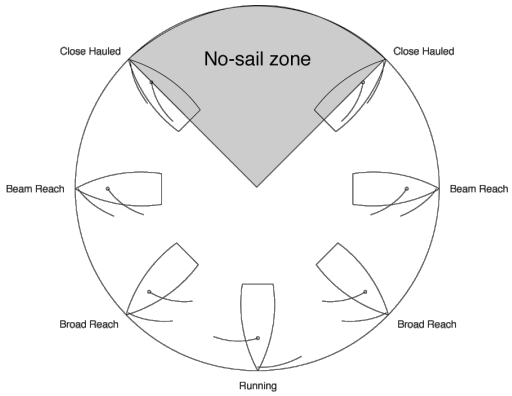


Fig. 4: Boat headings determined by wind direction

C. Genetic Algorithm Approach

The above control scheme requires many simplifications to the complex dynamics of the autonomous sailboat. For high dimension systems and systems with highly nonlinear dynamics, it is often difficult to design a controller with classical techniques. As such, we can apply learning methods to learn a controller that can adapt to the dynamics of the system. In this work, we apply a genetic algorithm to learn a controller that can adapt to the dynamics of the system. The genetic algorithm is a population-based optimization algorithm that mimics the process of natural selection. The algorithm starts with a population of random solutions and iteratively improves them through selection, crossover, and mutation operations.

In this work, we consider a population of controllers, each represented by the control signal as a function of time. The fitness of each controller is evaluated based on its performance in navigating the sailboat to the desired waypoints. The best-performing controllers are selected for reproduction, and their parameters are combined to create new controllers. The new controllers are then mutated by adding small random perturbations to their parameters. This process continues until a stopping criterion is met, such as a maximum number of generations or a satisfactory level of performance.

For this work, we define the population size to be 100, the number of generations to be 100, the mutation rate to be 0.65 and the crossover rate to be 0.2.

IV. SIMULATION & RESULTS

A. Hierarchical Control

We simulate the hierarchical control architecture described in Sec. III-A for the following ordered list of waypoints:

$$\begin{bmatrix} x & y \end{bmatrix} = \begin{bmatrix} 10 & 10 \\ 20 & 5 \\ -30 & 50 \\ 40 & 10 \\ -50 & 20 \\ 100 & 0 \\ 150 & 15 \end{bmatrix}$$

We consider a waypoint "reached" when the boat is within 2 meters of the waypoint (computed by the norm of the waypoint position and the boat position). The path taken by the boat is shown in Fig. 5. Notice how two of the waypoints are unvisited. This is due to the fact that the boat was unable to reach them due to the waypoints being location within the "no-sail zone", and so the waypoint replanner selected the next waypoint in the list for the boat to travel towards. The boat was able to reach the other waypoints, and the replanning algorithm was able to successfully navigate around the no-sail zone.

B. Genetic Algorithm

We simulate the genetic algorithm described in Sec. III-C for the following ordered list of waypoints:

$$\begin{bmatrix} x & y \end{bmatrix} = \begin{bmatrix} 10 & 10 \\ 20 & 5 \\ -30 & 50 \\ 40 & 10 \\ -50 & 20 \\ 100 & 0 \\ 150 & 15 \end{bmatrix}$$

For the parameters selected, the best performing controller resulted in the path shown in Fig. 6. Notice how the boat was only able to reach a couple of the waypoints. This is due to the fact that with the limited computing resources, the

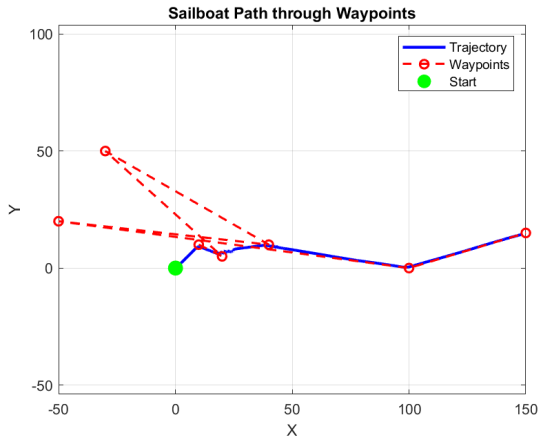


Fig. 5: Path taken by the boat using the hierarchical control architecture. The red dots represent the waypoints and nominal path, and the blue line represents the path taken by the boat.

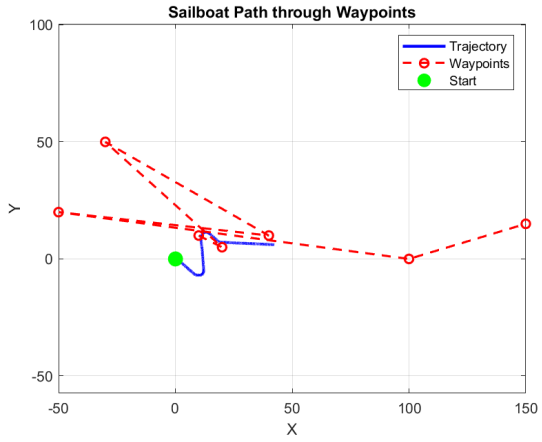


Fig. 6: Path taken by the boat using the genetic algorithm. The red dots represent the waypoints and nominal path, and the blue line represents the path taken by the boat.

genetic algorithm did not have sufficient time to evolve to a solution that is capable of reaching all of the waypoints. This highlights one of the drawbacks of genetic algorithms, which is that they can be computationally expensive and may not always converge to an optimal solution. Additionally, there are no optimality guarantees for the genetic algorithm, and so it is possible that the solution found is not the best possible solution.

V. CONCLUSION

In this work, we have presented a model and control strategy for an autonomous sailboat. The model incorporates the effects of wind and ocean currents on the vehicle's dynamics, allowing for accurate simulation of its behavior in real-world conditions. We have developed a control strategy that combines LQR-based path following with a lower-level angle-based controller to ensure stability and robustness in varying environmental conditions.

We have shown that the proposed model and control strategy can effectively navigate the sailboat through a series of waypoints, even in the presence of environmental disturbances. The simulation results demonstrate the effectiveness of the hierarchical control architecture, which allows for real-time re-planning of waypoints based on the vehicle's current state and environmental conditions.

Some limitations of the current work include the assumption of a constant wind speed and direction, as well as the simplification of the sail and rudder dynamics. Future work will focus on incorporating more realistic wind models, as well as exploring advanced control strategies such as reinforcement learning and model predictive control to further improve the performance of the autonomous sailboat. Additionally, we will investigate the integration of sensor data for real-time feedback and adaptation of the control strategy to enhance the robustness of the system in dynamic environments.

REFERENCES

[1] C. Strömbeck, "Modeling, control and optimal trajectory determination for an autonomous sailboat," 2017. [Online]. Available: https://api.semanticscholar.org/CorpusID:116596277

[2] A. Araújo, R. Guerra, V. Mankina, P. Preux, C. Distante, C. Vasconcellos, D. Brandão, L. Gonçalves, and E. Clua, "Towards an autonomous sailboat navigation control architecture," in *2024 Latin American Robotics Symposium (LARS)*. IEEE, 2024, pp. 1–6.

[3] A. P. Negreiros, W. S. Correa, A. P. de Araujo, D. H. Santos, J. M. Vilas-Boas, D. H. Dias, E. W. Clua, and L. M. Gonçalves, "Sustainable solutions for sea monitoring with robotic sailboats: N-boat and f-boat twins," *Frontiers in Robotics and AI*, vol. 9, p. 788212, 2022.

[4] Z. Sun, J. Yu, W. Zhao, F. Hu, J. Wang, and Q. Jin, "Autonomous sailboat velocity prediction program considering the sea-surface wind velocity gradient," *Applied Ocean Research*, vol. 155, p. 104442, 2025.

[5] J. Melin, "Modeling, control and state-estimation for an autonomous sailboat," 2015.

[6] M. F. Silva, A. Friebe, B. Malheiro, P. Guedes, P. Ferreira, and M. Waller, "Rigid wing sailboats: A state of the art survey," *Ocean Engineering*, vol. 187, p. 106150, 2019.

[7] A. Miller and A. Rak, "Modelling of lake wind parameters to simulate environmental disturbances for a scaled ship model," *Electronics*, vol. 14, no. 1, p. 117, 2024.

[8] K. Cole and A. Wickenheiser, "Spatio-temporal wind modeling for uav simulations," *arXiv preprint arXiv:1905.09954*, 2019.

[9] T. Ellison, "The ekman spiral," *Quarterly Journal of the Royal Meteorological Society*, vol. 81, no. 350, pp. 637–638, 1955.

[10] U. of Hawai'i, "Ocean surface currents," Exploring Our Fluid Earth. [Online]. Available: https://manoa.hawaii.edu/exploringourfluidearth/physical/atmospheric-effects/ocean-surface-currents

[11] J. M. Kaihatu, R. A. Handler, G. O. Marmorino, and L. K. Shay, "Empirical orthogonal function analysis of ocean surface currents using complex and real-vector methods," *Journal of atmospheric and oceanic technology*, vol. 15, no. 4, pp. 927–941, 1998.

[12] M. Van der Mheen, C. Pattiaratchi, S. Cosoli, and M. Wandres, "Depth-dependent correction for wind-driven drift current in particle tracking applications," *Frontiers in Marine Science*, vol. 7, p. 305, 2020.

[13] Q. Sun, Z. Qiao, C. Strömbeck, Y. Qu, H. Liu, and H. Qian, "Tacking control of an autonomous sailboat based on force polar diagram," in *2018 13th World Congress on Intelligent Control and Automation (WCICA)*. IEEE, 2018, pp. 467–473.