

Stochastic Optimal Control of a Sailboat

Cole Miles, Alexander Vladimirovsky

Abstract—In match race sailing, competitors must steer their boats upwind in the presence of unpredictably evolving weather. Combined with the tacking motion necessary to make upwind progress, this makes it natural to model their path-planning as hybrid stochastic optimal control problems. Dynamic programming provides the tools for solving these, but the computational cost can be significant. We greatly accelerate a semi-Lagrangian iterative approach of Ferretti and Festa by reducing the state space dimension and designing an adaptive timestep discretization that is very nearly causal. We also provide a more accurate tack-switching operator by integrating over potential wind states after the switch. The method is illustrated through a series of simulations with varying stochastic wind conditions.

Index Terms—Stochastic Optimal Control, Switched Systems, Numerical Algorithms

I. INTRODUCTION

IN recent years, advances in numerical and mathematical methods have greatly impacted the sport of sailing. Yacht path planning algorithms are now heavily employed both as a part of match simulation software, and as a tool to provide live guidance on optimal decision-making during a race. Here, we focus on the problem of finding the optimal feedback control to perform live path planning in the presence of a stochastically-evolving weather system.

Weather conditions greatly impact the performance capabilities of a sailboat in a race. While it is possible to produce meteorological forecasts, there is an inevitable stochastic component to how weather conditions evolve during the course of a race which is impossible to predict exactly. It is critical that algorithms used for decision making correctly factor this randomness into their control policy. In [2], this randomness is modeled as a discrete-time Markov chain process which updates weather conditions at a fixed tick rate. In [3] this idea is extended to a continuous-time Markov chain, formulating the system as a free boundary problem whose solution is a switching curve giving the optimal control policy. Here, we will take the approach of [1] and describe optimality through dynamic programming, yielding a system of two coupled quasi-variational inequalities of Hamilton-Jacobi-Bellman (HJB) type with degenerate ellipticity.

For full physical accuracy, a model would have to include both the position and velocity of the sailboat in the state

space, and model the forces on the sailboat in different state configurations, leading to at best 5 coupled first-order equations in 5 state variables. An approach along these lines can be found in [2]. But in long course matches, inertial effects associated with accelerating up to the equilibrium speed are (in general) secondary, allowing one to reduce the system to 3 coupled first-order equations by assuming that the velocity vector can be changed instantaneously.

However, this assumption cannot be made in modeling the process of *tacking*, which is necessary when attempting to travel upwind. To make progress, sailors must travel at angles to the wind, repeatedly switching directions to steer in a zig-zag pattern on either side of the wind direction. Each time the bow is swung to the other side of the wind (a single *tack*), the boat is significantly slowed down for the period of time in which the wind pushes against the boat. In previous works ([1],[4]), this is incorporated by modeling the system as a *hybrid control* problem, with a continuous control corresponding to the angle of the sails and a discrete control corresponding to which tack we should be on. The lost inertia has been modeled as a fixed switching delay, reflecting the time lost due to the reduced speed from performing the tack.

In this work, we extend and improve upon the methods of Ferretti and Festa [1], who solve the problem using a semi-Lagrangian discretization. In Sec. II, we give the original problem statement and recast it to a reduced coordinate system in 2D, which alone greatly reduces the time needed to solve the problem numerically. In Sec. III, we set up the optimal control problem following the structure of [1], and provide an improvement to their *switching operator* to incorporate the wind evolution during the process of a tack. In Sec. IV, we describe our semi-Lagrangian numerical methods, including a discretization in which the system is nearly-causal and can be solved in a handful of iterations. We present the results of numerical simulations in Sec. V, producing optimal switching maps and highlighting the effect of our improved switching operator. We conclude by listing directions for future work in Sec. VI.

II. SYSTEM DYNAMICS

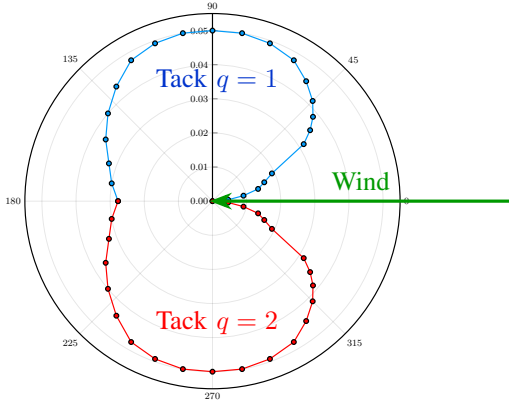
We consider a problem introduced in [1]: time-optimal sailboat navigation to a closed target set $\mathcal{T}$ when the wind direction evolves stochastically. For simplicity, we focus on their simplified model where the wind speed is fixed, and the angle ϕ of the upwind direction (measured counterclockwise from the y -axis) evolves under a drift-diffusion process:

$$d\phi = adt + \sigma dW, \quad (1)$$

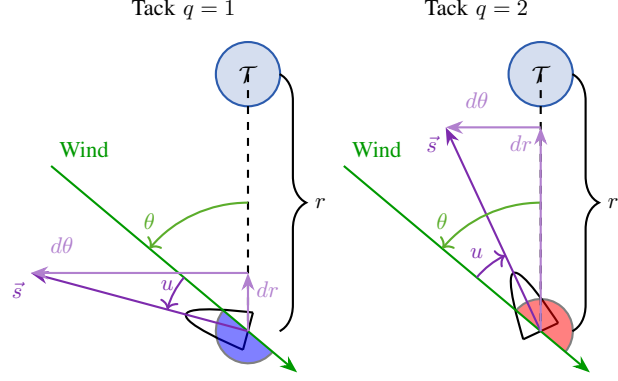
This work was supported in part by the NSF DMS (awards 1738010 and 2111522), and the DOE CSGF (award DE-SC0020347).

Cole Miles is with the Physics Department, Cornell University (e-mail: cmm572@cornell.edu).

Alexander Vladimirovsky is with the Mathematics Department, Cornell University (e-mail cmm572@cornell.edu)



(a) The polar speed plot $s(u)$ used in this work. Note the symmetry about $u = 0^\circ$, non-convexities near 0° and 180° , and that the speed vanishes at $u = 0^\circ$. These properties are common to all polar plots.



(b) A diagram of the system setup in reduced coordinates.

Fig. 1: Boat dynamics relative to the wind and relative to a target.

where a is the drift rate, σ is the strength of the Brownian motion, and dW is the differential of a Wiener process.

We control the angle of the boat's velocity relative to the upwind direction. But at any given time we are also restricted to an interval of angles determined by the sailboat's current *tack*, or configuration of sails. We represent the tack as a controlled discrete variable $q \in \{1, 2\}$; $q = 1$ represents the starboard tack, which we model as having access to the angles $[0, \pi]$, while the port tack $q = 2$ has access to $[-\pi, 0]$. For notational convenience, we will use $(-1)^q u$ to encode the velocity angle relative to the upwind direction, with the steering control variable $u \in [0, \pi]$. So, the steering angle u is measured counterclockwise for $q = 1$ and clockwise for $q = 2$, with $u = 0$ corresponding to motion directly against the wind. Due to our inertia-less approximation, whenever u changes, the boat is assumed to instantly start travelling in the new direction with the angle-dependent speed $s(u)$. The speed profile (termed the *polar plot*) capturing this dependence is determined by the exact geometry of a specific boat. A typical example [5] is provided in Fig. 1a.

Assuming that the boat's location is $(x(t), y(t))$, the system dynamics in absolute coordinates is then

$$\begin{aligned} dx &= -s(u) \sin(\phi - (-1)^q u) dt, \\ dy &= s(u) \cos(\phi - (-1)^q u) dt, \\ d\phi &= a dt + \sigma dW. \end{aligned} \quad (2)$$

We now simplify by focusing on radially symmetric problems: if we assume that the target $\mathcal{T}$ is circular, we no longer care about an absolute angle of the boat's position relative to the target. Shifting to a coordinate system centered at $\mathcal{T}$, we can reduce the (x, y, ϕ) state space to just two coordinates (r, θ) , where $r = \sqrt{x^2 + y^2}$ is the radial distance of the boat from the center of $\mathcal{T}$ and $\theta = \phi + \text{atan2}(-x, -y)$ is the angle of the wind measured counterclockwise relative to the straight line between the boat and the target; see Fig. 1b. In

this reduced representation, the dynamics are governed by

$$\begin{aligned} dr &= -s(u) \cos(\theta - (-1)^q u) dt, \\ d\theta &= \left[\frac{s(u)}{r} \sin(\theta - (-1)^q u) + a \right] dt + \sigma dW. \end{aligned} \quad (3)$$

which describe the projections of the velocity vector onto the radial and angular components, combined with the drift-diffusion evolution of the wind. We refer to the coefficients on dt in each equation as $\dot{r}$ and $\dot{\theta}$, respectively. New from the original system is the $\dot{\theta} \propto \frac{1}{r}$ relationship, which will force us to utilize an adaptive timestepping scheme that chooses smaller timesteps near the target to counteract the rapidly increasing angular speed. We note that θ still remains bounded since $\mathcal{T}$ has positive radius.

III. STOCHASTIC OPTIMAL CONTROL

Assuming we start from a position $\mathbf{r} = (r, \theta)$ and tack q , the total time-to-target $T_\mu(\mathbf{r}, q) = \min\{t \mid \mathbf{r}'(t) \in \mathcal{T}, \mathbf{r}'(0) = \mathbf{r}, q'(0) = q\}$ is a random variable whose expectation we are trying to minimize¹ by choosing a feedback steering/tack-switching policy μ . This $T_\mu(\mathbf{r}, q)$ can be split into two pieces: the time spent continuously controlling the sails plus the time spent performing tack-switches. Each tack-switch incurs a *switching delay* C due to the speed loss from the wind pushing directly against the boat's intended direction during the maneuver². We model the boat's propelled speed as falling to zero (i.e., $s(u) = 0$) during a tack-switch.

We thus define the *value function* to encode the minimal expected time to $\mathcal{T}$ from each starting configuration:

$$v(\mathbf{r}, q) := \inf_{\mu} \mathbb{E}[T_\mu(\mathbf{r}, q)]. \quad (4)$$

¹Strictly speaking, the objective in [1] is different since they minimize $\mathbb{E} \left[\int_0^T e^{-\lambda t} dt \right] = \mathbb{E}[(1 - \exp(-\lambda T))/\lambda]$, where $\lambda > 0$ is a time-discounting coefficient chosen to improve the convergence of value iterations. Our objective is obtained by taking the limit with $\lambda \rightarrow 0$ and the numerical tests indicate that value iterations based on our discretization converge even in this "undiscounted" case.

²For simplicity of modeling, we will treat this delay as constant, though state-dependent C would not present additional computational challenges.

By Bellman's Optimality Principle, this v must satisfy

$$v(\mathbf{r}, q) = \min \left(\tau + \inf_{\mu} \Sigma_{\mu, \tau} v(\mathbf{r}, q), \quad C + \mathcal{N}v(\mathbf{r}, q) \right) + o(\tau), \quad (5)$$

where the first argument to the min is to be interpreted as the best expected time-to- $\mathcal{T}$ if we stay on the current tack at least for a small time τ , while the second is the best expected time if we switch tacks immediately. More formally, we define the operators $\Sigma_{\mu, \tau}$ and $\mathcal{N}$ as:

$$\Sigma_{\mu, \tau} v(\mathbf{r}, q) = \mathbb{E}_{\mathbf{r}'; \mu, \tau} [v(\mathbf{r}', q) | \mathbf{r}], \quad (6)$$

$$\mathcal{N}v(\mathbf{r}, q) = \mathbb{E}_{\mathbf{r}'; \emptyset, C} [v(\mathbf{r}', 3 - q) | \mathbf{r}], \quad (7)$$

where we employ the compact notation $\mathbb{E}_{\mathbf{r}'; \mu, \tau} [\cdot | \mathbf{r}]$ to indicate the expectation of a function over evolved positions $\mathbf{r}'$, starting from position $\mathbf{r}$ and using control policy μ for a time τ . In (7), the symbol $\emptyset$ in place of the control policy indicates that during the tacking time C , no steering control is taking place as $s(u) = 0$ for the duration. Importantly, the version of operator (7) used in [1] does not treat θ as evolving during the tack, instead setting $\mathcal{N}v(\mathbf{r}, q) = v(\mathbf{r}, 3 - q)$. As we show in section V, correctly incorporating these switch-duration dynamics is critical in high-drift problems.

A standard argument based on an Ito-Taylor expansion of (5) yields the quasi-variational inequality that must be satisfied by a smooth value function:

$$\max \left(v - \mathcal{N}v - C, \quad H(\mathbf{r}, q, \nabla_{\mathbf{r}} v) - \frac{\sigma^2}{2} v_{\theta\theta} \right) = 0, \quad (8)$$

with the Hamiltonian

$$H(\mathbf{r}, q, \mathbf{p}) = \max_u \left(-\dot{\mathbf{r}}(\mathbf{r}, q, u) p_1 - \dot{\theta}(\mathbf{r}, q, u) p_2 \right) - 1. \quad (9)$$

If v is not smooth, one can still interpret it as a weak solution (in particular, the unique *viscosity solution* [6]) of (8).

As θ is the only randomly evolving variable and obeys drift-diffusive dynamics, both expectations in (6) and (7) take a similar form as averaging v over Gaussian-distributed final θ' . For small τ , an approximation to (6) taking $u = \mu(\mathbf{r}, q)$ and $\dot{\mathbf{r}}, \dot{\theta}$ constant for a time τ yields:

$$\Sigma_{u, \tau} v(r, \theta, q) = \frac{1}{\sqrt{2\pi\tau\sigma}} \int_{-\infty}^{\infty} d\theta' \exp \left\{ -\frac{(\theta' - \theta - \dot{\theta}\tau)^2}{2\tau\sigma^2} \right\} v(r + \dot{\mathbf{r}}\tau, \theta', q), \quad (10)$$

where v is interpreted as 2π -periodic in θ' , and $\dot{\mathbf{r}}, \dot{\theta}$ both implicitly depend on (r, θ, q, u) . Eq. (7) is the same formula under the substitutions $q \rightarrow 3 - q, \tau \rightarrow C, \dot{\mathbf{r}} = 0, \dot{\theta} = a$.

IV. NUMERICAL IMPLEMENTATION

A. Semi-Lagrangian Discretization

For a fixed small τ , an approximation of the value function v can be obtained as a fixed point of the mapping defined by ignoring the $o(\tau)$ term in (5). It can be shown that if the schemes approximating both arguments in the min are monotone, stable, and consistent with (8), then iterating this mapping is guaranteed to converge [7]. As in [1], to

build a monotone scheme we discretize the state space on a rectangular grid and perform *value iterations* of (5) starting from an initial guess v^0 to produce a sequence of approximate value functions v^n converging to the solution of the discretized version of (5). Using monotone schemes, choosing v^0 such that the first iteration produces $v^1 \leq v^0$ everywhere is sufficient to ensure convergence [8].

To approximate the continuous-control operator $\Sigma_{\mu, \tau}$, we change the variable of integration in (10) to $\xi = (\theta' - \theta - \dot{\theta}\tau)/\sqrt{2\tau\sigma^2}$ and use the Gauss-Hermite quadrature [9]:

$$\begin{aligned} \Sigma_{u, \tau} v^n(\mathbf{r}, q) &= \frac{1}{\sqrt{\pi}} \int_{-\infty}^{\infty} d\xi e^{-\xi^2} v^n(r + \dot{\mathbf{r}}\tau, \sqrt{2\tau\sigma}\xi + \theta + \dot{\theta}\tau, q) \\ &\approx \frac{1}{\sqrt{\pi}} \sum_{i=1}^M w_i v^n(r + \dot{\mathbf{r}}\tau, \sqrt{2\tau\sigma}\xi_i + \theta + \dot{\theta}\tau, q), \end{aligned} \quad (11)$$

where M is the number of quadrature points used, ξ_i are the roots of the M^{th} Hermite polynomial $H_M(\xi)$, and w_i are

$$w_i = \frac{2^{M-1} M! \sqrt{\pi}}{M^2 [H_{M-1}(\xi_i)]^2}. \quad (12)$$

The sum in (11) requires evaluating v^n off-grid, which we accomplish through interpolation, as is common for *semi-Lagrangian schemes* [10]. The latter are guaranteed to be monotone as long as the interpolation used is zeroth or first order [11]. In our implementation, we define the numerical operator $S_{u, \tau}$ by the sum of (11) evaluated using bilinear interpolation between the 4 closest gridpoints. We use $M = 2$ quadrature points, which reduces $S_{u, \tau}$ to the form in [1], averaging between the two points $\mathbf{r}_{\pm} = (r + \dot{\mathbf{r}}\tau, \theta + \dot{\theta}\tau \pm \sigma\sqrt{\tau})$.

Due to $\dot{\theta}$ rapidly increasing as $\frac{1}{r}$ near the target, we adaptively choose our time discretization as:

$$\tau(\mathbf{r}, q, u) \propto \min \left(\frac{\Delta r}{\dot{\mathbf{r}}(\mathbf{r}, q, u)}, \frac{\Delta \theta}{\dot{\theta}(\mathbf{r}, q, u)} \right). \quad (13)$$

We choose the constant of proportionality in (13) to be 1.5, to step into the middle of the first grid cell along the direction of motion. We then denote the scheme S as the optimization of this operator over control angles u :

$$Sv^n(\mathbf{r}, q) = \min_u \left(\tau(\mathbf{r}, q, u) + S_{u, \tau(\mathbf{r}, q, u)} v^n(\mathbf{r}, q) \right). \quad (14)$$

The optimal control policy solution is then precisely the arg min of (14). In practice, since our polar plot datasets contain a discrete set of $s(u)$ values, we perform this minimization by a simple grid search over the tabulated angles. The accuracy of this procedure can be also improved by expanding the available set of control angles through interpolation.

The same reparameterization and Gauss-Hermite quadrature approach is used to produce a numerical operator N which approximates the switching operator of (7) as

$$Nv^n(\mathbf{r}, q) = \frac{1}{\sqrt{\pi}} \sum_{i=1}^M w_i v^n(r, \sqrt{2C\sigma}\xi_i + \theta + aC, 3 - q). \quad (15)$$

To improve the accuracy, our implementation uses $M = 3$ for this switching quadrature since $C \gg \tau$ in our experiments.

Our full method is summarized below in Algorithm 1.

```

1 Initialize: set  $v^0 = 0$  on  $\mathcal{T}$  and  $v^0 = \infty$  elsewhere.
2 Initialize: set  $\text{maxdiff} = \infty$  and  $n = 0$ .
3 while  $\text{maxdiff} > \epsilon$  do
4   foreach  $r \in \mathcal{R}$  do
5     foreach  $\theta \in \Theta$  do
6       foreach  $q \in \{1, 2\}$  do
7         foreach  $u_i \in \mathcal{U}$  do
8            $\tau = 1.5 \min \left( \frac{\Delta r}{\dot{r}(q, u_i)}, \frac{\Delta \theta}{\dot{\theta}(r, q, u_i)} \right)$ 
9            $S_i^\Delta = S_{u_i, \tau} v^n(1r, q)$ 
10           $S^\Delta = \min_i S_i^\Delta$ 
11           $N^\Delta = C + N v^n(r, q)$ 
12           $v^{n+1}(r, q) = \min(S^\Delta, N^\Delta)$ 
13         $\text{maxdiff} = \max_{r, q} (|v^{n+1}(r, q) - v^n(r, q)|)$ 
14       $n = n + 1$ 

```

Algorithm 1: Semi-Lagrangian value iterations to a convergence criterion ϵ . Here, $\mathcal{R}, \Theta, \mathcal{U}$ refer to the discrete set of gridpoints sampled for the state and control spaces.

B. Gauss-Seidel Iterations

Gauss-Seidel relaxation can greatly reduce the number of iterations needed to reach convergence by exploiting causality. We heuristically find that time-parameterized stochastic-optimal paths almost always have monotone decreasing $r(t)$. Wherever this holds, the value function only depends on the values of gridpoints with smaller r . This can be exploited by updating v^n in-place, sweeping along increasing r so that points later in the sweep see already-updated values at smaller r rather than old values from the previous iteration.

An additional change can be made to our discretization to further take advantage of this causal dependence. To ensure the currently updated point only depends on those at smaller r , we choose the adaptive timestep so that we always step at least a full grid length along the r axis:

$$\tau(r, q, u) = 1.5 \frac{\Delta r}{\dot{r}(r, q, u)}. \quad (16)$$

We refer to this second discretization as “row-wise Gauss-Seidel” (rwGS) relaxation. Our experiments show that it requires a nearly-constant number of iterations, only slowly beginning to increase at very fine grids. Meanwhile, both the Gauss-Jacobi (GJ) and the standard Gauss-Seidel (GS) require a number of iterations roughly linear in N_R (the number of grid points along the r direction). As shown in Fig. 2, on the finest tested grid, GJ requires a factor of ≈ 100 more iterations than GS, which itself requires a factor of ≈ 100 more iterations than the rwGS. However, this massive speedup in the latter is not completely free. Comparing (13) and (16), we see that the row-wise iteration uses larger timesteps than the standard discretization in areas of the domain where $\Delta\theta/\dot{\theta} < \Delta r/\dot{r}$, in particular close to the target. Additional numerical tests (summarized in Supplementary Materials) show that, despite slightly larger discretization errors in rwGS,

these errors decrease with the same rate as in GS under the grid refinement.

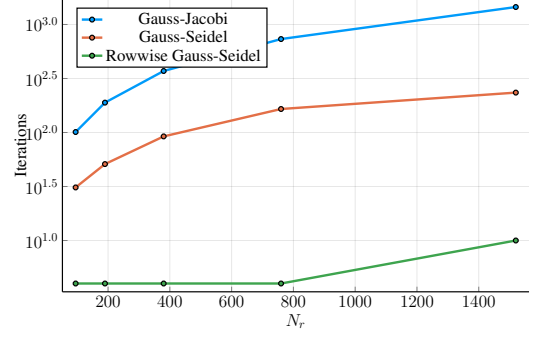


Fig. 2: The number of iterations required to reach convergence with a fixed $\epsilon = 10^{-8}$ and an increasing number of grid points along each state dimension. In this plot, we have fixed $\sigma = 0.05, a = 0.1$. At each point, N_θ is scaled accordingly with N_r .

C. Live Simulations / Control Synthesis

Using the computed grid-approximation of the value function, we can perform live sailboat navigation using a derived “direction grid” recording the controls u_i that minimize S^Δ at each gridpoint, as well as a “switchgrid” that indicates where it is optimal to switch to the opposite tack (i.e., wherever $N^\Delta < S^\Delta$ in Algorithm 1). In our simulations, we evolve the system (3) using a simple Euler–Maruyama method. We switch to the opposite tack whenever at least 3 of the surrounding 4 gridpoints have $N^\Delta < S^\Delta$; otherwise, we stay on the current tack and use the optimal control value of the nearest gridpoint in the direction grid.

V. NUMERICAL EXPERIMENTS

We now apply our algorithm to solve the optimal control problem (3)–(4) for a variety of system parameters, and sample stochastic optimal trajectories for each scenario³. For all results shown, the target radius is $R = 0.1$ and the domain is $(r, \theta) \in [R, 2.0] \times [0, 2\pi]$. Target boundary conditions are set as $v(r, \theta, q) = 0$ for $r < R$, exit boundary conditions as $v(r, \theta, q) = 10^6$ for $r > 2.0$, and the switching delay $C = 2$.

We first focus on problems with zero drift ($a = 0$), but varying the diffusivity coefficient σ . In Fig. 3, we plot switchgrid solutions for a collection of σ values, noting that these grids are defined and plotted in (r, θ) state space, not the absolute position space. Points are marked red if $\text{switchgrid}(r, \theta, q = 1) = 1$, blue if $\text{switchgrid}(r, \theta, q = 2) = 1$, and white if both are 0. The darker shaded regions correspond to the switchgrids obtained if we use the zero-evolution switching operator of [1] as compared to our (7). The black circle at the center marks the target set $\mathcal{T}$. In general, stochastic optimal trajectories “bounce” between these red and blue switching fronts as they approach $\mathcal{T}$.

³To ensure the computational reproducibility, we provide the full source code, movies illustrating these simulations, and additional convergence data at https://eikonal-equation.github.io/Stochastic_Sailing/

With $\sigma = 0$, the switching fronts lie at constant θ , corresponding to the angles that locally maximize the projection of the polar speed plot (Fig. 1a) onto the axis of the wind. Once a sailboat hits this front, it can switch tack and stay at this constant angle that maximizes its speed towards the target (see Fig. 4(a)). As σ increases, these switching fronts contract more tightly around $\theta = 0^\circ, 180^\circ$. As the future wind state grows more uncertain with larger σ , the optimal policy becomes more conservative and constrains the boat to smaller excursions in θ before switching to the opposite tack. We observe that in these zero-drift problems, our improved switching operator makes the switching grids modestly more conservative, switching earlier by a few degrees in Fig. 3(b,c).

In Fig. 4, we plot samples of stochastic optimal trajectories corresponding to these control policies, with blue markers indicating where tack switches occur. Similarly to [1], we initialize six boats at positions $(r, \theta) \in \{(1.8, 0.0), (1.93, \pm 0.39)\}$, and tacks $q \in \{1, 2\}$. We then evolve and control each boat's state until it reaches its target as described in Sec. IV-C. We can see that, as foreshadowed by the switchgrids, at higher values of σ the trajectories tend to stay within tighter cones from the target. Next, in Figs. 5 and 6 we show switchgrids and sampled trajectories for problems in which the drift parameter a is nonzero, with a fixed $\sigma = 0.05$. As the drift increases, we observe the switchgrids and the trajectories becoming skewed and asymmetric towards negative θ . This reflects that it is optimal to “lead” the wind in anticipation of future wind evolution, allowing the boat to stay in the portion of state space that allows it to steer at the optimal 40° - 45° angle while the wind evolves.

Additionally, some “wave”-like features start to emerge in the switchgrids for large a . Empirically, we observe that the number of “waves” in the switchgrid corresponds to the number of 180° turns the wind tends to make by the time an average boat at $r = R$ reaches the target. Each “wave” can thus be interpreted as corresponding to the optimal control during one (average) half-cycle of the wind. We notice that accurately modeling the switching operator is critical in these high-drift scenarios, as the two options produce dramatically different wave-like structures in the switchgrids.

VI. CONCLUSIONS

We have introduced an accelerated semi-Lagrangian approach to solve a hybrid stochastic optimal control problem arising in match race sailing. By working in a reduced coordinate system and designing a nearly-causal discretization, we were able to produce control policies on high-resolution grids which provide insight into tack-switching strategies under unpredictable weather.

While the conversion to reduced coordinates greatly accelerates our solution, it also introduces several limitations since we had to assume that the problem is radially symmetric. This restricts the target shape to a circle, and does not allow the inclusion of state constraints such as obstacles or nearby coastline. However, it does still allow for modeling match races between multiple agents. Our approach will require keeping track of (r, θ) coordinates for each agent, yielding an

overall state space with one less dimension than in [4]. For this reason, we hope that our approach will allow to rapidly test the implications of various weather models or match strategies in “ideal” conditions far away from any barriers.

While this paper focuses on the risk-neutral task of finding a feedback policy μ that minimizes the expected time to target $\mathbb{E}[T_\mu]$, practitioners often prefer *robust* control policies when bad outcomes are costly. Two such popular approaches are the “risk-sensitive” control [12], [13] and H^∞ control [14]. But neither of these provides any direct bounds on the likelihood of bad scenarios; i.e., $w_\mu(c) = \mathbb{P}(T_\mu \geq c)$. An efficient method for minimizing $w_\mu(c)$ for all threshold values c simultaneously was recently introduced for piecewise-deterministic Markov processes in [15]. We hope that it can be similarly extended to the current context. It would be also interesting to extend the methods of multiobjective optimal control (e.g., [16], [17]) to find Pareto-optimal policies reflecting the rational tradeoffs between conflicting optimization criteria (e.g., $\mathbb{E}[T_\mu]$ vs $w_\mu(c)$ or even $w_\mu(c_1)$ vs $w_\mu(c_2)$).

Acknowledgements: The authors are grateful to Roberto Ferretti for sparking their interest in this class of problems.

REFERENCES

- [1] R. Ferretti and A. Festa, “Optimal Route Planning for Sailing Boats: A Hybrid Formulation,” *Journal of Optimization Theory and Applications*, vol. 181, no. 3, pp. 1015–1032, 2019.
- [2] A. B. Philpott, S. G. Henderson, and D. Teirney, “A Simulation Model for Predicting Yacht Match Race Outcomes,” *Operations Research*, vol. 52, no. 1, pp. 1–16, Feb. 2004.
- [3] L. Vinckenbosch, “Stochastic Control and Free Boundary Problems for Sailboat Trajectory Optimization,” 2012, publisher: Lausanne, EPFL.
- [4] S. Cacace, R. Ferretti, and A. Festa, “Stochastic hybrid differential games and match race problems,” *Applied Mathematics and Computation*, vol. 372, p. 124966, 2020.
- [5] “ORC sailboat data,” Tech. Rep. [Online]. Available: <https://jieter.github.io/orc-data/site/>
- [6] A. Bensoussan and J. Menaldi, “Stochastic hybrid control,” *J. Math. Anal. Appl.*, vol. 249, no. 1, pp. 261–288, 2000.
- [7] G. Barles and P. Souganidis, “Convergence of approximation schemes for fully nonlinear second order equations,” *Asymptotic Analysis*, vol. 4, no. 3, pp. 271–283, 1991.
- [8] R. Ferretti and H. Zidani, “Monotone Numerical Schemes and Feedback Construction for Hybrid Control Systems,” *Journal of Optimization Theory and Applications*, vol. 165, no. 2, pp. 507–531, 2015.
- [9] M. Abramowitz and I. A. Stegun, Eds., *Handbook of mathematical functions: with formulas, graphs, and mathematical tables*, 9th ed., ser. Dover books on mathematics. Dover Publ, 2013.
- [10] M. Falcone and R. Ferretti, *Semi-Lagrangian approximation schemes for linear and Hamilton-Jacobi equations*. SIAM, 2014, vol. 133.
- [11] R. Ferretti, “A Technique for High-order Treatment of Diffusion Terms in Semi-Lagrangian Schemes,” *Communications in Computational Physics*, vol. 8, no. 2, pp. 445–470, June 2010.
- [12] R. A. Howard and J. E. Matheson, “Risk-Sensitive Markov Decision Processes,” *Management Science*, vol. 18, no. 7, pp. 356–369, 1972.
- [13] W. H. Fleming and W. M. McEneaney, “Risk-sensitive control on an infinite time horizon,” *SIAM Journal on Control and Optimization*, vol. 33, no. 6, pp. 1881–1915, 1995.
- [14] T. Başar and P. Bernhard, *H[∞]-optimal control and related minimax design problems*. Birkhauser, 1995.
- [15] E. Cartee, A. Farah, A. Nellis, J. Van Hook, and A. Vladimirovsky, “Quantifying and managing uncertainty in piecewise-deterministic Markov processes,” *preprint: <https://arxiv.org/abs/2008.00555>*, 2020.
- [16] A. Kumar and A. Vladimirovsky, “An efficient method for multiobjective optimal control and optimal control subject to integral constraints,” *Journal of Computational Mathematics*, vol. 28, pp. 517–551, 2010.
- [17] A. Desilles and H. Zidani, “Pareto front characterization for multi-objective optimal control problems using Hamilton-Jacobi approach,” *SIAM J. Control Optim.*, vol. 57, no. 6, pp. 3884–3910, 2018.

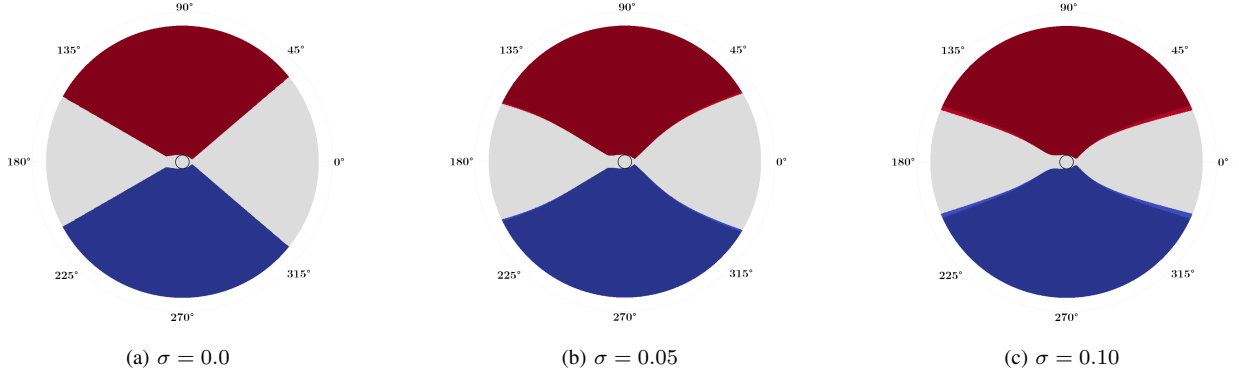


Fig. 3: Switchgrids for stochastic zero-drift ($a = 0$) problems in (r, θ) space. Red points indicate that $\text{switchgrid}(r, \theta, q = 1) = 1$, blue points indicate that $\text{switchgrid}(r, \theta, q = 2) = 1$, and white points indicate that both are 0. Darkened regions correspond to the switchgrids obtained when the zero-evolution switching operator $\mathcal{N}v(r, q) = v(r, 3 - q)$ from [1] is used instead of our (15).

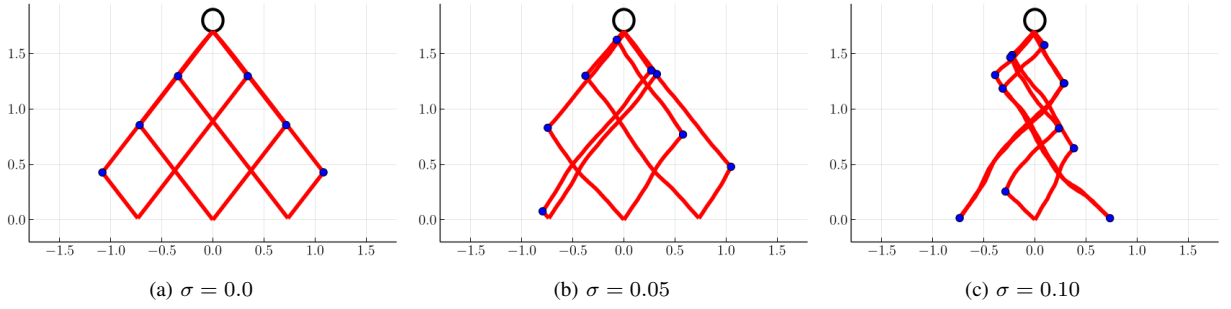


Fig. 4: Six sample trajectories for stochastic zero-drift ($a = 0$) problems: starting from 3 different (r, θ) positions and 2 different starting tacks ($q = 1, 2$). Within each panel, all trajectories use the same sampled wind evolution. Blue points indicate the tack switch events.

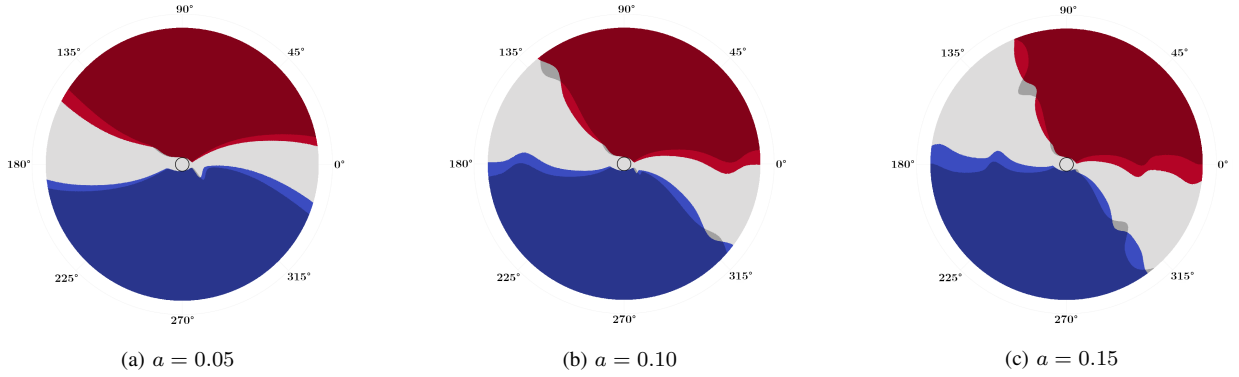


Fig. 5: Switchgrids for stochastic ($\sigma = 0.05$) nonzero-drift problems. Colorings are as in Fig. 3.

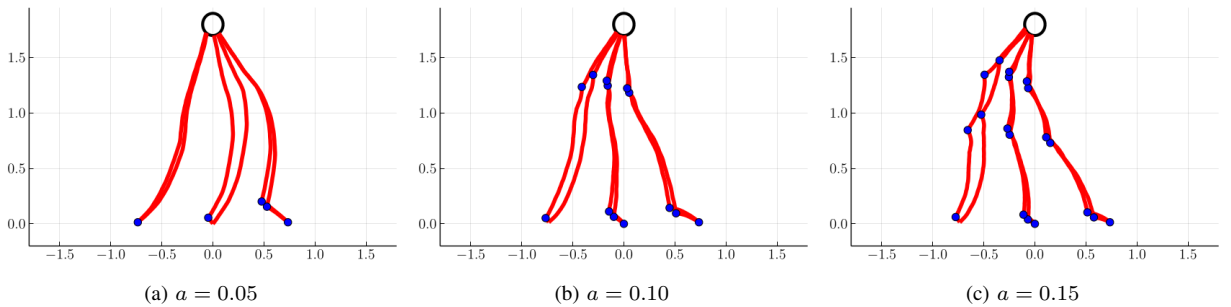


Fig. 6: Six sample trajectories for stochastic ($\sigma = 0.05$) non-zero-drift problems: starting from 3 different (r, θ) positions and 2 different starting tacks. Within each panel, all trajectories use the same sampled wind evolution. Blue points indicate the tack switch events.