BELL LABS INNOVATIONS

STRATEGIC WHITE PAPER

TOLERATING ADVERSARIES IN THE ESTIMATION OF NETWORK PARAMETERS FROM NOISY DATA: A NONLINEAR FILTERING APPROACH

David T. Stott and Lloyd G. Greenwald
LGS Innovations–Bell Labs

O. Patrick Kreidl and Brian DeCleene
BAE Systems–AIT

Abstract—Estimating network parameters from noisy data is a hard problem that can be made even more difficult by the presence of a malicious adversary who may corrupt the measurement process by capturing a trusted node or perturbing data externally. The adversary may have complete knowledge of the networking protocols that rely on the parameter estimates and may adjust its effect on the system to push protocols into incorrect operating regimes. This work focuses on studying how an adversary may impact the estimation of link quality (LQ) of a communications link. We propose a nonlinear filtering solution that simultaneously tracks both the quality of a link and the state of the adversary, tracking the latter to tolerate better the corruption in tracking the former. We provide empirical results while considering several types of adversarial perturbation, including ones that falsely report the LQ measurements or jam a link. Extensions of these analytical techniques and empirical results show how assumptions about symmetry between the LQ of each direction of a bidirectional link can improve adversary tracking and, in turn, LQ estimation.

TABLE OF CONTENTS

I. INTRODUCTION

Network protocol operation often relies on measured data and subsequent estimation of key network parameters. For example, modulation schemes for wireless networks achieve high rates by adapting to channel conditions estimated from measurements of signal-to-noise ratio (SNR), signal strength, or symbol error rate[1,2]. The accuracy of these estimates directly affects the performance of the adaptive modulation scheme. Similarly, recent Mobile Ad-hoc NETwork (MANET) routing protocols[3,4] make use of estimates of link quality (LQ) to improve throughput over algorithms that rely on only minimum hop count.

Estimation of parameters from noisy data is a classic problem, often addressed with Kalman filters[5] or Bayesian estimators[6,7]. The study of optimal estimation[8] formulates these problems as the minimization of an error metric based on mathematical models of the underlying dynamical system and measurement process. Robust estimation[9] extends solutions to deal with uncertainty in the given models (e.g., unknown variances in Gaussian noise processes), a reality for practical systems.

This paper applies optimal/robust estimation techniques to network monitoring problems with the twist that strategic adversarial influences may affect measurements, directly or indirectly. This introduces the additional tasks of detecting the effects of an adversary in measurements and, when appropriate, removing them so as to prevent detrimental impact to protocol operation. While this problem clearly includes classical parameter estimation as a sub-problem, the focus here is on the new challenges that arise when accounting for the adversary.

To ground our analyses and results, we focus on the problem of estimating LQ, defined as one minus the packet loss probability. MANET control algorithms often rely on LQ estimates (LQEs). For example[3,4], advocate the use of LQ to improve throughput over minimum hop count when constructing overlays. This introduces a potential threat to the MANET: an adversary who is able to manipulate LQEs and has knowledge of the routing algorithm can manipulate the active topology. Such an adversary can affect single-path routing algorithms such as OLSR[10] or DSDV[11] as well as multi-path routing algorithms such as those based on network coding[12]. In network coding, adversarial influence on LQEs can lead to subgraph constructions that under-utilize high-quality links or over-utilize low-quality links, ultimately impacting end-to-end MANET quality of service. For example, the well-studied black-hole attack[13] manipulates LQ to attract traffic that is subsequently dropped. The ability to attract traffic also creates opportunities for network partitioning or malicious network reconnaissance.

We apply nonlinear filtering to track simultaneously the state of the system and the state of the adversary. The contributions of this paper are to (1) formulate the problem of tolerating an adversary affecting the measurement process, (2) provide a nonlinear filtering solution that takes into account ***a priori*** knowledge of potential adversary behavior as well as information about the channel, including any assumption of symmetry in the LQ in bidirectional links, and (3) evaluate the solution using data from a real-world MANET testbed.

Recently[14], demonstrated attacks on ODMRP[15], a high-throughput MANET multicast routing protocol that uses a LQ metric suitable for a multicast setting[16]. They show attacks that effectively exploit the routing protocol's use of LQ to draw traffic toward the malicious nodes. They consider attacks that manipulate LQ locally as well as globally. Our solutions focus on tolerating adversaries in local LQEs; global manipulation can be protected through cryptography. In[14] the authors only consider the case of a single adversary, which manipulates LQ to artificially inflate the quality of selected links. In contrast, our approach can both classify the adversary type and differentiate adversary behavior from noise. Furthermore, instead of relying on an accusation procedure (which may be subject to attack) to explicitly isolate a malicious node, our approach tolerates the adversary's effect.

The paper is organized as follows. In Sec. II we formulate the problem of tolerating adversaries as a sequential mixed-state estimation problem and describe a recursive particle filtering solution. In Sec. III we apply our solution to the LQE problem. We begin by studying the problem with unidirectional measurements and then extend our solution to bidirectional measurements with symmetry assumptions. In both cases, we provide empirical results using simulations driven by data from a real world MANET testbed. Sec. IV discusses our conclusions.

II. SEQUENTIAL MIXED-STATE ESTIMATION

This section casts the problem of tolerating an adversary as a sequential mixed-state estimation problem in a form suitable for standard recursive particle filtering solutions. Our main model, depicted in Figure 1, extends the classical state estimation framework (e.g., see footnotes 6, 8 to account for influences by an adversary. This extended model recognizes that an adversary can influence the estimation process by either (a) affecting the underlying state being measured, e.g., jamming a channel, or (b) affecting the measurements sensed at the node, e.g., a remote node falsifying the number of messages it reports having received. Generally, the adversary attempts to impair the overall system performance by causing the true stochastic process to diverge from what the estimator's models expect. Our goal is to build a ***resilient*** estimator (i.e., one that better tolerates adversary influences), which we accomplish by also estimating the (discrete) adversary state and adapting the estimator's models accordingly. The following subsections elaborate upon the mathematical details.

A. CLASSICAL STATE ESTIMATION

In general, a state estimation problem is defined by two mathematical models, usually referred to as the ***system model*** and the ***measurement model***. The former characterizes the initial ($t = 0$) state and its stochastic evolution over time, while the latter characterizes for each time t how the observable measurement z_t relates statistically to the (hidden) state x_t. Specifically, the process in each time is $t = 1, 2, \ldots$ is described by two equations.

1) A ***system equation*** $x_t = f(x_{t-1}, \omega_t)$, expressing the ***state*** x_t as a function of the preceding state and random disturbance ω_t characterized by a conditional results using simulations driven by data from a real-world characterized by probability distribution $P_{\omega_t|x_{t-1}}$; the initial state x_0 is characterized by probability distribution Px_0.

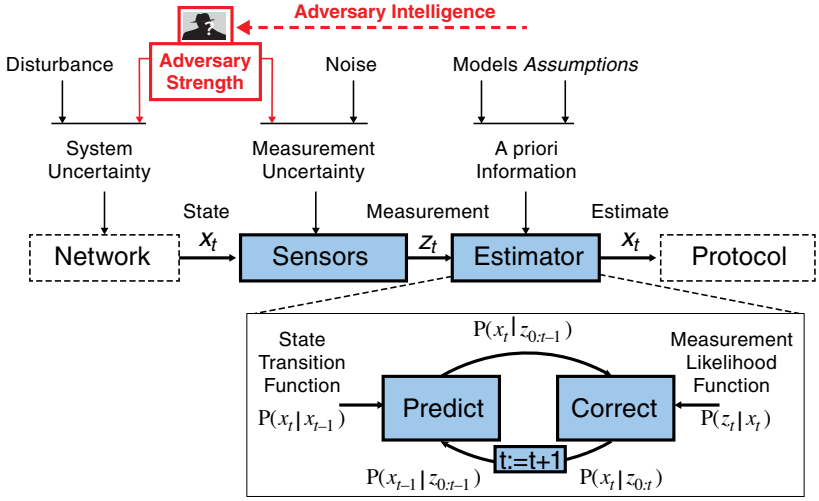
2) A ***measurement equation*** $z_t = h(x_t, v_t)$, expressing ***measurement*** z_t as a function of the current state x_t and random noise v_t characterized by a probability distribution Pv_t.

Here, all disturbance and noise random variables are taken to be mutually independent, implying that the state x_t when conditioned on the preceding state x_{t-1} is independent of all other past random variables. Similarly, the measurement z_t when conditioned only on the current state x_t.

Given these models, the problem at each time t is to design a function γ that maps the history of measurements $z_{0:t}$ to an estimate $\hat{x}$ of the underlying (hidden) state. A given error function $e(\hat{x}, x_t)$ defines the quality of the estimate $\hat{x}$ with respect to the true system state x_t, and the design objective is posed as selecting γ to minimize the expected total error, e.g., the popular minimum-mean-square-error criterion, which corresponds to $e(\hat{x}, x_t) = (\hat{x} - xt)^2$.

It is well known[17,18] that the so-called ***belief state,*** or posterior distribution $P_{x_t|z_{0:1}}$, is a sufficient statistic for optimal estimation. Under the above independence assumptions, the following recursion holds: $P_{x_t|z_{0:1}} \propto P_{z_t|x_t} \int P_{x_t|x_{t-1}} P_{x_{t-1}|z_{0:t-1}} dx_{t-1}$. As depicted in Figure 1, this recursion encapsulates (a) the ***prediction step,*** propagating uncertainty forward via the transition function $P_{x_t|x_{t-1}}$ implied by system equation f, and (b) the ***correction step,*** rebalancing the uncertainty given the latest measurement via the ***likelihood function*** $P_{z_t|x_t}$ implied by measurement equation h. For example, equating the estimate to the (conditional) mean $E[x_t|z_{0:1}]$ based on the belief state is known to minimize the mean-square-error discussed above.

Figure 1 | Recursive Estimation Solution Model



B. ACCOUNTING FOR ADVERSARY INFLUENCE

We account for adversary influences by augmenting the state model to include a discrete-valued component, indexing different types of adversaries. Let s_t denote this (hidden) adversary state at time t, in which case the filtering equation generalizes to $P_{x_t|z_{0:t}} \propto P_{x_t,s_t} \sum_{s_{t-1}} \int P_{x_t,s_{t-1}|x_{t-1},s_{t-1}} P_{x_{t-1},s_{t-1}|z_{0:t-1}} dx_{t-1}$. Assuming the adversary does not change states based its on knowledge of system state x_tt, this leads to separable mixed-state dynamics, i.e., $P_{x_t,s_t|x_{t-1},s_{t-1}} = P_{x_t|x_{t-1}} P_{s_t|s_{t-1}}$. Altogether, in our extended setup, the system model must also define an adversary state transition function $P_{s_t|s_{t-1}}$, and the measurement model must define an adversary-dependent likelihood function $P_{z_t|x_t,s_t}$. Sec. III provides examples of such extensions for LQE.

Closed-form expressions of the filtering equation are available in only certain cases of the underlying models, even without the twist of a mixed-state extension. Particle filtering uses an sample based approximation of the belief state to circumvents this limitation. The implementation in Sec. III implements a simple form of particle filtering, Sequential Importance Resampling[19].

III. LINK QUALITY ESTIMATION

This section applies the concepts from Sec. II to the problem of resiliently estimating time-varying LQ of a channel between two nodes. The job of a resilient estimator is to estimate accurately the LQ even when an adversary is present. Our simulation-based analysis includes two sets of experiments, which use the same four adversary types.

A. BASIC EXPERIMENTAL SETUP

If $x_t \in [0, 1]$ denotes the LQ at time t for the channel between two nodes, each message between them is dropped with probability $1–x_t$, independently of any other message. The first set of experiments (Sec. III-B) involves a mea- surement process whereby one node estimates LQ based on reports from the other node on the channel. With this simple measurement process, two of the four adversaries have identical impacts on the measurements; hence no estimator can discriminate between them without addi- tional information. The second set (Sec. III-C) enhances the measurement process by combining reports from the remote node with locally generated reports. This allows the estimator to discriminate between all four adversary types and to use correlation between the directions to improve accuracy.

Both sets of experiments consider the same four types of adversaries, each with strength α (0, 1) as follows:

0) inactive: the adversary exerts no influence.
1) over-reporter: the adversary reports that each dropped message was received with probability α.
2) under-reporter: the adversary reports that each received message was dropped with probability α.
3) external jammer: the adversary drops each message with probability α.

We say that the false-reporters are in the ***measurements*** because they alter only the reports, whereas the jammer is ***in the system*** because it affects the actual link. In all experiments, the adversary is initially inactive (i.e., s_0 = inactive = 0). In some experiments, the adversary dynamics follow a preset pattern; others follow a simple 4-state Markov process with

$$P(s_t \mid s_{t-1}) = \begin{cases} 1 - 1/\lambda & \text{if } s_{t-1} = s_t = 0 \\ 1/3 & \text{if } s_{t-1} = 0 \text{ and } s_t \neq 0 \\ 1 - 1/\mu & \text{if } s_{t-1} \neq 0 \text{ and } s_t = s_{t-1} \\ 1/\mu & \text{if } s_{t-1} \neq 0 \text{ and } s_t = 0 \\ 0 & \text{otherwise} \end{cases}$$

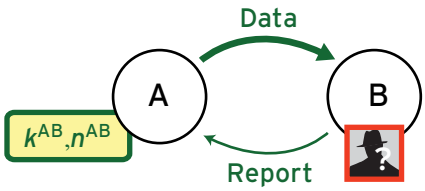
Positive-valued parameters μ, λ, and α are specified as part of the experiment setup.

The experimental analysis uses a C++ simulator for a 24-node MANET driven by real-world, proprietary traces DARPA provided as part of a larger 3-hour military training exercise. The traces specify each node's location and the pathloss between them on a per-second basis. The results in this paper are based on the last 15-minute time slice, in which the nodes converge upon a common location at up to 35 mph (and up to 55 mph for one UAV). In each experiment, the simulator (i) reads in a particular trace, (ii) sequentially executes the measurement, adversary and estimation processes, and (iii) evaluates the estimation error. Repeated experiments of the same scenario are used to assess on-average estimation performance.

B. UNIDIRECTIONAL LQE PROBLEM

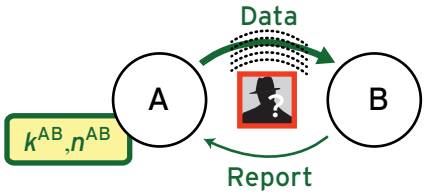
1) ***Unidirectional Measurement Process:*** The measurement process, as depicted in Figure 2, includes a report for every timestep, t. Node A sends regular traffic over the channel. Every message from node A includes a sequence number, which node B uses to compute the number of messages A sent, n_t^{AB}. Node B sends a LQ report to A with n_t^{AB} and the number of messages it received, k_t^{AB}. For simplicity, we assume that the report is transmitted reliably.

FIGURE 2 | LQ MEASUREMENT PROCESS WITH ADVERSARY IN THE MEASUREMENTS



An adversary at node B influences the estimation process at node A by lying about k_t^{AB} in the report. In order to tolerate an adversary in the measurements, a resilient estimator should ***invert*** the effect of the adversary to obtain the LQE that correctly predicts the message success probability. By contrast, an adversary ***in the system***, as depicted in Figure 3, alters the probability of success for all messages in the system. Hence, a resilient estimator should detect the adversary in the system but should ***track*** (rather than invert) the effect.

FIGURE 3 | LQ MEASUREMENT PROCESS WITH ADVERSARY IN THE SYSTEM



2) ***Unidirectional LQE Filter Model:*** The filtering estimator models the system dynamics using two time-varying parameters, the LQ, x_t, and its derivative, $\frac{dx_t}{dt}$. Initially x_0 is drawn from a Beta distribution and $\frac{dx_0}{dt} = 0$. The derivative follows a variation of a random walk as $\frac{dx_t}{dt} = \frac{dx_{t-1}}{dt}$ + Gaussian(0, $\sigma_{\dot{x}}^2$). When the formula forces x_t above 1 or below 0, the model truncates x_t to 0 or 1 and resets the derivative to 0 (otherwise the model would tend to ***stick*** at 0 or 1).

To model the adversary dynamics, the filter model defines a_t, the probability vector of being in each adversary state at time t. Although the truth process has $s_0 = 0$, the filtering model generalizes to any initial distribution, a_0.

3) ***Empirical Unidirectional LQE Results:*** This section provides simulation results for the unidirectional LQE problem. The results are compared against a non-resilient baseline estimator that computes LQ as the average of the 10 most recent measurements. Table I shows the default simulation parameters for the system process.

TABLE I | SYSTEM PROCESS PARAMETERS

PARAMETER	VALUE	DESCRIPTION
p	25	Average message rate
α	0.5	Adversary strength
λ	100	Expected time in inactive adversary state
μ	50	Expected time in active adversary state

The number of message sent k_t^{AB} is geometrically distributed with mean p. Because each message is received with probability x_t, the number of messages received follows a binomial distribution: $k_t^{AB} \sim Binomial(n_t^{AB}, x_t)$.

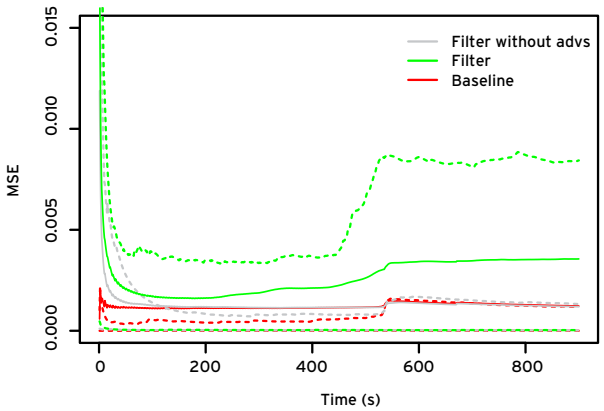
Table II shows the default parameters for the filter. Because the estimator relies on imprecise ***a priori*** knowledge of the system parameters, some values are sub-optimal (e.g., λ, μ, and a_0).

TABLE II | FILTER MODEL PARAMETERS

PARAMETER	VALUE	DESCRIPTION
$N_{particles}$	2000	Number of particles
σ_{dx}^2	.001	Variance for random walk on
σ_{x^2}	.001	Variance for random noice
r, s	1,1	Beta distribution parameters for
α	25	Average message rate
λ	0.5	Adversary strength
μ	100	Expected time in inactive adversary state
a_0	[¼,¼,¼,¼]	Initial adversary state probability vector

Figure 4 demonstrates the accuracy of the filtering a proach without an adversary and the penalty for being resilient. The simulator executes 5 runs from node 10 to each of the other 23 nodes. For each run, the simulator computes MSE as a function of time. It then computes the mean and 10th and 90th percentiles over the runs.

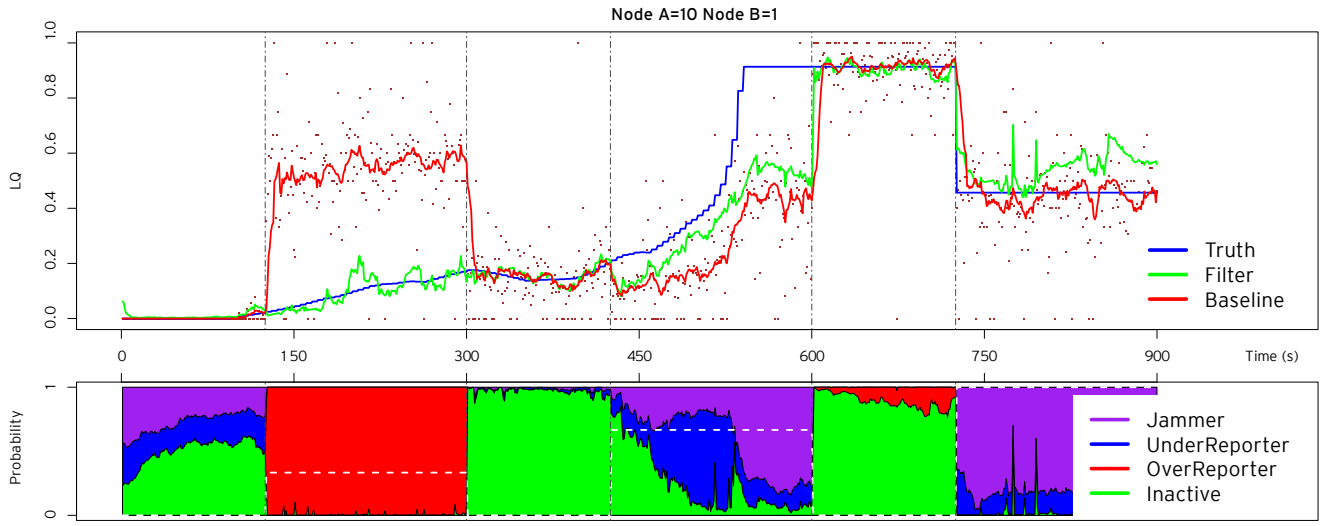
FIGURE 4 | FILTERING APPROACH ACCURACY WITHOUT ADVERSARY



The red line is the (non-resilient) baseline. The gray line is standard filter with the extra knowledge that the adversary is inactive. The two approaches behave similarly. Next, the green line is the standard filter allowing for the possibility of an adversary. Although the green line performs up to a factor of two worse than the baseline, the penalty is reasonable if the approach is resilient against the adversaries.

To understand better how the filtering approach works, consider the single run between nodes 10 and 1 in Figure 5. The adversary cycles through all three active adversaries, one every 300 seconds. The top part of the plot shows the true LQ (x_t) in blue, the baseline estimate in red, and the filter's $\hat{x}_t$ in green. The reported measurements, as $z_t = k_t^{AB}/(n_t^{AB}$, are shown as dots. The bottom part shows the filter's adversary state probability vector, a_t , in solid colors. Each color corresponds to a different adversary where the height of the color at time t equals $a_t[s]$, the computed probability that the adversary is in state s at time t. The dashed white line indicates the true adversary state (0 for inactive, 1/3 for over-reporter, etc.).

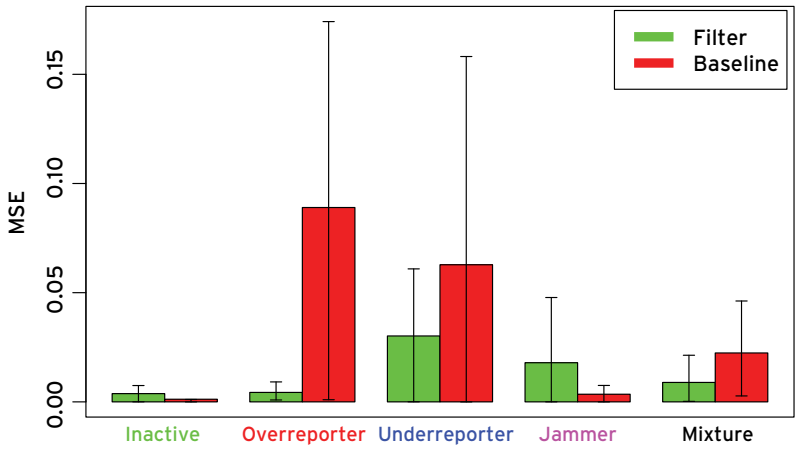
FIGURE 5 | EXAMPLE SINGLE RUN FOR SINGLE-LINK PROBLEM



For $t < 100$, the LQ (truth) is zero. Though the filter correctly predicts the LQ, it cannot determine the correct adversary state (because inactive, under-reporter, and jammer would all report $k = 0$ messages). At $t = 125$, the over-reporter adversary becomes active. The baseline (red) follows the corrupt measurements. The filter immediately detects the adversary and inverts its effect. At $t = 300$, the adversary turns off; the filter quickly determines this with high probability. For $425 < t < 600$, the under-report is active while the nodes approach each other. The baseline fails to track the truth accurately. The filter is considering several possibilities, as indicated by the size of the green, blue and purple areas in the lower plot. One possibility is that there is no adversary: due to the apparent drop in x, the filter gives this a low probability. Another possibility is that either the under-reporter (blue) or jammer (purple) is active. Toward the end of the period, the filter determines that an adversary is active, but favors the incorrect one. The filter chooses $\hat{x}_t$, which is less than x_t, but closer to it than the baseline. For $725 < t < 900$, the jammer adversary is active, causing the truth to appear to drop. The baseline correctly tracks this adversary, which is in the system. The filter must consider two possible adversaries, jammer and under-reporter. Because it cannot tell the two possibilities apart, it should ideally assign them equal probabilities.

Lastly, a summary plot in Figure 6 quantifies the improvement of the filtering approach over the baseline. As before, each data point combines 5 runs from node 10 to each of the other 20 active nodes. The baseline and the resilient estimator run in parallel such that they see identical measurements ($\boldsymbol{z}_{0:T-1}$). The boxes and whiskers indicate the mean, 10th and 90th percentiles of the final ($\boldsymbol{t} = \boldsymbol{T} - 1$) MSE of each run. The boxes for the filtering approach are green; the baseline's are red. The results are repeated first for each of the four adversary types and for a random mixture of all the adversaries (according to the transition formula in Sec. III-A). The rightmost boxes use a random mixture of the adversaries. The filter has a average error of .0089 compared to the baseline's .022.

FIGURE 6 | FILTERING APPROACH IMPROVEMENT OVER BASELINE



The graph shows that the baseline performs poorly against the over-reporter and under-reporter adversaries. It does well against the jammer because it tracks the measurements. The filter performs worse than the baseline when the adversary is inactive (as was shown in Figure 4). It does surprisingly well against the over-reporter. Because the filter cannot distinguish between the under-reporter and the jammer, it chooses an average value that performs moderately well for either.

C. BIDIRECTIONAL LQE PROBLEM

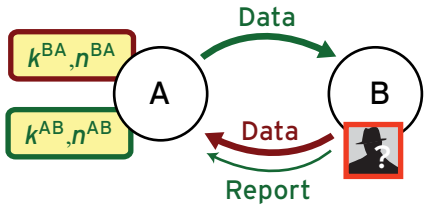
The bidirectional LQE problem uses measurements with reports for both directions of a channel to differentiate between the under-reporter adversary and the external jammer. As before, the error (MSE) is computed for the LQ in the A to B direction.

The DARPA data set yields a single LQ between a pair of nodes, independent of the direction. To realistically simulate a bidirectional channel (e.g., to account for co-channel interference in frequency duplex systems), asymmetric, dependent noise is needed. We use bivariate Gaussian noise to add correlated walks to each direction. Formally, we introduce the sequence of noise vectors as, $\begin{bmatrix} \omega_t^{AB} \\ \omega_t^{BA} \end{bmatrix} \sim \text{Gaussian}\left(\begin{bmatrix} \omega_{t-1}^{AB} \\ \omega_{t-1}^{BA} \end{bmatrix}, \sigma^2 \begin{bmatrix} 1 & \rho \\ \rho & 1 \end{bmatrix} \right)$. The correlation, ρ takes values from 0 to 1, i.e., from independent to identical walks. The new LQ pair is the sum of the single link case and noise as $(x_t^{AB}, x_t^{BA}) = (x_t + \omega_t^{AB}, x_t + \omega_t^{BA})$ truncated to fit [0,1].

1) ***Bidirectional Measurement Process:*** Figure 7 illustrates the measurement process. Both nodes send data simultaneously and locally record the number of messages received and the total number of messages and report the result to each other. Node A combines its local and reported measurements into $\boldsymbol{z}_t = \{k_t^{BA}, n_t^{BA}, k_t^{AB}, n_t^{AB}\}$.

The false reporter adversaries alter the reported measuremen k_t^{AB}, and n_t^{AB}, but not the reverse link k_t and n_t. Only the jammer affects the LQ in the B to A direction.

FIGURE 7 | BIDIRECTIONAL LQ MEASUREMENT

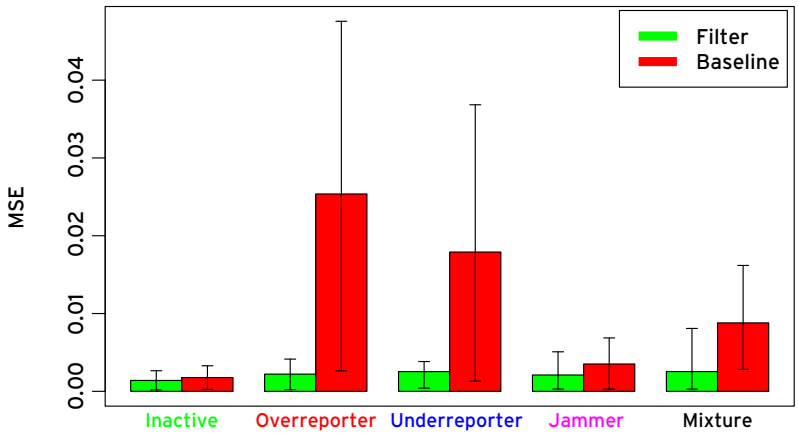


2) ***Bidirectional LQE Filter Model:*** The baseline estimator combines the individual measurement reports as $k_t = k_t^{AB} + k_t^{BA}$ and $n_t = n_t^{AB} + n_t^{BA}$. The filtering estimator's belief state expands to track k_t^{AB} and k_t^{BA}. Though one could track separate σ_{dx}^2 for each direction, the implementation uses a single value for both directions. In the update phase, the filter draws bivariate random noise with parameters σ_ω^2 = .001 and ρ = .75, whereas the system process uses σ^2 = .005 and ρ = .75.

3) ***Empirical Bidirectional LQE Results:*** The experimental procedure follows from Sec. III-B.3. The summary plot in Figure 8 quantifies the improvement of the filtering approach over the baseline for the bidirectional LQE problem.

The results show that the filtering approach accurately predicts and compensates for the adversary state. The filter performs slightly better than (though comparably to) the baseline for the inactive adversary and the jammer. For all other cases, the resilient filtering approach tolerates the adversary that easily confuses the baseline approach.

FIGURE 8 | FILTERING APPROACH IMPROVEMENT FOR BIDIRECTIONAL MODEL



IV. DISCUSSION

We have provided a nonlinear filtering approach for estimating network parameters from data that tolerates the presence of adversaries affecting measurements. By inverting the impact of an adversary on parameter estimation, we prevent the adversary from manipulating higher-level networking protocols. Our approach is based on the formulation of an optimal Bayesian filter that tracks both system state and adversary state, and on the use of a particle filter to tackle the nonlinear models inherent to packet-based networking. Moreover, the particle filtering solution is extensible to a wide range of system, measurement and adversary models.

We focused our efforts on the problem of link quality (LQ) estimation, illustrating how to (i) model a variety of adversaries, and (ii) detect and respond to each adversary type. Our empirical results show that, in the absence of an adversary, the filter performs comparably to a simple base-line algorithm that ignores the adversary; in the presence of an adversary, the filter simultaneously tracks the system state and adversary state to provide resilience, in the sense that our LQEs are more accurate than those of the baseline algorithm. While our solution can estimate LQ solely from unidirectional measurements of the link, it is seen to achieve greater resilience when link symmetry assumptions hold that allow the filter to combine measurements from both directions.

There are a number of interesting problems for future work. In the LQ estimation experiments discussed here, key model parameters were chosen manually (e.g., parameter ***Nparticles*** must be large enough to allow for adequate approximation of the belief state, yet computation time grows linearly with the number of particles)—methods to tune algorithm parameters automatically would be valuable. Note that configuration used for Figure 8 takes about 2 ms per simulated second on a commodity laptop. It would also be interesting to apply our techniques to other parameter estimation problems that arise in networking, e.g., round trip time estimation. Finally, the formulation presented here restricts itself to memory-less adversaries but a strategic adversary, utilizing both memory and perhaps detailed knowledge of the estimator's models, is better able to manipulate estimation while avoiding detection. Accommodating this generalized class of adversaries requires model enhancements that project our problem into the realm of a two-player multi-stage stochastic game with imperfect state information[18,20]. Solutions to such problems are known to be highly sensitive to assumptions on what each player's state-of-knowledge about the other is[17,21]. These generalizations have yet to be explored for network monitoring applications.

ACKNOWLEDGMENT

This material is based upon work under subcontract #069217 issued by BAE Systems National Security Solutions, Inc. and supported by the Defense Advanced Research Projects Agency (DARPA) and the Space and Naval Warfare System Center (SPAWARSYSCEN), San Diego under Contract No. N66001-08-C-2013. Distribution Statement A: Approved for Public Release, Distribution Unlimited.

REFERENCES

[1] G. Holland, N. Vaidya, and P. Bahl, "A rate-adaptive MAC protocol for multi-hop wireless networks," in Proc. of the 7th ***Annual Intl. Conf. on Mobile Comput. and Netw.*** (MobiCom '01). ACM, 2001, pp. 236–251.

[2] J. Zhang and I. Marsic, "Link quality and signal-to-noise ratio in 802.11 WLAN with fading: A time-series analysis," in ***IEEE 64th Veh. Technol. Conf. (VTC-2006 Fall)***, vol. 5, Sep. 2006, pp. 2176–2181.

[3] A. Tønnesen. (2004, Dec.) OLSRD link quality extensions. [Online]. Available: http://www.olsr.org/docs/README-Link- Quality.html

[4] D. S. J. De Couto, D. Aguayo, J. Bicket, and R. Morris, "A high- throughput path metric for multi-hop wireless routing," in ***Proc. of the 9th Annual Intl. Conf. on Mobile Comput. and Netw. (MobiCom'03)***. ACM, 2003, pp. 134–146.

[5] R. E. Kalman, "A new approach to linear filtering and prediction problems," ***ASME J. Basic Eng.,*** vol. 81, no. 1, pp. 35–45, 1960.

[6] Y. Ho and R. Lee, "A Bayesian approach to problems in stochastic estimation and control," ***IEEE Trans. Autom. Control,*** vol. 9, no. 4, pp. 333–339, Oct. 1964.

[7] T. M. Mitchell, ***Machine Learning.*** McGraw-Hill, 1997.

[8] A. Gelb, Ed., ***Applied Optimal Estimation.*** Cambridge, MA: TheMIT Press, 1974.

[9] P. J. Huber, ***Robust Statistics.*** Hoboken, NJ: John Wiley & Sons, 1981.

[10] T. Clausen and P. Jocquet, "Optimized link state routing protocol (OLSR)," ***IETF,*** Oct. 2003, RFC 3626.

[11] C. E. Perkins and P. Bhagwat, "Highly dynamic destination- sequenced distance-vector routing (DSDV) for mobile computers," ***SIGCOMM Comput. Commun. Rev.,*** vol. 24, no. 4, pp. 234–244,1994.

[12] S. Chachulski, M. Jennings, S. Katti, and D. Katabi, "Trading structure for randomness in wireless opportunistic routing," ***SIG-COMM Comput. Commun. Rev.,*** vol. 37, no. 4, pp. 169–180, 2007.

[13] H. Deng, W. Li, and D. P. Agrawal, "Routing security in wireless ad hoc networks," ***IEEE Commun. Mag.,*** vol. 40, no. 10, pp. 70–75, Oct. 2002.

[14] J. Dong, R. Curtmola, and C. Nita-Rotaru, "On the pitfalls of high-throughput multicast metrics in adversarial wireless mesh networks," in ***5th Annual IEEE Comm. Soc. Conf. on Sensor, Mesh and Ad Hoc Commun. and Netw. (SECON '08),*** Jun. 2008, pp. 224–232.

[15] S. J. Lee, W. Su, and M. Gerla, "On-demand multicast routing protocol in multihop wireless mobile networks," ***Mobile Netw. Appl.,*** vol. 7, no. 6, pp. 441–453, 2002.

[16] S. Roy, D. Koutsonikolas, S. Das, and Y. Hu, "High-throughput multicast routing metrics in wireless mesh networks," in ***26th IEEE Intl. Conf. on Distrib. Comput. Syst. (ICDCS 2006)***, Jul. 2006.

[17] H. S. Witsenhausen, "Separation of estimation and control for discrete time systems," ***Proc. IEEE,*** vol. 59, no. 11, pp. 1557–1566, Nov. 1971.

[18] D. P. Bertsekas, ***Dynamic Programming and Optimal Control.*** Belmont, MA: Athena Scientific, 1995.

[19] M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking," ***IEEE Trans. Signal Process.,*** vol. 50, no. 2, pp. 174–188, Feb. 2002.

[20] S. D. Patek and D. P. Bertsekas, "Stochastic shortest path games," ***SIAM J. Control and Optimization,*** vol. 37, no. 3, pp. 804–824, 1999.

[21] Y.-C. Ho, "Team decision theory and information structures," ***Proc. IEEE***, vol. 68, no. 6, pp. 644–654, Jun. 1980.