Adaptive Speed Control for Autonomous Surface Vessels

Sean Kragelund
Vladimir Dobrokhodov
and Aurelio Monarrez
Department of Mechanical
and Aerospace Engineering
Naval Postgraduate School
Monterey, CA 93943
Email: spkragel@nps.edu

Michael Hurban
United States Navy
Chula Vista, CA 91915
Email: mahurban@gmail.com

Curtis Khol
United States Naval Academy
Annapolis, MD 21402
Email: m143474@usna.edu

***Abstract*—The paper addresses the problem of speed control for the SeaFox unmanned surface vessel (USV). This small, versatile robotic platform can operate over a wide range of speeds, making it attractive for a number of scientific, commercial, and naval applications. This versatility, however, comes at a price. The vessel operates in displacement mode at low speeds and operates in planing mode at high speeds. These two regimes are connected via a highly unstable transition mode, where steady state operation is not possible, making autonomous operations challenging. Speed following is one of the key challenges in automating this class of vessel, as this capability is adversely affected by (i) the inherently slow dynamic response of the propulsion system, (ii) significant variation of the vessel's hydrodynamics in three distinct operating modes, and (iii) significant coupling between these hydrodynamics and the propulsion force. This paper presents a comparative study of three adaptive control algorithms developed for speed-holding capability on the SeaFox USV: (i) classical PID control with gain scheduling, (ii) model reference adaptive control, and (iii) L_1 adaptive control. Beginning with a description of the system identification experiments that informed our understanding of the open-loop plant dynamics, this paper proceeds through controller design and simulation, and presents results from open ocean sea trials. The experimental results provide a basis for an objective comparison of each algorithm's speed following performance and explicitly highlight the benefits of adaptive controllers.**

Keywords*—*Autonomous unmanned surface vehicle; Adaptive control; Velocity control; L_1 adaptation; Model reference adaptive control; PID control; Gain scheduling; Unsteady hydrodynamics; SeaFox USV

I. Introduction

UNmanned systems are playing an increasing role in a wide range of maritime operations. The use of autonomous underwater vehicles (AUVs) in scientific, commercial, and naval applications has become widespread. Unmanned surface vessels (USVs) likewise offer the promise of increased maritime presence at substantial cost savings and increased safety over manned research vessels. Potential marine science applications are many: conducting surveys with towed sensors, tracking pollution or marine mammals, and making in situ observations at the air-sea interface, to name just a few. USVs are also well-suited for dangerous naval missions such as mine hunting/sweeping, maritime interdiction, or force protection operations. USVs can also leverage their surface presence to serve as communications gateways for underwater systems, autonomous robots, and divers. Eventually, autonomous USVs may be expected to launch, recover, and sustain aerial or underwater vehicles without human intervention.

The potential for USVs to collaboratively interact with other vehicles and sensor systems motivates the work in this paper. To perform this task safely, USVs must be able to precisely control their relative position and speed under a variety of system configurations and environmental conditions. This ability is critical in naval operations such as maritime interdiction or force protection in which USVs must operate in close proximity with other vessels. A USV in a riverine convoy, for example, must be capable of matching the speed of the other convoy vessels, potentially over a very wide speed range. Accurate speed control also benefits many oceanographic applications, and can be essential for proper sensor employment. A USV conducting a survey using a towed hydrodynamic depressor and sensor payload, for example, must control the sensor depth via its tow speed.

Speed control of a high speed USV is challenging, since these vessels generally experience three distinct operating regimes. At low speeds, these vessels operate in displacement mode whereby the hull is supported by hydrostatic (buoyancy) forces (Fig. 1). As they accelerate to higher speeds, they pass through a nonlinear, unsteady transition mode before entering the planing mode, where the hull is supported predominantly by hydrodynamic lift (Fig. 2) [1]. The hydrodynamics which govern a vessel's equations of motion differ in each speed regime, and developing a conventional controller for precise regulation over the entire range of speed commands is exceedingly difficult. Modern adaptive control techniques, however, offer a promising approach for enabling robust speed control over a wider speed range. The technical significance is that a single adaptive controller can be designed to accommodate a variety of vessel types, loading configurations, and mission profiles while placing less stress on a vessel's actuators and mechanical structure.

The authors gratefully acknowledge the support of Dr. Tom Drake at the Office of Naval Research via contract N0001413WX20883.

Fig. 1. The Naval Postgraduate School's SeaFox USV operating in displacement mode.

Fig. 2. The Naval Postgraduate School's SeaFox USV operating in planing mode.

In addressing these challenges, this paper presents a methodology for developing and testing a set of adaptive speed controllers for the Naval Postgraduate School's SeaFox USV. First, extensive sea trials were conducted to develop a rich set of experimental data for characterizing SeaFox's three operating regimes and its inherent nonlinear phenomena. Next, system identification was performed to develop a nonlinear mathematical model of the SeaFox speed dynamics. Recognizing that this model still has significant uncertainty, subsequent experimental data was used to ensure that this model represents the maximum achievable speed-holding performance. The resulting model serves two purposes: (i) as a nominal plant for the classical control design approach, and (2) as a reference model for specifying realistic, achievable performance expectations for the adaptive control design approach. Using this model, three different adaptive speed controllers were developed for comparative study: 1) a classical proportional-integral-derivative (PID) controller with gain scheduling; 2) a model reference adaptive controller (MRAC); and 3) an L_1 adaptive controller. Results from numerical simulation and sea trials provided an objective basis of comparison for estimating achievable performance and robustness for each algorithm. Finally, each controller was implemented on the SeaFox USV and tested during a series of experiments on Monterey Bay. The paper concludes by presenting detailed results from these sea trials and suggestions for future research.

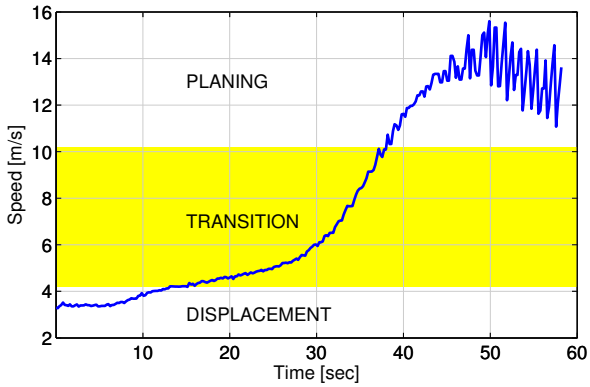

Fig. 3. SeaFox speed vs. time for a 75% throttle command, illustrating three distinct speed regimes.

II. Development of a SeaFox Dynamic Model

A. SeaFox Unmanned Surface Vessel

The SeaFox USV was designed and manufactured by Northwind Marine (Seattle, WA) as a remote-controlled platform for intelligence, surveillance, reconnaissance, anti-terrorism force protection, and maritime interdiction operations [2]. SeaFox is a 5-meter long, rigid-hulled inflatable boat (RHIB). Its water jet propulsion system is powered by a JP5-fueled, 200-HP racing engine and can deliver a top speed of 20 m/s. Standard sensing systems include daylight and low-light navigation cameras for remote operation, as well as twin daylight and infrared gyro-stabilized camera turrets for video surveillance. The Naval Postgraduate School (NPS) modified this platform for fully-autonomous operations by integrating a payload computer with the primary autopilot system; the original remote control link was retained to provide an emergency stop function. NPS algorithms running on the payload computer generate rudder and throttle commands that are sent directly to the SeaFox autopilot. Navigational sensors include a satellite compass that uses differential Global Positioning System (GPS) data for accurate heading information and a tactical-grade inertial measurement unit for precise attitude estimation. The SeaFox supports a number of research programs at NPS, including autonomous riverine navigation using sonar for underwater obstacle detection and avoidance [3].

B. Speed Control Problem Statement

Like most power boats, the SeaFox USV operates in displacement mode at low speeds and planing mode at high speeds, with a rapid, speed-dependent transition between these two speed regimes. Fig. 3 illustrates the highly nonlinear speed response when SeaFox accelerates from rest in response to a 75% throttle command. Note that while this speed profile clearly shows three distinct speed regimes, it is not possible to maintain steady state speeds in the transition regime (by definition). The objective of this research is to develop a uniform control architecture capable of adapting to the unknown hydrodynamics in each operating regime for accurate speed following over a wide range of desired reference speeds, under a variety of environmental and loading conditions.

C. System Identification Approach

Several authors have developed methods for modeling surface vessel dynamics. Triantafyllou and Hover [4] and Fossen [5] develop six degree-of-freedom dynamic models based on physical equations of motion. These models are fairly accurate when the applied forces are known and the vessel operates in the displacement regime, where hydrodynamic characteristics (e.g., wetted area) are approximately constant; they break down as the vessel transitions into the planing regime. Conversely, Savitsky's extensive research [6] on the hydrodynamics of high speed planing craft is based largely on empirical data and a thorough knowledge of a vessel's hull geometry, propulsion performance, and drag characteristics. Lacking the experimental facilities needed to obtain this data for the SeaFox USV, we elected to perform system identification on sea trial data.

First, we collected experimental data during a series of sea trials to map forward speed as a function of several different open-loop throttle commands. In this way, we characterized the three distinct speed regimes and the bounds of the underlying nonlinear phenomena. Next, we performed system identification to develop mathematical models representing the USV speed dynamics in each regime. Finally, we combined these models into a single nonlinear model suitable for controller design and simulation in MATLAB.

1) Experimental Data Collection: We selected Lake San Antonio near Paso Robles, CA for initial speed testing, as this secluded location offered the most benign test environment. Lake San Antonio provided hours of testing free of wind, waves, and currents, while its topography allowed SeaFox to perform long, straight line trajectories to decouple speed dynamics from steering dynamics. These sea trials were designed to capture the transient speed dynamics during various throttle step inputs, and to identify the boundaries between each operating regime. Open-loop throttle commands were provided to the SeaFox autopilot as a percentage of full throttle, accelerating from 0% to 99% throttle, then decelerating to 0% throttle, according to the following test program:

1) step to full throttle, then step down to zero throttle in a single increment
2) step up and step down in 50% throttle increments
3) step up and step down in 25% throttle increments
4) step up and step down in 10% throttle increments
5) identify throttle setting at which transition occurs to the nearest 1% increment

2) System Identification: The speed test results reveal that for a given throttle input command, SeaFox's corresponding output velocity tends to overshoot its eventual steady state value. During system identification, this overshoot behavior can be captured with an underdamped second order linear model. The test results also indicate that the transition between displacement and planing modes begins at different speeds, depending on whether SeaFox is accelerating or decelerating. This hysteresis is clearly shown in Fig. 4, where the red (blue) line indicates the maximum (minimum) speeds achieved for a given throttle command during acceleration (deceleration). The shaded region indicates the variation between the maximum and minimum speeds recorded while operating in planing mode. Moreover, Fig. 4 clearly shows that the planing regime separates into two phases corresponding to acceleration and

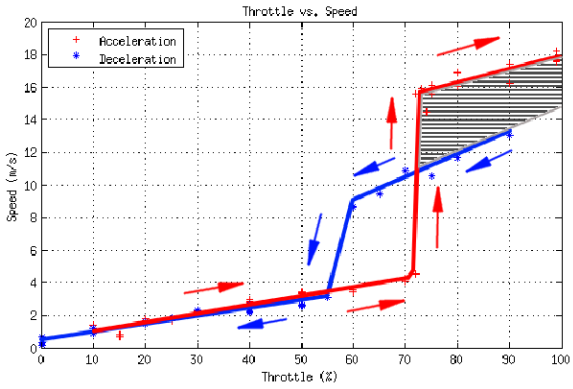

Fig. 4. Output speed vs. throttle input from experimental data, illustrating hysteresis between acceleration and deceleration

deceleration. This hysteresis required two individual subsystems to accurately capture the acceleration and deceleration behaviors within a single speed regime. As a result, a total of six distinct models were used to represent the dynamics for the entire SeaFox speed envelope. To identify these models, the raw test data was first pre-processed using the MATLAB Curve Fitting Toolbox to perform local regression (LOESS) for each sea trial that exceeded 20 seconds in duration. This method uses weighted, linear least squares to fit the data to a second order polynomial [7]. This smoothed data was then used to identify a second order linear model for each subsystem. In this step, we applied two different system identification techniques:

1) using MATLAB's System Identification Toolbox to estimate each process model directly, and
2) using MATLAB's fminsearch function to identify model parameters which minimize the difference between the actual SeaFox speed response and the simulated response of each subsystem model

Finally, the individual models for all six speed regimes were integrated in Simulink to approximate the complete SeaFox speed dynamics. During simulation, logic selects the appropriate speed regime subsystem based on the current speed and throttle setting. Additional details concerning these system identification results and their implementation within the SeaFox model are presented in [8].

III. Controller Design

This chapter addresses the design of three control algorithms representing qualitatively different control architectures: (i) PID, (ii) MRAC, and (iii)L_1 controllers. On the one hand, they represent increasingly complex control architectures in an attempt to address the nonlinear nature of the speed control problem. On the other hand, they demonstrate what can be achieved for the price of increased controller complexity, and highlight the required trade-offs. Each controller was integrated with the SeaFox model discussed in the previous chapter and was simulated both with and without realistic operational disturbances. Comparisons were then performed for simulated reference commands in the displacement and planing regimes. This provided a basis for objective numerical comparison of their relative performance and robustness under challenging

operational conditions. The following sections describe the three control architectures in detail.

A. Proportional Integral Derivative (PID) Controller with Gain Scheduling

A conventional linear feedback controller widely used in industry is the Proportional-Integral-Derivative (PID) controller. This simple and intuitive architecture computes the error between a desired command signal and the measured output of a dynamic system. It operates by producing a control signal which is proportional to the current error, the accumulation of previous errors, and the instantaneous rate of change of the error [9]. As such, this control signal is comprised of a proportional term (P), an integral term (I), and a derivative term (D) as illustrated in Fig. 5. Some well-known considerations for implementing PID controllers in physical hardware include:

- Compensating for integrator wind up that can occur during actuator saturation. Our implementation utilizes back-calculation as described in [9]
- Replacing pure differentiation in the derivative term, D, with a filtered derivative scheme to prevent amplification of high frequency noise.
- Discretizing the controller equations for use in a digital computer.

The mathematical equation for the PID controller output $u(t)$ can be stated in the form:

$$u(t) = K_p \left(e(t) + K_i \int_0^t e(\tau) d\tau + K_d \frac{de(t)}{dt} \right) ,$$

where K_p, K_i, and K_d represent the proportional, integral, and derivative gain values, respectively, and $e(t) = r(t) - y(t)$ is the error between the reference command $r(t)$ and the plant's measured output $y(t)$. This equation can be broken into its constituent parts to explicitly denote the contributions provided by the proportional, integral, and derivative terms as follows:

$$u(t) = P(t) + I(t) + D(t) ,$$

where:

$$P(t) = K_p e(t) ,$$
$$I(t) = K_p \int_0^t K_i e(\tau) d\tau , \text{and}$$
$$D(t) = K_p K_d \frac{de(t)}{dt} .$$

A back-calculation term is added to the integrator's input to prevent wind-up:

$$\frac{1}{T_t} \left(v(t) - u(t) \right) ,$$

where T_t is the tracking time constant governing how quickly the controller output recovers from actuator saturation, when the saturated output $v(t)$ is defined as follows:

$$v(t) = \begin{cases} u_{min}, & u(t) < u_{min} \\ u(t), & u_{min} \leq u(t) \leq u_{max} \\ u_{max}, & u(t) > u_{max}. \end{cases}$$

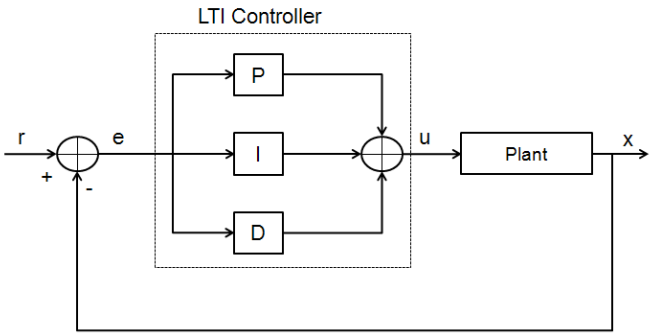

Fig. 5. Block diagram of a basic PID controller [9]

This modification results in a new expression for the integral term:

$$I(t) = K_p \int_0^t \left(K_i e(\tau) + \frac{1}{T_t} \left(v(\tau) - u(\tau) \right) \right) d\tau .$$

Next, we replace the ideal differentiator $de(t)/dt$ in the derivative term with a filtered approximation in the Laplace domain:

$$s \Rightarrow \frac{s}{1 + s\frac{K_d}{N}} ,$$

where K_d/N is the time constant for this 1st order filter. The derivative term thus becomes:

$$\dot{D}(t) = -\frac{N}{K_d} D(t) + K_p N \dot{e}(t) .$$

Finally, the PID controller is implemented in software using finite difference equations in place of the continuous time expressions for $P(t)$, $I(t)$, and $D(t)$.

The above architecture defines a linear controller with constant gain parameters. In general, it will perform poorly outside of the linear operating envelope it was designed for. To address the different operating modes of the SeaFox USV, gain-scheduling logic was implemented so the controller can utilize different PID gain values for each operating mode. In simulation, this logic simply selects a set of high speed PID gains when the USV is traveling faster than 10 m/s; otherwise it defaults to a set of low speed PID gains for displacement mode. Incorporating this simple parameter adjustment scheme into a standard PID controller transforms it into a rudimentary adaptive controller.

B. Model Reference Adaptive Controller (MRAC)

A model reference adaptive control system (MRAC) is typically comprised of four parts as shown in Fig. 6:

1) a reference model with desired output behavior,
2) a plant containing unknown parameters,
3) a feedback control law, and
4) an adaptation mechanism for adjusting the parameters in the control law

An MRAC controller operates by adapting its output signal, $u(t)$, to minimize the difference between the outputs of a known reference model and the uncertain plant being controlled. This error signal, $e(t)$, is fundamentally different from

the error signal computed by the PID algorithm. While both are considered tracking errors, a PID controller computes this signal relative to an external command. Conversely, an MRAC controller computes this signal relative to the output of a reference model. This reference model is used to specify the ideal system response to external commands, within achievable performance limits of the plant and the chosen model structure. For the SeaFox speed dynamics, we selected a first order reference model with a step response settling time of 20 seconds. This model structure was chosen to minimize overshoot in the SeaFox speed response, within performance limits determined from sea trials and system identification. The adaptation mechanism is designed to guarantee asymptotic stability and ensure that the tracking error converges to zero [10]. There are two main ways to design MRAC adaptation laws. The first, called direct MRAC, adjusts the control law by estimating the controller parameters directly. The second, called indirect MRAC, utilizes a non-adaptive control law derived by estimating the plant parameters [11]. The MRAC controller implemented in this work utilizes a composite adaptation scheme, which combines features of both direct and indirect MRAC through the use of state predictors [12]. In the composite MRAC framework, the adaptation laws are augmented to include the state prediction error as well as the model tracking error, as described below.

Consider the following state dynamics and desired reference model for our system:

$$\dot{x}(t) = Ax(t) + bu(t)$$
$$\dot{x}_m(t) = A_m x_m(t) + b_m r(t)$$

where $x(t)$ is the SeaFox forward velocity, $u(t)$ is the throttle input channel, $r(t)$ is the desired speed command, and the reference model parameters (A_m, b_m) were chosen to specify desired system performance. Since the SeaFox autopilot limits throttle command values in order to protect the engine, this system has the following input constraints:

$$u(t) = \begin{cases} u_{min}, & u_c(t) < u_{min} \\ u_c(t), & u_{min} \leq u_c(t) \leq u_{max} \\ u_{max}, & u_c(t) > u_{max}, \end{cases}$$

where $u_c(t)$ is the controller output, but $u(t)$ is constrained to lie within the throttle saturation limits u_{min} and u_{max}. The system dynamics can now be rewritten as:

$$\dot{x}(t) = Ax(t) + bu_c(t) + b\Delta u(t)\,,$$

where $\Delta u(t) = u(t) - u_c(t)$ is a measure of the control deficiency [13]. This quantity is detrimental to the closed loop system's error dynamics, but we can remove its effect by adding a new adaptive gain, $k_u(t)$, to the reference model as shown:

$$\dot{x}_m(t) = A_m x_m(t) + b_m \left(r(t) + K_u(t)\Delta u(t)\right)$$

Similarly, the state predictor in this composite MRAC architecture becomes:

$$\dot{\hat{x}}(t) = A_p\tilde{x}(t) + A_m x(t) + b_m \left(r(t) + K_u(t)\Delta u(t)\right)\,,$$

where $\tilde{x}(t) = \hat{x}(t) - x(t)$ is the state prediction error, and A_p was chosen such that $|A_p| > |A_m|$. This ensures that the state prediction errors converge to zero much faster than

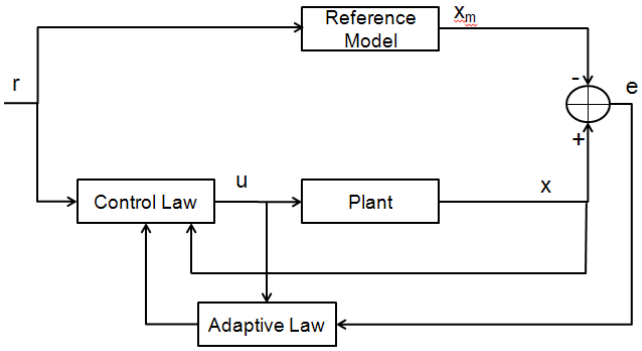

Fig. 6. Block diagram of a model reference adaptive controller (MRAC) [10]

the tracking errors $e(t) = x(t) - x_m(t)$. Finally, a Lyapunov analysis similar to [12] and [13] was conducted in order to prove that the following composite adaptation laws guarantee stability of the closed loop system:

$$\begin{aligned}
\dot{K}_x(t) &= -\Gamma_x x(t)\left(e(t) - \Gamma\tilde{x}(t)\right) sgn(b), \quad & K_x(0) &= K_{x_0}\,, \\
\dot{K}_r(t) &= -\Gamma_r r(t)\left(e(t) - \Gamma\tilde{x}(t)\right) sgn(b), \quad & K_r(0) &= K_{r_0}\,, \\
\dot{K}_u(t) &= \Gamma_u \Delta u(t)\left(e(t) - \Gamma\tilde{x}(t)\right) b_m, & K_u(0) &= K_{u_0}\,, \\
u(t) &= K_x(t)x(t) + K_r(t)r(t)\,,
\end{aligned}$$

where $K_x(t)$, $K_r(t)$, and $K_u(t)$ are the controller gains; Γ, Γ_x, Γ_r, and Γ_u are the adaptation gains; and $u(t)$ is the composite MRAC control law. A detailed Simulink implementation of this controller, including techniques utilized for controlling parameter drift are presented in [8].

C. L_1 Adaptive Controller

The L_1 control design [11] is a relatively new but nevertheless well-established approach in the field of robust adaptive control. While the ultimate goal of the L_1 adaptive controller is still the same (to achieve desired control performance in the presence of significant plant uncertainties and external disturbances), the key difference and main benefit of the L_1 architecture results from decoupling the high adaptation rate from the robustness properties. In the MRAC architecture, the adaptation rate and robustness are closely coupled, thereby leading to significant degradation in robustness when a fast adaptation rate is desired. Thorough details of the L_1 adaptive control formulation and the synthesis of the controller can be found in [11]. Here we briefly present the advantages of the L_1 adaptation approach for the SeaFox application:

- Guaranteed robustness in the presence of fast adaptation
- Separation of adaptation and robustness
- Guaranteed transient response
- Guaranteed time-delay margin
- Uniformly scaled transient response dependent on changes in initial conditions, unknown parameters, and reference inputs

The implementation of the L_1 controller closely follows the architecture presented in [11], [14], and [15]. A block diagram

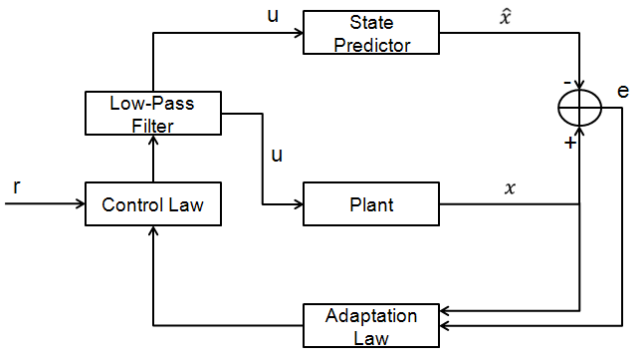

Fig. 7. Block diagram of an L_1 adaptive controller [11]

of the L_1 controller is shown in Fig. 7, providing a simple introduction to the design of the L_1 controller and highlighting its differences from the MRAC architecture.

The L_1 architecture defines a control signal such that the contribution of unknown parameters or unmodeled dynamics are canceled. Unlike the MRAC approach, the uncertain values of the plant parameters are allowed to be time-varying. This is another major advantage of the L_1 architecture over the MRAC architecture. Indirect MRAC allows only slowly varying parameters, while L_1 fast adaptation significantly relaxes this constraint. While the L_1 controller structure in Fig. 7 looks similar to the MRAC structure in Fig. 6, the reference model is not an explicit part of the L_1 controller. Instead, the reference model is accounted for in the design of the state predictor. Furthermore, the L_1 controller has a low pass filter in the control channel, whereas the MRAC architecture does not. The low pass filter and the state predictor are instrumental in decoupling the fast adaptation dynamics from the robustness properties. This decoupling allows a designer to increase the adaptation rate via the adaptation gain, while separately tuning the robustness properties via the parameters of the low pass filter. The chosen filter parameters determine the filter's bandwidth, directly affecting the performance and stability of the system. We consider unmodeled propulsion system dynamics as well as unmodeled internal dynamics in the nonlinear system with time- and state-dependent nonlinearities due to propulsion system interactions with the unsteady vessel hydrodynamics. The dynamics of the SeaFox speed control channel can be written as:

$$\begin{aligned}
\dot{x}(t) &= A_m x(t) + b\left(\mu(t) + f(t, x(t), z(t))\right), x(0) = x_0\,,\\
\dot{x_z}(t) &= g(t, x_z(t), x(t)), x_z(0) = x_{z0}\,,\\
z(t) &= g_0(t, x_z(t))\,,\\
y(t) &= c^T x(t)\,,
\end{aligned}$$

where $x(t)$ is the system state; A_m is a known Hurwitz matrix specifying the desired closed-loop dynamics; b, c are known constant vectors; $y(t)$ is the system output; f is an unknown mapping which represents the unknown system nonlinearities; $x_z(t)$ and $z(t)$ are the state and the output of unmodeled dynamics; g and g_0 are unknown nonlinear continuous functions; and $\mu(t)$ is the output of $\mu(s) = F(s)u(s)$, where $u(t)$ is the control input and $F(s)$ is an unknown, BIBO stable and proper transfer function with a DC gain of known sign.

The system defined above needs to verify a set of assumptions about (A1) the uniform boundedness of $f(t)$, (A2) the semi-global uniform boundedness of partial derivatives of $f(t)$ with respect to time and state, (A3) the stability of unmodeled dynamics of x_z, and (A4) partial knowledge of the actuator dynamics. Thorough details on the implication of these assumptions can be found in [11].

The L_1 control objective is to design a full-state feedback adaptive controller to ensure that the output of the system $y(t)$ tracks a given bounded, continuous reference signal $r(t)$ with uniform and quantifiable performance bounds. To achieve this objective we follow the design guidelines presented in [11]. The elements of the L_1 adaptive controller are introduced below. Consider the following state-predictor:

$$\begin{aligned}
\dot{\hat{x}}(t) &= A_m \hat{x}(t) + b\hat{\nu}(t),\ \hat{x}(0) = \hat{x}_0\,,\\
\hat{\nu}(t) &= \hat{\omega}(t)u(t) + \hat{\theta}^T(t)x(t) + \hat{\sigma}(t),\\
\hat{y}(t) &= c^T \hat{x}(t),
\end{aligned}$$

where $\hat{x}(t) \in R^n$ is the state of the predictor, while $\hat{\omega}(t), \hat{\theta}^T(t), \hat{\sigma}(t) \in R$ are the adaptive estimates.

The adaptive laws are defined via the projection operator as follows:

$$\begin{aligned}
\dot{\hat{\omega}}(t) &= \Gamma \mathrm{Proj}\left(\hat{\omega}(t), -\tilde{x}^T(t)Pbu(t)\right)\ ,\ \hat{\omega}(0) = \hat{\omega}_0\,,\\
\dot{\hat{\theta}}(t) &= \Gamma \mathrm{Proj}\left(\hat{\theta}(t), -\tilde{x}^T(t)Pb\|x\|_\infty\right)\ ,\ \hat{\theta}(0) = \hat{\theta}_0\,,\\
\dot{\hat{\sigma}}(t) &= \Gamma \mathrm{Proj}\left(\hat{\sigma}(t), -\tilde{x}^T(t)Pb\right)\ ,\ \hat{\sigma}(0) = \hat{\sigma}_0\,,
\end{aligned}$$

where $x(t) \in R^n$ is the system state vector, which might not be measured; $\tilde{x} = \hat{x}(t) - x(t)$ is the prediction error; Γ is the adaptation gain, and $P = P^T > 0$ is the solution of the algebraic Lyapunov equation $A_m^T P + P A_m^T = -Q$ for arbitrary symmetric $Q = Q^T > 0$. The projection operator ensures that $\hat{\omega}(t) \in \Omega$, $\hat{\theta}(t) \in \Theta$, $|\sigma(t)| \leq \Delta$ are all bounded.

The control signal is generated as the output of the following feedback system:

$$u(s) = -kD(s)\left(\hat{\nu}(s) - k_g r(s)\right),$$

where $\hat{\nu}(s)$ is the Laplace transform of $\hat{\nu}(t)$, $k > 0$, $k_g = -1/c^T A_m^{-1}$ is the feed-forward gain, and $D(s)$ is a strictly-proper, stable transfer function that implies that

$$C(s) = \frac{kF(s)D(s)}{1 + kF(s)D(s)}$$

will be a strictly-proper, stable transfer function with unity DC gain ($C(0) = 1$) for all $F(s)$. The proof of the performance bounds of this architecture can be found in [11].

The reference also has a chapter addressing the practical tuning challenges of both the MRAC and L_1 controllers and provides explicit guidelines for a practical tuning procedure. Note that the desired speed dynamics for the SeaFox USV were obtained in the system identification step. The transfer function $D(s)$ for the uncertainty is usually chosen as a low pass filter and its bandwidth is adjusted based on the desired trade-off between performance and robustness. $D(s)$, together with the choice of state predictor, guarantees that the L_1 adaptive controller stays in the low-frequency range even in the presence of fast adaptation and large reference inputs. The results for a number of test cases are presented in the next section.

IV. Simulation Results

Each controller was implemented in Simulink to support fast tuning and comparative analysis using the SeaFox speed dynamics model developed during system identification. Extensive simulations compared the speed tracking performance and controller effort required to follow reference commands in the displacement and planing regimes. Additional simulations provided qualitative analysis of each speed controller's robustness to constant and periodic disturbances which simulated currents, winds, and waves. Finally, prior to implementing each control algorithm on the SeaFox autopilot, the C++ code for each controller was tested in a Simulink S-Function to ensure correctness with prior simulations.

Detailed simulation results are presented in [8]. By way of a brief summary, all three controllers were able to maintain reference commands in the displacement or planing regimes when no external disturbances were present. In simulations which applied constant or periodic disturbances, however, the L_1 controller performed best.

While the ability to design, tune, and test each controller in simulation was extremely valuable, perhaps no mathematical model can perfectly capture the type of complex, nonlinear dynamics observed in the SeaFox USV. As expected, all three controllers required additional tuning during sea trials, and their real-world performance did not match their simulated performance. Obviously, there is no substitute for real world testing, and the remainder of this paper will focus on results from recent sea trials.

V. Sea Trial Results

A. Design of Experiment

After implementing each control algorithm on the SeaFox, we scheduled initial sea trials at Lake San Antonio for July 11, 2013. The experimental objectives were to tune each controller for proper operation in the displacement regime; then, schedule permitting, begin tuning the controllers to operate in the planing regime. Obtaining an accurate, robust speed following capability in the displacement regime was our highest priority, since SeaFox operates in this speed range for the NPS riverine research program. Indeed, most of the missions being performed by unmanned maritime vehicles (e.g., surveillance, mapping, or communications relay) take place at slower speeds. During sea trials on Lake San Antonio, under typically benign environmental conditions, we tuned our PID controller to achieve good performance for speeds in the displacement regime. Figs. 8 and 9 show performance results for step commands of 3 m/s, and 4 m/s, respectively. Unfortunately, safety considerations due to the low lake water level and a high volume of recreational boat traffic prevented us from completing our initial engineering runs for all three controllers. Since we were unable to test our algorithms in the planing regime (e.g. above 5 m/s), high speed testing will be the focus of future work.

Meanwhile, follow-up experiments were scheduled on Monterey Bay during July 17-18, 2013. The main objectives for these sea trials were to:

1) complete tuning the MRAC and L_1 adaptive controllers for operation in the displacement regime, and

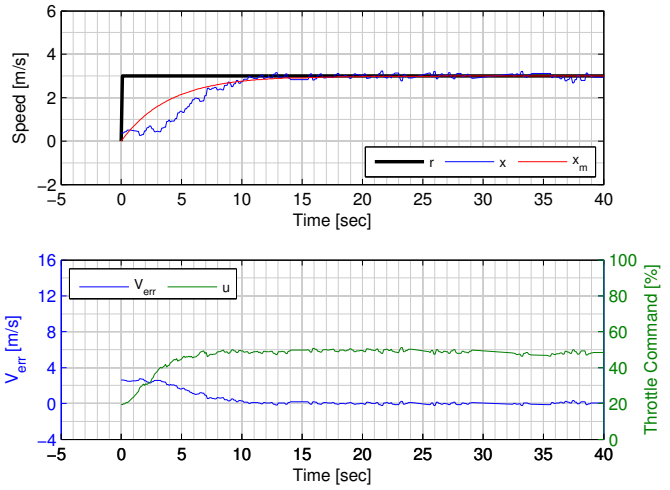

Fig. 8. SeaFox PID controller results from Lake San Antonio for a 3 m/s step command.

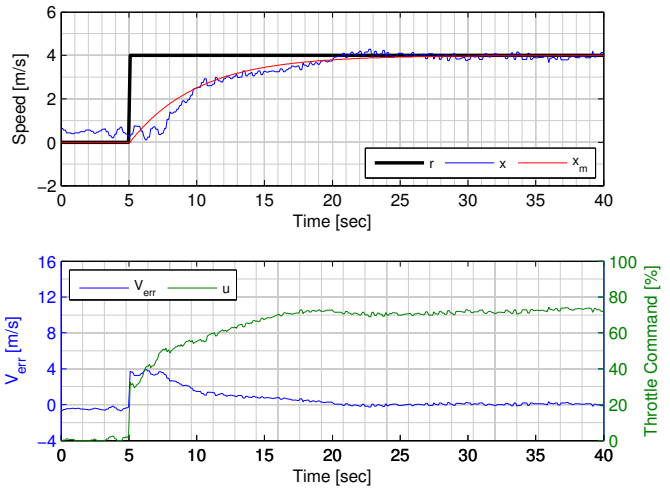

Fig. 9. SeaFox PID controller results from Lake San Antonio for a 4 m/s step command.

2) obtain an objective performance comparison of all three controllers under similar environmental conditions.

Safety is the primary concern when testing experimental control algorithms for unmanned systems. Conducting sea trials on the open ocean benefits from a much larger operating area, with fewer obstacles and less boating traffic, than Lake San Antonio can offer. The Monterey Bay National Marine Sanctuary (MBNMS), however, restricts the use of small jet-powered watercraft like the SeaFox USV to only a few designated operating zones. For these experiments, therefore, SeaFox was programmed to conduct each sea trial with a constant small rudder angle of three degrees. This rudder setting produced large circular trajectories which allowed researchers to safely monitor the experiments from a stationary chase boat instead of following the SeaFox through rough seas. Researchers could also conduct sea trials over longer periods of time without leaving the MBNMS permitted operating area. Ultimately, these trajectories offered the best compromise between safety

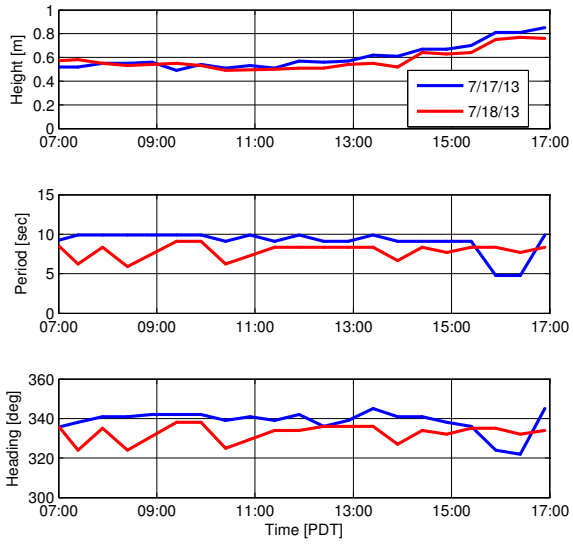

Fig. 10. Surface wave characteristics during Monterey Bay sea trials

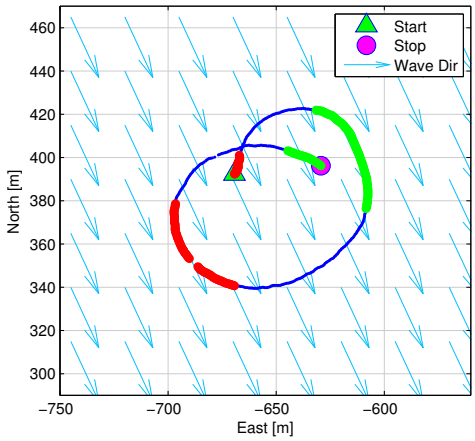

Fig. 11. Typical SeaFox trajectory during Monterey Bay sea trials, showing predominant wave direction.

and the completely-decoupled speed and turning dynamics of a straight line trajectory.

One challenge of comparing controller performance on the open ocean is that wind and wave conditions can change quickly; in general, they are not repeatable from one experiment to the next. On Monterey Bay, the sea surface is typically calm in the morning, with surface waves building in height as wind speed increases in the afternoon. Fig. 10 illustrates the wave conditions present on Monterey Bay during our sea trials, as recorded by a near shore buoy [16]. In this environment, circular trajectories had the additional benefit of subjecting the USV to all potential wind and wave disturbances, regardless of their prevailing directions. Fig. 11 shows a typical sea trial trajectory with the predominant wave direction shown in light blue. Portions of the trajectory heading into the waves are marked in red, while portions of the trajectory with following seas are marked in green.

B. Controller Performance

Experiments were conducted to determine the step response of each controller for speed commands of 3 m/s and 4 m/s. As shown in Fig. 3, transition from displacement to planing mode can begin at speeds as low as 4.2 m/s, so the 4 m/s speed command represents the top end of this operating regime. These sea trials were obtained within an hour of each other on July 18, 2013, so each controller was subjected to similar wind and wave disturbances. The surface waves during these experiments had a mean significant wave height of 0.7 meters, with a peak period of 8.0 seconds, originating from a heading of 334 degrees.

As in Fig. 11, the following step response plots have been annotated to indicate when the SeaFox was aligned with (green bar) or against (red bar) the predominant wave direction. The top plot in each figure shows the speed command (black), output speed (blue), and the ideal step response (red) for the chosen first order reference model: $A_m = -0.2, b_m = 0.2$. While the PID controller does not utilize this model, it has been included in the PID step response plots for comparison purposes. The bottom plot in each figure shows the error signal (blue) and the corresponding controller output (green). This output was sent to the SeaFox autopilot as a throttle command. For the PID controller plots, the error signal shown is the difference between the commanded speed and the output speed. For the MRAC and L_1 controllers, the error signal shown is the tracking error between the actual and reference model output speed. While the L_1 controller utilizes prediction error ($\tilde{x}(t)$) instead of tracking error ($e(t)$) in its adaptation laws, tracking error is shown on the L_1 step response plots for direct comparison with the MRAC response.

Figs. 12, 13, and 14 present step response results for the PID, MRAC, and L_1 controllers, respectively, for a 3 m/s speed command. This speed is comfortably located in the center of the displacement mode operating regime, and all three controllers performed well. The largest oscillations in their response plots are well correlated with an 8-second wave period, especially when the USV is aligned with the prevailing wave direction. Interestingly, following seas created a larger disturbance for each controller than head seas did. Next, in order to compare each controller's relative performance—despite the lack of a true steady state condition in this dynamic environment—we define the following metrics:

$$\overline{e(t)} \equiv \frac{1}{t_f - t_{st}} \int_{t_{st}}^{t_f} e(\tau) d\tau \,,$$
$$\overline{u(t)} \equiv \frac{1}{t_f - t_{st}} \int_{t_{st}}^{t_f} u(\tau) d\tau \,,$$
$$P(t) \equiv \int_{t_{st}}^{t_f} |u(\tau) - \overline{u(\tau)}|^2 d\tau \,,$$

where $\overline{e(t)}$ and $\overline{u(t)}$ are the mean values of the controller's error $e(t)$ and output $u(t)$ signals, respectively. Each metric is computed over the interval from t_{st} to the end of the experiment at t_f. Using these metrics, the mean $\overline{e(t)}$ and standard deviation σ_e of the error signal can be computed for each controller. Likewise, $P(t)$ measures the control effort (in excess of $\overline{u(t)}$) required to maintain the commanded speed in "steady state". These results are presented in Table I. While all

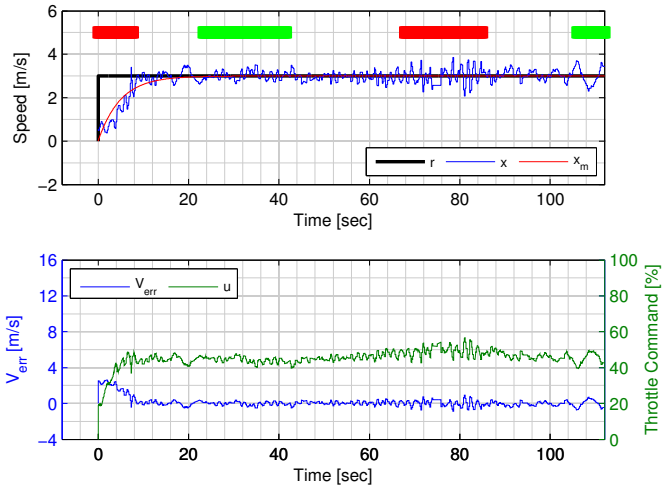

Fig. 12. SeaFox PID controller results from Monterey Bay for a 3 m/s step command

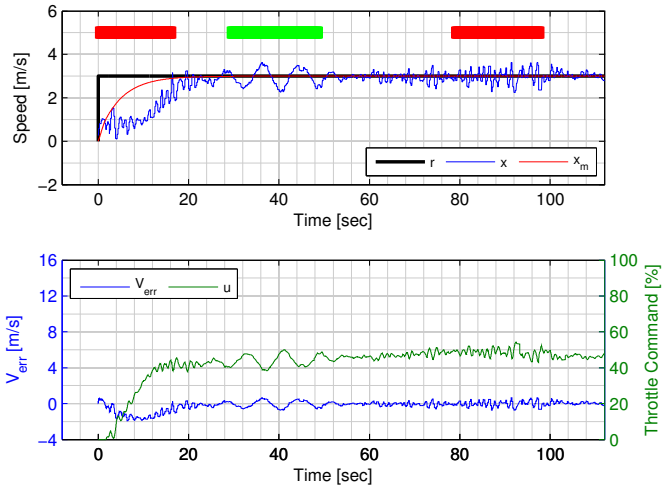

Fig. 13. SeaFox MRAC controller results from Monterey Bay for a 3 m/s step command

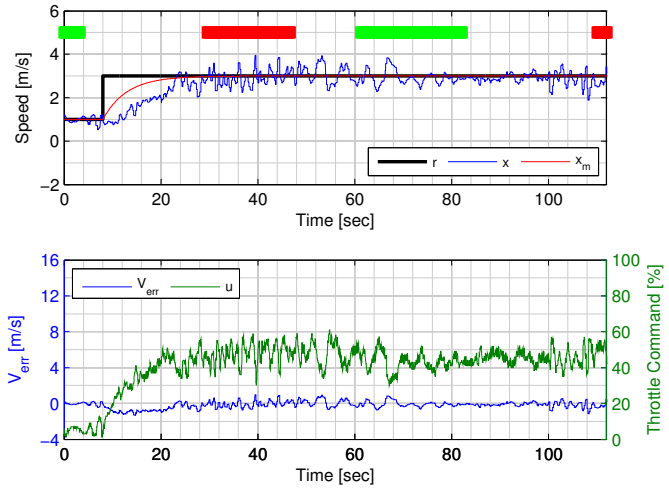

Fig. 14. SeaFox L_1 controller results from Monterey Bay for a 3 m/s step command

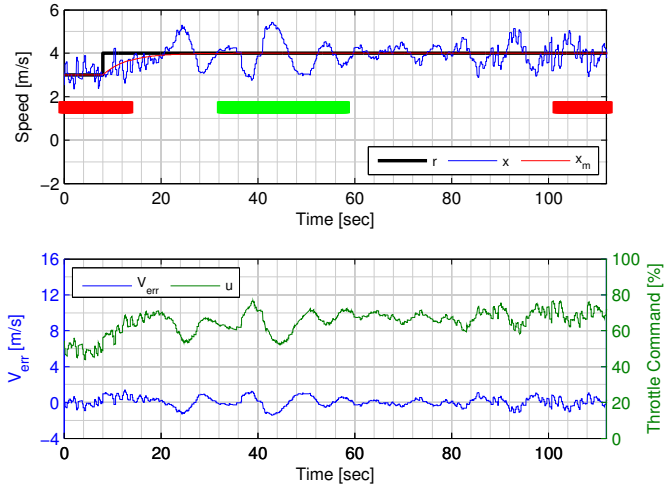

Fig. 15. SeaFox PID controller results from Monterey Bay for a 4 m/s step command

three controllers successfully maintain the commanded speed of 3 m/s, the MRAC and L_1 controllers do so with less error and variation than the PID controller. This is accomplished at a cost of larger control effort. The L_1 controller, however, is less likely than the PID or MRAC controllers to pass low frequency disturbances from wave action to the throttle actuator.

Similar results are shown in Figs. 15, 16, and 17 for a 4 m/s speed command, with performance metrics presented in Table II. This speed represents the upper boundary of the displacement mode operating regime, and the controller requires more effort to maintain the commanded speed. As before, the PID controller is least accurate in terms of the error variation, but also expends the least amount of control effort. For this speed command, the MRAC controller response oscillates, producing the largest throttle values of all controllers and requiring the most control effort. Again, the L_1 controller appears to block transmission of most low frequency disturbances to the throttle actuator.

VI. Future Work

The sea trial results presented above focused exclusively on speeds in the displacement regime, but we remain eager to test our algorithms in other parts of the SeaFox speed envelope. Future work will focus on tuning our speed control algorithms

TABLE I. Performance Metrics for 3 m/s Speed Command

	PID	MRAC	L_1
$\overline{e(t)}$	0.13	-0.04	-0.06
σ_e	0.55	0.28	0.35
$P(t)$	27.5	28.3	52.7

TABLE II. Performance Metrics for 4 m/s Speed Command

	PID	MRAC	L_1
$\overline{e(t)}$	0.04	-0.02	-0.26
σ_e	0.55	0.27	0.52
$P(t)$	43.9	98.7	67.3

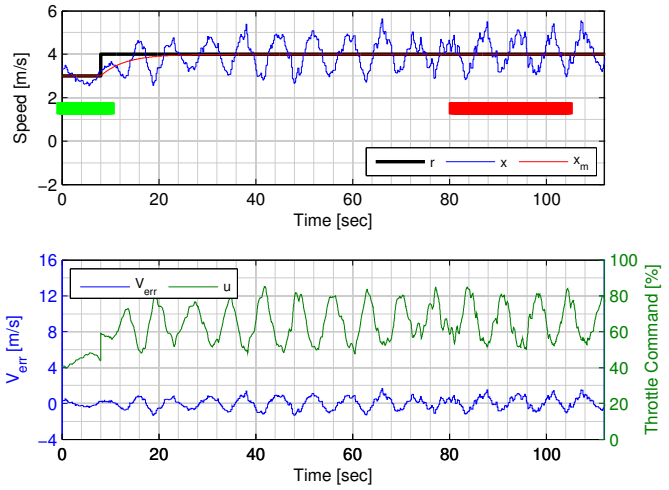

Fig. 16. SeaFox MRAC controller results from Monterey Bay for a 4 m/s step command

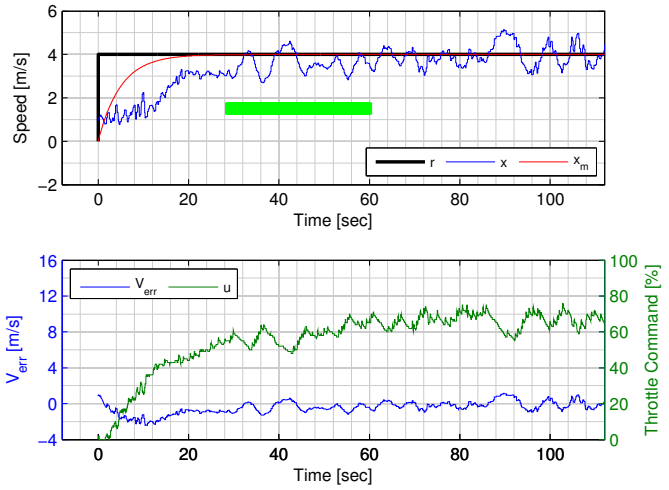

Fig. 17. SeaFox L_1 controller results from Monterey Bay for a 4 m/s step command

for operation in the planing regime and conducting additional sea trials to evaluate their performance at high speeds. This work will also investigate whether PID gain scheduling will allow this linear controller to operate successfully in both regimes.

VII. Conclusion

Adaptive controllers have potential for increasing the utility of unmanned vehicles in a wide variety of applications. However, the main benefit of an adaptive control architecture, namely robust performance under various operational conditions, comes at the price of increased system complexity. This generally requires more development effort, particularly while tuning the adaptive controller, but can lead to improved operational capabilities. This paper describes the use of adaptive control techniques to address the problem of accurate speed control on a small, versatile, unmanned surface vessel characterized by complex speed dynamics. Initial results are promising, but additional testing is required to validate the proposed adaptive control architectures over a wider speed range.

References

[1] R. Marshall, *All About Power Boats*, Camden, ME: Intl. Marine/Ragged Mt. Press, 2002.

[2] Maritime Reporter, *Little USV, Big Applications*. MarineLink.com, Apr. 2004. [Online]. Available: http://www.marinelink.com/news/article/little-usv-big-applications/323082.aspx [Accessed: 29 Jul. 2013].

[3] O. Yakimenko and S. Kragelund, "Real-Time Optimal Guidance and Obstacle Avoidance for UMVs," in *Autonomous Underwater Vehicles*. N. Cruz, Ed., New York: InTech, 2011. [E-book]. Available: http://www.intechopen.com/books/autonomous-underwater-vehicles/real-time-optimal-guidance-and-obstacle-avoidance-for-umvs

[4] M. Triantafyllou and F. Hover, "Maneuvering and Control of Marine Vehicles," Massachusetts Institute of Technology: Cambridge, MA, Tech. Rep., 2002.

[5] T. Fossen, *Marine Control Systems*. Trondheim, Norway: Marine Cybernetics, 2002.

[6] D. Savitsky, "Planing Craft," *Naval Engineers Journal*, vol. 97, no. 2, 1985, pp. 113–141.

[7] J. Yang et al., *Applied Numerical Methods using MATLAB*. Hoboken, NJ: John Wiley and Sons, 2005.

[8] M. Hurban, "Adaptive Speed Controller for the SeaFox Autonomous Surface Vessel," M. Mech. Eng. thesis, MAE, NPS, Monterey, CA, 2012. Available: http://hdl.handle.net/10945/6811 [Accessed 26 Apr. 2013]

[9] K. Åstrom and T. Hägglund, *PID Controllers*, vol. 2. Research Triangle Park, N.C.: Intl. Soc. for Meas. and Control, 1995.

[10] B. Ioannou and P. Fidan, *Adaptive Control Tutorial*. Cambridge, MA: Cambridge University Press, 2006.

[11] N. Hovakimyan and C. Cao, *L_1 Adaptive Control Theory: Guaranteed Robustness with Fast Adaptation*. Philadelphia: SIAM, 2010.

[12] M. Duarte and K. Narendra, "Combined Direct and Indirect Approach to Adaptive Control," *IEEE Trans. on Automatic Control*, vol. 34, no. 10, 1989, pp. 1071–1075.

[13] S. Karason and A. Annaswamy, "Adaptive Control in the Presence of Input Constraints," *IEEE Trans. on Automatic Control*, vol. 39, no. 11, 1994, pp. 2325–2330.

[14] N. Hovakimyan and C. Cao, "Stability Margins of L_1 Adaptive Control Architecture," *IEEE Trans. on Automatic Control*, vol. 55, no. 2, 2010, pp. 480–487.

[15] N. Hovakimyan and C. Cao, "Adaptive Controller for Systems with Unknown Time-Varying Parameters and Disturbances in the Presence of Non-Zero Trajectory Initialization Error," *Intl. Journal of Control*, vol. 81, no. 7, 2008, pp. 1147–1161.

[16] Central and Northern California Ocean Observing System, *Buoy Data for Station 46240–Cabrillo Point, Monterey Bay (158)*. CenCOOS - Ocean Data Portal, July, 2013. [Online]. Available: http:3http://139.121.160.34/CeNCOOS/DataPortal.html