MIT Open Access Articles

Experiments in dynamic control of autonomous marine vehicles using acoustic modems

Experiments in Dynamic Control of Autonomous Marine Vehicles Using Acoustic Modems

Eric Gilbertson, Brooks L. Reed, Josh Leighton, Mei Yi Cheung and Franz S. Hover

Abstract— Marine robots are an increasingly attractive means for observing and monitoring in the ocean, but underwater acoustic communication ("acomms") remains a major challenge, especially for real-time control. Packet loss occurs widely, bit rates are low, and there are significant delays. We consider here strategies for feedback control with acomms links in either the sensor-controller channel, or the controller-actuator channel. On the controller-actuator side we implement sparse packetized predictive control (S-PPC), which simultaneously addresses packet-loss and the data rate limit. For the sensor-controller channel we study a modified information filter (MIF) in a Linear Quadratic Gaussian (LQG) control scheme. Field experiments were carried out with both approaches, regulating crosstrack error in a robotic kayak using acomms. Outcomes with both the S-PPC and MIF LQG confirm that good performance is achievable.

I. INTRODUCTION

Marine robots have played an increasing role in ocean operations over recent years, with the proliferation of many commercial platforms, systems and sensors. The trend is toward tetherless operations, for which each vehicle has to carry its own power source and have a means of wireless communication. Underwater communication over distances beyond about one hundred meters is almost exclusively made through acoustics, and the wireless nature of acoustic communications ("acomms") lends itself naturally to operations with multiple agents.

Acoustic communications bring many challenges, however. Packet loss occurs frequently, caused by ambient noise, multipath, and changing environmental conditions. Even when packets are successful, the data rate can be very low (O(100bits/sec)). The channel is subject to the speed of sound ($\sim$ 1450m/s in water), imposing a significant delay. This delay is exacerbated by duty cycle limitations from the time needed to transmit at low bit rates, guard times to combat multipath, as well as interference effects between multiple users [1]. These bad properties of acomms have limited their use in high-performance, real-time tasks. Such tasks could include traditional dynamic control and, more broadly, groups of vehicles tracking a submarine or marine animal, characterizing an evolving oil plume, or perhaps collaboratively moving a large object.

Our goal in this paper is to investigate several recent schemes that can maximize the effectiveness of acomms in rate-critical missions. As shown in Figure 1, there are at least

E. Gilbertson, J. Leighton, M. Cheung and F. Hover are with the MIT Department of Mechanical Engineering. B. Reed is in the MIT/WHOI Joint Program in Oceanographic Engineering. {egilbert,brooksr8,jleight,mc2922,hover} at mit.edu.
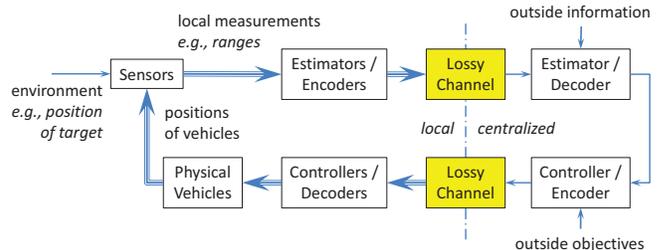
Fig. 1. Block diagram of a feedback system with acoustic communication in the sensor-controller link and the controller-actuator link.

two possible locations for an acomms link in a standard feedback loop: sensor data transmitted to a fusion site (sensor-controller channel), and command data sent to one or more actuated agents (controller-actuator channel). We address both of these. Our approach is aimed towards implementation using commercially-available acoustic modems, such as the WHOI Micro-Modem [2]. As such, we focus on packet loss, rate limits, and delay to stay independent of more detailed descriptions of signal-to-noise ratios, modulation methods, and signal processing.

A number of prior works describe the roles of acoustic modems in controlling AUVs. In [3] Caiti *et al.* characterize packet-loss rates for different packet sizes. In [4], two AUVs perform a coordination task, but vehicles simply stay in their current position if packets are not received. Multiple AUVs with acoustic modems are used for distributed navigation in [5], and low packet loss is ensured by using low data rates.

More fundamentally, the impacts of packet loss and limited data rate can be combined in the information-theoretic capacity of the channel, in a feedback interconnection. For open-loop unstable systems, a number of compact results exist [6], [7]. Few autonomous vehicles are open-loop unstable, however, and more often the goal is performance in following reference trajectories and rejecting disturbances. In this regard, an essential inequality between the Bode integral and the channel capacity was developed by Martins & Dahleh [8]. A familiar engineering approach to exploit a rate-limited channel is quantization [9], [10].

One specific and practical technique for a lossy controller-actuator channel is packetized predictive control (PPC) [11], a variant of model predictive control (MPC). PPC makes a natural fit for packet loss, as an entire receding-horizon control trajectory computed by MPC is sent in each packet. Nagahara and Quevedo articulated PPC using a mixed ℓ_1/ℓ_2 cost function, that addresses the data rate limit by sparsifying the trajectory to be sent [12]. We will call this approach

sparse packetized predictive control (S-PPC). In related prior work within marine robotics, Naeem *et al.* [13] used a genetic algorithm-based MPC for heading control on an AUV, but did not consider any effects of communication constraints or delays.

For packet loss in the sensor-controller channel, Kalman filter-based LQG control with missed information has been studied extensively, *e.g.* [14]. An LQG approach using a modified information filter (MIF) was developed by Gupta *et al.* [15]; the method does not require acknowledgements. Linear coding for estimation under packet loss is considered in [16], but a major drawback is that acknowledgements are required.

We present in this paper experimental results from separate implementations of S-PPC and MIF-LQG in dynamic control of crosstrack error for a robotic kayak using acomms. To our knowledge, neither packetized predictive control nor MIF-based LQG control have been applied to autonomous vehicles.

To justify studying a feedback loop with only one lossy link, we observe that many physical and operational conditions can lead to strong directional asymmetries in acoustic performance. Both directions are uniquely affected by multipath, reflections, obstructions, and ambient noise (e.g., ships, waves) that are ever-present in the shallow-water environment. Acoustic communication channels suffer from interference with long-baseline and other acoustic systems operating in the same frequency band. On-board machinery is also a major issue; local propulsion noise can cause errors on the receiving end. These asymmetries make the treatment of only one lossy channel an interesting and relevant inquiry.

A reliable acoustic link will still impose a delay of course, but this can be handled directly through linear-time-invariant modeling and control. Thus for the current work, our assumption is that one channel is lossy and delayed, while the other is reliable and instantaneous.

The paper is organized as follows. We formulate the S-PPC problem for the controller-actuator channel in Sec. II, and the MIF-LQG problem for the sensor-controller channel in Sec. III. The experimental setup is given in Sec. IV, and results in Sec. V. Crosstrack error can be regulated well with very low data rates, and robustness to packet loss is strong.

II. S-PPC PROBLEM FORMULATION

When a packet is dropped in the controller-actuator channel, common tactics are to simply zero the control or to hold the last valid command. In both cases conservative gains are needed, to stay within a safe operating regime. Alternatives have emerged recently, based on the concept Model Predictive Control (MPC). MPC develops new control trajectories at every time step and executes the first entry; MPC has high interest because the requisite real-time dynamic optimizations can often be run at kiloHertz rates with today's computing power [17]. An MPC variant known as Packetized Predictive Control (PPC) [11] sends a multi-step trajectory on every cycle, and the idea is that if a packet is lost, the buffered commands can be executed

open-loop until a new packet arrives. The approach will usually provide far better performance than zero- or hold-input control, while simultaneously operating near constraint boundaries and enforcing practical inequality constraints via the MPC computations. In accordance with the rate limits of acoustic communication, we have implemented a sparse PPC (S-PPC) that simultaneously addresses loss and data rate constraints [12], via a mixed ℓ_1/ℓ_2 cost function (similar to that used in compressed sensing). Since S-PPC focuses on the controller-side link, we assume that a good state estimate is available instantaneously, and that it contains an acknowledgement of the delayed control packet.

To allow for sparse trajectories, the control actions sent are defined as increments in the heading setpoint ϕ^d for the kayak (see Sec. IV-B). The trajectory **u** is indexed by the time *sent*, and it takes one time step Δt to reach the vehicle. On arrival **u** is written to a buffer **u**', and thus we have $\phi_{k+1}^d = \phi_k^d + u_k'$, whether the packet was received or not.

To lay out the formal optimization problem, let the S-PPC planning horizon be indexed by $\tau = 1, \ldots, T$, and the complete mission by $k = 1, \ldots, N$. The weighting matrix for the terminal state in the horizon is P, and for the rest of the trajectory it is Q. We assume a state space model of the system given by matrices $[A, B, C, 0]$. Control trajectory **u** is computed by solving the following finite-horizon minimization problem at each time step $k \in 1, \ldots, N-1$:

$$\text{minimize} \quad \left\| P^{1/2} \boldsymbol{x}_{k+T+2} \right\|_2 +$$
$$\sum_{k+2}^{k+T+1} \left\| Q^{1/2} \boldsymbol{x}_\tau \right\|_2 + \sum_k^{k+T} \| u_\tau \|_1 \tag{1}$$

$$\text{subject to} \quad \boldsymbol{x}_{\tau+1} = A\boldsymbol{x}_\tau + B\phi_\tau^d, \tau = k, \ldots, k+T+1 \tag{2}$$
$$\phi_{\tau+1}^d = \phi_\tau^d + u_\tau, \ \ \tau = k, \ldots, k+T \tag{3}$$
$$|\phi_\tau^d| \le \phi^{d,max}, \tag{4}$$
$$\tau = k+1, \ldots, k+T+1$$
$$|\boldsymbol{x}_\tau| \le \boldsymbol{x}^{max}, \tag{5}$$
$$\tau = k+2, \ldots, k+T+2$$
$$|u_\tau| \le u^{max}, \ \ \tau = k, \ldots, k+T \tag{6}$$

Constraints (4) set a limit on the commanded heading relative to the trackline, ensuring that the linearizations stay valid. Constraints (5) and (6) are state and control limits. The inputs to the optimization are the most recent state estimate, $\boldsymbol{x}_k$, and the newest heading setpoint ϕ_k^d, which also serves as the control acknowledgment. The short time needed to compute the S-PPC trajectory is included in the outbound transmission delay.

III. MIF-LQG PROBLEM FORMULATION

We study performance of the MIF-LQG strategy proposed in Gupta *et al.* [15] for a lossy link between the sensor and controller. In these experiments, instead of sending the most recent measurements or the innovation to the controller at each time step, we send a specialized information vector. We note that our implementation is a subset of Gupta's algorithm: the entire message is either successfully transmitted

or lost. Let the sensor noise covariance be R_s and the process noise covariance be Q_p. The measurement at time k is z_k, and the control u_k. The control gain matrix is K, designed in this case as an LQR. At each time step we implement the following.

Sensor Side: Initializing suitably, make the following calculations:

$$
\begin{aligned}
Y_{k|k} &= Y_{k|k-1} + C^T R_s^{-1} C \\
\gamma_k &= Y_{k|k-1} A Y_{k-1|k-1}^{-1} \\
\lambda_k &= C^T R_s^{-1} z_k \\
g_k &= \lambda_k + \gamma_k g_{k-1}, \text{ and send it} \\
Y_{k+1|k} &= Q_p^{-1} - Q_p^{-1} A (A^T Q_p^{-1} A + Y_{k|k})^{-1} A^T Q_p^{-1}.
\end{aligned}
$$

Controller Side:

1) Update Y and compute γ_k as in the first two steps on the sensor side. From a zero initial value, update Ψ by

$$ \Psi_k = Y_{k|k-1} B u_{k-1} + \gamma_k \Psi_{k-1}. $$

2) If the message is successfully received, update the state estimate as

$$ \hat{x}_k = Y_{k|k}^{-1}(g_k + \Psi_k). $$

Else, update it as

$$ \hat{x}_k = A\hat{x}_{k-1} + B u_{k-1}. $$

3) Compute the new control $u_k = -K\hat{x}_k$.
4) Evolve Y as in the last step on the sensor side.

IV. EXPERIMENTAL SETUP

A. Hardware and Operating Environment

We perform control experiments on a Wavesport Fuse 35 whitewater kayak, shown in Figure 2, with a $220N$ thrust Minn Kota Riptide thruster under the bow of the boat. The kayak is $1.8m$ long with a mass of $40kg$ including all onboard electronics and batteries. During operation a WHOI Micro-Modem hangs underneath at a depth of two meters. The vehicle runs MOOS-IvP autonomy software [18] integrated with custom control algorithms. Forward speed

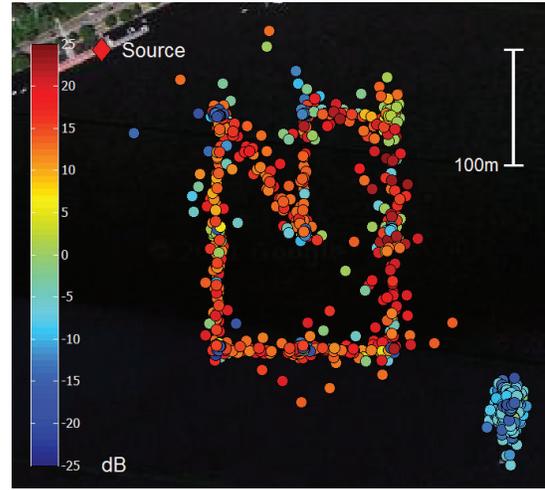Fig. 2. Autonomous surface vehicle operating in Boston Harbor.

Fig. 3. Experimental Micro-Modem performance data in the Charles River Basin, which is not a power-limited environment but rather one limited by multipath. There is a stone wall about $10m$ behind the source, a trench under the low-SNR circles in the lower right, and a shoreline just out of the field of view on the lower right. The source is fixed on a dock, the paths are made by a kayak towing a modem, and the blue cluster is from a second kayak station-keeping. This SNR value indicates sound pressure level relative to ambient noise.
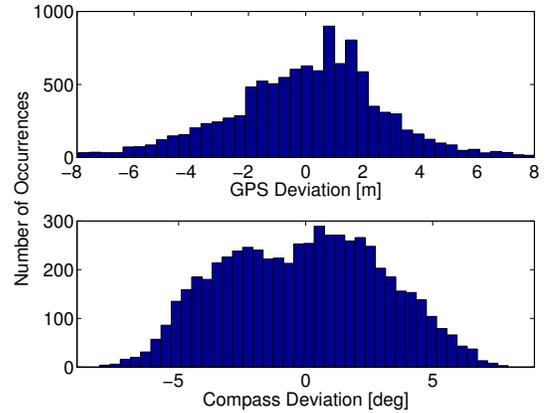

Fig. 4. Example noise histograms from kayak GPS and compass sensors, with the vehicle stationary. The GPS is the uBlox NEO-6 sampled at five Hertz; the variance is $13.3m^2$. The compass is the OceanServer OS5000 sampled at ten Hertz, and then filtered by MOOS at a $1.95Hz$ cutoff; the variance is $9.6deg^2$.

during experiments was $2.1m/s$, and the cycle time was six seconds.

We conduct experiments on the Charles River in Boston, MA, an environment characterized by sporadic acomms behavior, as shown in Figure 3. Packet losses can range from 5% to over 70% depending on conditions and user settings.

The kayak is equipped with a GPS receiver to measure position and tilt-compensated compass to measure heading angle. In designing Kalman filters, we used a sensor noise covariance of $10m^2$ for the crosstrack position based on GPS; raw compass measurements were passed through a first-order low-pass filter having time constant $1.95s$, and we modeled the noise on this signal with a variance of $10deg^2$. Data supporting these settings are given in Figure 4.
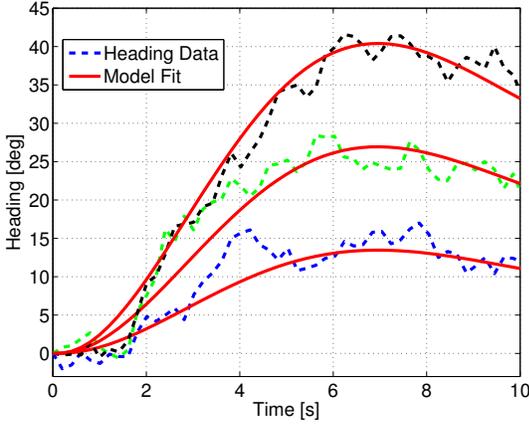
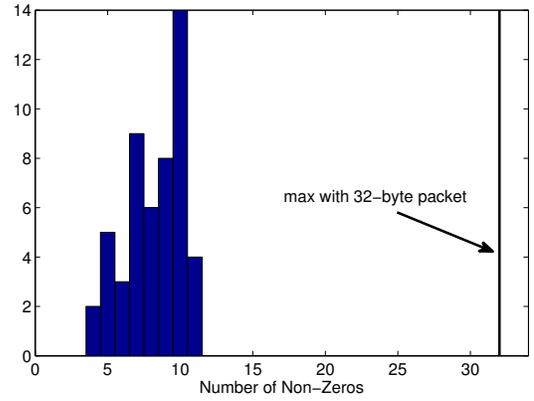Fig. 5. Kayak closed-loop heading model fit on experimental data in waves.

Fig. 6. Histogram of the numbers of non-zero entries in control plans computed by S-PPC, for a run with 50% packet-loss on a calm day, shown in Figure 7. The run was was 52 time steps long, and each plan was 32 steps.

TABLE I

COMPARISON OF THE MOOS AND S-PPC TRACKLINE CONTROLLERS.

Controller	MOOS Trackline	S-PPC
Crosstrack Sensor(s)	GPS@5Hz	GPS@ 5Hz
		Compass@10Hz
Crosstrack Filtering	None	Five-state KF
Control Output	Heading command	Heading command
ϕ/ϕ^d Model	1	3rd-order all-pole TF (7)
Update Rate	0.5s	6s
Control Delay	< 0.5s	6s
Packet Loss	No	Yes

B. System Model

We use the kayak's local PID heading controller for all our experiments, and model the closed-loop behavior as a (stable) third-order, all-pole transfer function

$$\frac{\phi(s)}{\phi^d(s)} = \frac{c_0}{s^3 + c_2 s^2 + c_1 s + c_0}, \qquad (7)$$

with ϕ the heading and ϕ^d the command. The angles are defined as deviations from a fixed trackline heading. We arrived at the fit $c_0 = 1.7$, $c_1 = 2.5$, and $c_2 = 7.5$ in Equation 7, through ten-, twenty-, and thirty-degree step responses, as shown in Figure 5. These traces indicate a rise time of about four seconds, and 30% overshoot. This model holds only for small inputs; the image also illustrates the effect of one-Hertz chop that is typical in our light-air operations.

We use the simplest kinematic model for crosstrack error, assuming that the vehicle is translating only in the direction it is pointing: $\dot{e} = V\sin(\phi) \approx V\phi$. The state vector for the system model is taken as $\boldsymbol{x} = [\ddot{\phi}, \dot{\phi}, \phi, e]^T$, and we convert the system to discrete time with a zero-order hold to obtain the state-space matrices $[A, B, C, 0]$. The loop time Δt is set based on the delay of one acoustic link, transmitting either the control trajectory for S-PPC or the information vector for MIF-LQG. Both of our strategies assume constant time step and delay, valid for acoustic ranges less than one kilometer where message coding and decoding times dominate.

C. S-PPC Experimental Setup

Performance of the S-PPC controller under packet-loss conditions is compared to that of the MOOS Trackline, which is used in a number of marine vehicles today [19], [20] and so provides a reasonable baseline; see Table I. MOOS Trackline is an inner-outer loop that modulates the desired vehicle heading so as to steer it toward a point on the trackline, some lead distance l_d ahead. The result for small errors is a proportional map for desired heading: $\phi^d \simeq e/l_d$, with $l_d = 15m$ a typical value, and ϕ^d in radians.[1]

We made the S-PPC computations on a shore computer with Matlab and CVX [21], [22]. Faster solvers exist for both MPC and mixed ℓ_1/ℓ_2 optimization, such as [23], but CVX was chosen for ease of implementation since the computational delays are still small - about 0.2s - compared to the delays and time step using acoustics. For state estimation during the S-PPC tests we ran a Kalman Filter (KF) at five Hertz onboard, with full access to the vehicle sensors. The state weighting matrix Q was diagonal, with $Q_e^{1/2} = 100$ to strongly penalize crosstrack error and $Q_\phi^{1/2} = 0.1$ to very lightly penalize heading. All other state weightings were zero. The terminal matrix was set at $P = 10Q$, the sparsity parameter $\mu = 1$, and $\phi^{d,max} = 30deg$. State and control limits were subsumed by the desired heading limit.

In S-PPC each packet carries the control actions for every step in the prediction horizon. We chose a prediction horizon length of $T = 32$ steps, or 192s, for our experiments. In real-world marine robots operating with disturbances and physical sensor and actuator limits, we can immediately recognize a practical quantization, so that one byte suffices to describe a single heading command. The idea behind S-PPC, however, is that the ℓ_1 cost function will create sparse control packets that can be encoded even more efficiently, freeing space for other payload data, longer prediction plans, or finer

---

[1]The linear form written is based on approximation of the tangent function. For errors less than one meter, MOOS Trackline controller increases the lead distance proportionally, effectively lowering the gain to limit oscillations.

quantization. A histogram illustrating the very low number of nonzero control actions in each control trajectory for one of the experimental runs is shown in Figure 6. The largest number of nonzero coefficients in a given 32-step plan is eleven, and this requires only 143 bits to code instead of 256, including five time index bits for each nonzero command. Clearly there are many tradeoffs to be considered in the length of the prediction horizon, quantization of control commands, and encoding methods. These are outside our present scope and we refer the reader to [11], [12] for further discussion.

D. MIF-LQG Experimental Setup

For the MIF-LQG experiments we set the LQR state penalty matrix to be diagonal with $\mathbb{Q}_e^{1/2} = 31$ and all other diagonal entries $\mathbb{Q}^{1/2} = 10$, so that error in crosstrack position is most highly penalized. We set the control penalty as $\mathbb{R}^{1/2} = 316$.

E. Physical Connectivity

We conducted two different sets of experiments for each control scheme: one set relies completely on Wi-Fi communication with artificially-imposed packet size and losses, and the other on communication through acoustic modems. In the S-PPC Wi-Fi set of experiments, at every time step the kayak transmits GPS and compass measurements to shore over Wi-Fi, with essentially no delay. The shore computes the new set of control commands and randomly decides whether the packet will be lost, based on a pre-set probability. The shore then simulates a delay, and transmits the successful control packet, or nothing, over Wi-Fi to the kayak. The dual is carried out for the MIF-LQG scheme, with the delay and packet-loss instead imposed in the sensor-controller channel.

In the S-PPC acoustic modem set of experiments, sensor signals are transmitted from kayak to shore via Wi-Fi, and control command packets are sent from shore to kayak using an acoustic modem. The dual is carried out for the MIF-LQG control scheme, with the acoustic modem transmitting between sensor and shore. In the interest of space, this paper focuses on lossy conditions via all-Wi-Fi connectivity; no acoustics results for S-PPC, and only one for the MIF-LQG, are reported.

V. EXPERIMENTAL RESULTS

A. S-PPC

We present S-PPC results from experiments on two different days. Conditions on the first day, 17 October 2012, were relatively calm with about five knots of wind and rippled water; there were intermittent motorboat wakes. The second day, 2 November 2012, was windier, with gusting and shifty following winds around 5-10 knots, resulting in chop.

Results on the first day show how the system performs when disturbances are nearly negligible. Figure 7 gives a comparison of four runs: one run with the MOOS Trackline, and one run each for the S-PPC with no loss, with 50% loss, and with 80% packet loss. Most noticeable, the lack of large physical disturbances brings out the precision limits
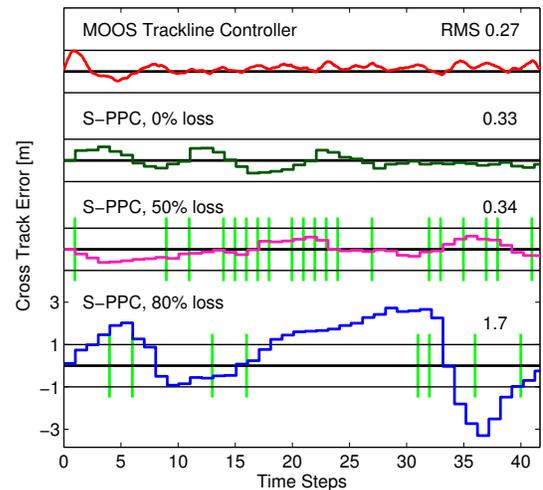

Fig. 7. Comparison of trajectories when using MOOS Trackline and S-PPC controllers, with varying packet loss in calm conditions of 17 October 2012. Green vertical bars indicate successful packets. S-PPC has a six-second update period, while MOOS Trackline runs at two Hertz (i.e., twelve MOOS steps per S-PPC step in the plot).

of our sensors and actuators. The desired heading in MOOS is rounded to the nearest degree, and our thruster servo has backlash and stiction. MOOS Trackline performance is consistent with other works that use similar controllers, sensors and update rates [24], [25], and the S-PPC without packet losses is almost as good. The deterioration in performance with 50% packet loss is minor, which is surprising since there are many twelve-second and longer periods with no crosstrack information. With 80% packet loss, these vacancies cause the vehicle to drift more substantially between hits, although the closed-loop behavior remains stable. Large deviations that might appear to indicate an instability are in fact simply the consequence of large gaps in packet arrival.

Results from the second day, with more wind and waves, are shown in Figure 8. The effect of larger disturbances is visible here through a negative error bias, the result of no integral action. The MOOS Trackline performs well as expected, but the S-PPC with no packet loss has substantially higher crosstrack errors. In one S-PPC timestep the boat is subjected to multiple waves; these not only force it to move laterally but also in yaw (Figure 5), which can confuse the estimator. Higher rates of packet loss lead to worse and worse performance, although again we have seen no evidence of instability in our many trials.

B. MIF-LQG

Crosstrack errors over time are shown in Figure 9 for trials with acoustic-modems on 27 November 2012. Errors are shown in Figure 10 for Wi-Fi-only MIF-LQG trials on 24 October 2012 for calm wind conditions, and in Figure 11 on 16 October 2012 for strong wind conditions.

Results from 27 November act mainly as a demonstration that we can control the kayak with sensor signals via the actual acoustic modem. In this trial packet loss rate was about 5% and wind conditions were zero to five knots.
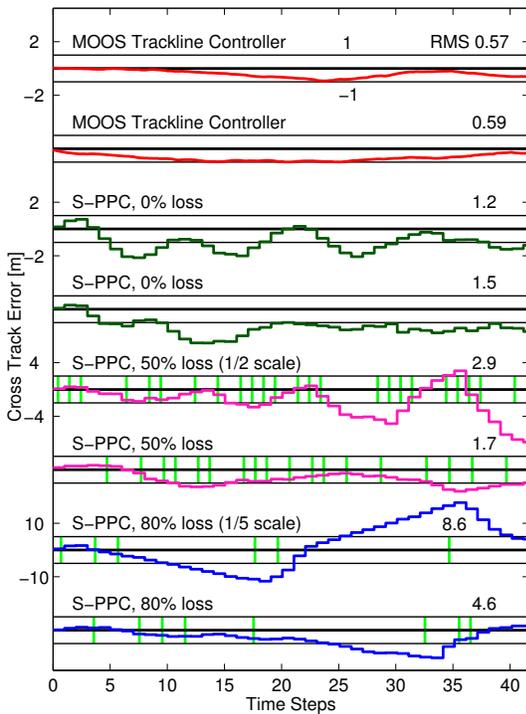
Fig. 8.   Comparison of trajectories when using MOOS Trackline and S-PPC controls with varying packet loss, in moderate wind/wave conditions on 2 November 2012.
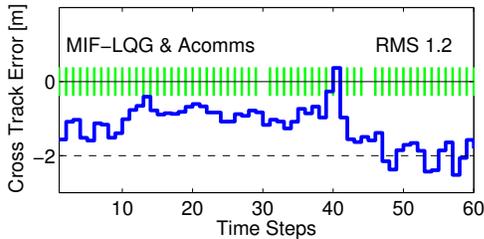

Fig. 9.   MIF-LQG control tests with acoustic communications on a day with light wind, 27 November 2012. Green vertical bars indicate packet successes. Each time step represents six seconds.

24 October had close to zero wind, resulting in very low crosstrack errors; RMS levels were essentially the same for trials with no loss, 30% packet loss, and 70% packet loss. These errors were also significantly lower than that of a run with no crosstrack feedback (labeled Constant Set Heading), as expected. These results corroborate our findings with S-PPC.

16 October had significant chop with sustained winds of fifteen knots, gusting to thirty. As with S-PPC, we see a bias due to the disturbances, and a reduction in control fidelity with increased packet loss. Most dramatically, in the 70%-loss case the vehicle cannot maintain the trackline at all. Underlying this failure is the fact that the boat is operating well outside of the linear model; heading commands from the LQR exceed sixty degrees late in the run. At this signal level, progress toward the trackline is substantially over-estimated, so that corrective actions are too small. Meanwhile, the error
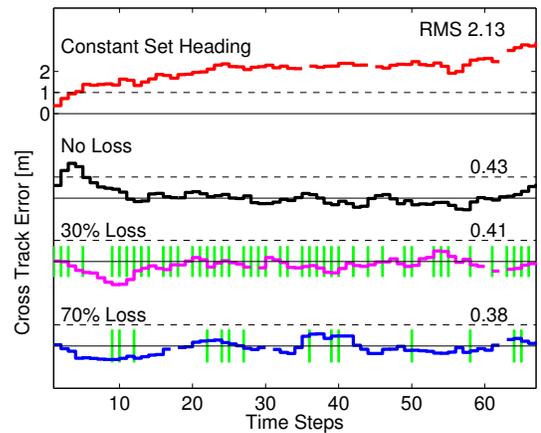

Fig. 10.   MIF-LQG control tests with varying packet loss on a day with nearly no wind, 24 October 2012. Several apparent flyers were removed from this plot (gaps).
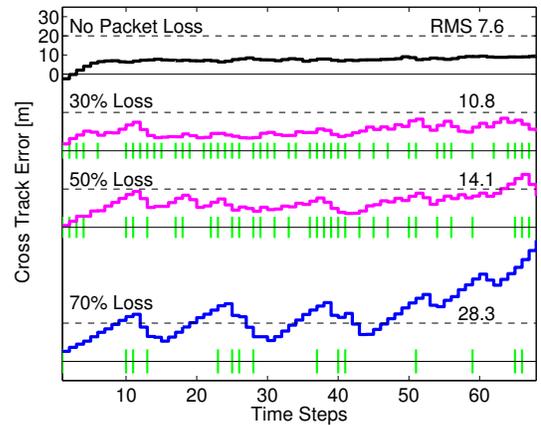

Fig. 11.   MIF-LQG control tests with varying packet loss on a day with strong wind, 16 October 2012.

signal to the controller is erroneously well-behaved in the absence of packets containing crosstrack error information.

VI. Conclusion

We have demonstrated cross-track error control of an autonomous kayak through acoustic communications. For a lossy controller-actuator channel, we implemented Naga-hara & Quevedo's sparse packetized predictive control, and showed that performance under acomms constraints can be comparable to that of standard trackline controllers that have fast update rates, full information and no delays. For a lossy sensor-controller channel , our implementation of Gupta *et al.*'s modified information filter and LQG yielded similarly strong results. These experiments are encouraging, and we expect that implementing both the S-PPC and MIF-LQG concepts simultaneously, for systems with two lossy links, could also yield good performance.

Acknowledgments

Schneider and Mike Benjamin at MIT; Keenan Ball and Sandipa Singh at WHOI; and MIT Sailing Master Franny Charles.

REFERENCES

[1] J. Heidemann, M. Stojanovic, and M. Zorzi, "Underwater sensor networks: applications, advances and challenges," *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 370, no. 1958, pp. 158–175, 2012.

[2] L. Freitag, M. Grund, S. Singh, J. Partan, P. Koski, and K. Ball, "The WHOI micro-modem: an acoustic communications and navigation system for multiple platforms," in *Proc. MTS/IEEE OCEANS*, 2005.

[3] A. Caiti, V. Calabro, G. Dini, A. Duca, and A. Munafo, "AUVs as mobile nodes in acoustic communication networks: Field experience at the UAN10 experiments," in *Proc. MTS/IEEE OCEANS*, 2011.

[4] L. Brignone, J. Alves, and J. Opderbecke, "GREX sea trials: first experiences in multiple underwater vehicle coordination based on acoustic communication," in *Proc. MTS/IEEE OCEANS*, 2009.

[5] A. Bahr, J. Leonard, and M. Fallon, "Cooperative localization for autonomous underwater vehicles," *International J. Robotics Research*, vol. 28, no. 6, p. 714, 2009.

[6] S. Tatikonda and S. Mitter, "Control under communication constraints," *IEEE Trans. Autonomatic Control*, vol. 49, no. 7, pp. 1056 – 1068, July 2004.

[7] G. Nair, F. Fagnani, S. Zampieri, and R. Evans, "Feedback control under data rate constraints: An overview," *Proceedings of the IEEE*, vol. 95, no. 1, pp. 108 –137, Jan. 2007.

[8] N. Martins and M. Dahleh, "Feedback control in the presence of noisy channels: Bode-like fundamental limitations of performance," *IEEE Trans. Autonomatic Control*, vol. 53, no. 7, pp. 1604–1615, 2008.

[9] M. Fu and L. Xie, "The sector bound approach to quantized feedback control," *IEEE Trans. Automatic Control*, vol. 50, no. 11, pp. 1698 – 1711, Nov. 2005.

[10] A. Ribeiro, G. Giannakis, and S. Roumeliotis, "Soi-kf: Distributed kalman filtering with low-cost communications using the sign of innovations," *Signal Processing, IEEE Transactions on*, vol. 54, no. 12, pp. 4782 –4795, Dec. 2006.

[11] D. Quevedo, J. Ostergaard, and D. Nesic, "Packetized predictive control of stochastic systems over bit-rate limited channels with packet loss," *IEEE Trans. Automatic Control*, vol. 56, no. 12, pp. 2854 –2868, Dec. 2011.

[12] M. Nagahara and D. Quevedo, "Sparse representations for packetized predictive networked control," in *Proc. IFAC World Congress*, 2011.

[13] W. Naeem, R. Sutton, J. Chudley, F. Dalgleish, and S. Tetlow, "An online genetic algorithm based model predictive control autopilot design with experimental verification," *International J. Control*, vol. 78, no. 14, pp. 1076–1090, 2005.

[14] B. Sinopoli, L. Schenato, M. Franceschetti, K. Poolla, and S. Sastry, "An LQG optimal linear controller for control systems with packet losses," in *Proc. 44th IEEE Conf. Decision and Control*. IEEE, 2005, pp. 458–463.

[15] V. Gupta, B. Hassibi, and R. Murray, "Optimal LQG control across packet-dropping links," *Systems & Control Letters*, vol. 56, no. 6, pp. 439–446, 2007.

[16] C. Robinson and P. Kumar, "Sending the most recent observation is not optimal in networked control: Linear temporal coding and towards the design of a control specific transport protocol," in *Proc. 46th IEEE Conf. Decision and Control*, dec. 2007, pp. 334 –339.

[17] Y. Wang and S. Boyd, "Fast model predictive control using online optimization," *IEEE Trans. Control Systems Technology*, vol. 18, no. 2, pp. 267–278, 2010.

[18] M. Benjamin, J. Leonard, H. Schmidt, and P. Newman, "An overview of MOOS-IvP and a brief users guide to the IvP helm autonomy software," *Massachusetts Institute of Technology, MIT CSAIL, Tech. Rep. TR-2009-28-07*, 2009.

[19] M. Benjamin, D. Battle, D. Eickstedt, H. Schmidt, and A. Balasuriya, "Autonomous control of an autonomous underwater vehicle towing a vector sensor array," in *Robotics and Automation, 2007 IEEE International Conference on*. IEEE, 2007, pp. 4562–4569.

[20] D. Eickstedt and S. Sideleau, *The backseat control architecture for autonomous robotic vehicles: A case study with the Iver2 AUV*. IEEE, 2009.

[21] M. Grant and S. Boyd, "Graph implementations for nonsmooth convex programs," in *Recent Advances in Learning and Control*, ser. Lecture Notes in Control and Information Sciences, V. Blondel, S. Boyd, and H. Kimura, Eds. Springer-Verlag Limited, 2008, pp. 95–110.

[22] CVX Research Inc., "CVX: Matlab software for disciplined convex programming, version 2.0 beta," http://cvxr.com/cvx, Sept. 2012.

[23] M. Zibulevsky and M. Elad, "L1-l2 optimization in signal and image processing," *IEEE Signal Processing Magazine*, vol. 27, no. 3, pp. 76–88, 2010.

[24] M. Caccia, M. Bibuli, R. Bono, and G. Bruzzone, "Basic navigation, guidance and control of an unmanned surface vehicle," *Autonomous Robots*, vol. 25, no. 4, pp. 349–365, 2008.

[25] C. Kitts, P. Mahacek, T. Adamek, K. Rasal, V. Howard, S. Li, A. Badaoui, W. Kirkwood, G. Wheat, and S. Hulme, "Field operation of a robotic small waterplane area twin hull boat for shallow-water bathymetric characterization," *J. Field Robotics*, vol. 29, pp. 924–928, 2012.