This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/LCSYS.2020.3048098, IEEE Control Systems Letters

Reputation-based Method to Deal with Bad Sensor Data

Daniel Silvestre

***Abstract*— The participation of citizens through mobile applications in detecting fires or other events, as well as scenarios where there exists a large number of sensors with different noise characteristics raises the question of which data points to accept for the estimation task. If the underlying state dynamics and noise statistics are known, there are various filter-based approaches in the literature, with the well-known example of the Kalman Filter. In this paper, we tackle the problem of selecting which points should be considered to estimate the state of a system with both sensor characteristics and dynamics unknown. By exploiting the techniques from resilient consensus, we first build the intuition that the choice must follow some scoring function. Thus, resorting to rating and reputation systems, we propose an algorithm that assigns scores to the measurements and maintains a pool of the points considered to have better quality. We prove that the rating procedure returns mean scores that are better for sensors with smaller variance and show through simulations the reduced mean error of the estimator in comparison with the state-of-the-art alternatives.**

***Index Terms*— Fault-tolerant systems; Estimation; Fault accommodation; Rating and Reputation Systems**

I. Introduction

WHENEVER a process is measured through various sensors with different levels of noise, a key question is how to decide which measurements should be kept. Moreover, this problem is worsened whenever there is faulty equipment or in situations prone to errors such as compasses near metals, GPS signals near large bodies of water, among others. Modeling the process as a dynamical systems has been attracting the attention of researchers from the control community, given its powerful tools of analysis and design. More so, with the recent interest of both academia and the industry in Cyber-physical Systems (CPSs) and Networked Control Systems (NCSs) and its applicability going beyond the digital world, it becomes of paramount importance to focus on the resilience to both faults and bad sensor data. Many applications share the common dynamical model for distributed algorithms being it: consensus [1], optimization [2]; motion coordination tasks such as flocking and leader following [3]; rendezvous problems [4]; computer networks resource allocation; computation of the relative importance of web pages using the PageRank algorithm [5]; clock synchronization; desynchronization at the Medium Access Control layer [6]; maintaining formations; just to name a few. In this paper, we tackled the scenario where a group of people contribute their location and the azimuth to a sighted forest fire through the FireLoc mobile application. However, GPS and compass measurements are subject to errors caused by multi-path, proximity to large metallic structure among others.

D. Silvestre is the Institute for Systems and Robotics (ISR/IST), University of Lisbon, Portugal and with COPELABS, Lusófona University, Lisbon Portugal, dsilvestre@isr.ist.utl.pt

This work was partially supported by the Portuguese Fundação para a Ciência e a Tecnologia (FCT) through Institute for Systems and Robotics (ISR), under Laboratory for Robotics and Engineering Systems (LARSyS) project UIDB/50009/2020, through project PCIF/MPG/0156/2019 FirePuma and through COPELABS, University Lusófona project UIDB/04111/2020.

When the dynamical model of the system is known, the research on fault detection and outlier removal can be divided into two categories: i) use of filters or observers to detect the attacked or faulty nodes through the generation of residuals or ii) performing model falsification by computing the reachable set of the dynamics. Topic i) typically can encompass the use of a) stochastic filters for the detection such as with a χ^2 test for Kalman Filters [7], unknown input observers [8], sliding mode observers [9], computing the variance of the state [10], etc. On the other hand, ii) employs a set-membership for a worst-case detection as the use of SVOs with polytopes in [11] or its version taking into account the stochastic information of some signals [12], using constrained zonotopes [13] or zonotopes [14], among others. However, modeling a forest fire behavior is a research area on its own and the uncertainties in the model would cause the aforementioned methods to perform poorly.

Another avenue that has been employed is the use of distance-metrics between the received values to discard erroneous data. In the realm of Machine Learning, the task can be accomplished using algorithms such as the k-means [15] or other unsupervised methods [16]. In the community of control, distributed algorithms typically resort to discarding f extreme values such as in [17], or by defining a scoring mechanism [18]. These approaches are related to the proposed solution in the sense that no model information is required.

The proposed solution is inspired by the concept of rating of data and reputation of the contributor in the design of our algorithm. An entity reputation is the consequence of a series of evaluations (ratings) attributed to it given a set of criteria. Even before the advent of mobile applications and websites, we implicitly used reputation whenever dealing with business and people by comparing their behavior against an expectation and by incorporating other important (reputable) entities' opinion. Thus, reputation has been applied in ranking systems [19], [20] and shown to mitigate the effect of a malicious entity

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/LCSYS.2020.3048098, IEEE Control Systems Letters

bribing voters. For additional material on reputation systems, we refer the interested reader to the surveys in [21], [22] and references therein.

The main contributions of this paper are:

- We propose a rating-based algorithm that employs the square of the ℓ_2-norm as a metric to rate bad information and a sliding window to cope with the process change;
- Based on the proposed rating mechanism, a reputation for each agent is introduced that improves the accuracy over the rating algorithm;
- Theoretical results are provided for the rating mechanism and the convergence of reputations.

Even though the main application of this paper is the detection of forest fires from all inputted data, we remark that the proposed mechanisms are useful whenever the process is hard to model as a dynamical equation or in case there are a lot of unknown parameters that would degrade the filter performance in the fault detection procedure.

The remainder of the paper is organized as follows. We formalize the estimation problem in Section II and present state-of-the-art approaches from the resilient consensus literature in Section III. Section IV presents the proposed rating and reputation algorithms with theoretical results being given in Section V. Simulations are presented in Section VI and conclusions and directions of future research being offered in Section VII.

Notation : We let 0_n denote the n-dimensional vector of zeros. The Euclidean norm for vector x is represented as $\|x\|_2 := \sqrt{x^\intercal x}$ and the cardinality of set $\mathcal{S}$ uses the notation $|\mathcal{S}|$. The expected value and the variance of a random variable X are denoted by $\mathbb{E}[X]$ and $\mathrm{Var}(X)$, respectively.

II. Problem Statement

In this paper, we tackle the problem of selecting the more accurate measurements provided by a variety of sensors or human contributors with respect to the underlying state of an unknown dynamical system. Formally, we assume that there is a continuous-time differential state equation:

$$\dot{x} = g(x, u) \tag{1}$$

such that $x \in \mathbb{R}^n$, $u \in \mathbb{R}^p$ and function g is unknown. Within our selected application of forest fire detection, x models the location and other auxiliary variables to describe this process and u stands for external inputs such as wind, available fuel, heat, terrain geography, etc. Depending on the application, the possible approach of estimating g via system identification might not be possible. Be it due to the complex dynamics associated with this process or because it is an initial alert, there is a large uncertainty regarding the state value x and therefore of the external inputs u.

Moreover, we assume that there is a collection of sensors or human actors providing measurements of a portion of the system state. This is modeled by the equation:

$$y(k) = C(k)x(k) + \eta(k) \tag{2}$$

where, for simplicity of notation, matrix $C(k) \in \mathbb{R}^{md\times n}$ models all possible measurements m of size d regarding the state x. The noise signal $\eta(k) \in \mathbb{R}^{md}$ is unknown and acts on all measurements for which no prior stochastic information or any bounds are known. Given that not all measurements will be available in each time instant k, some of the rows in $C(k)$ and in $\eta(k)$ can be set to zero (or any other value that corresponds to the null measurement). We remark that even though the state equation in (1) is written in continuous-time, measurements are received in discrete-time slots. To further simplify, after removing the null measurements, we define $\mathcal{M}(k)$ as the set of all received data at time k.

A summary of the problem addressed in this paper is given in Problem 1.

Problem 1: Given the available measurement set $\mathcal{M}(k)$ at time instant k, find an estimate $\tilde{x}(k)$ for the state $x(k)$ without prior knowledge of the function g and noise associated with each sensor/contributor, i.e., unknown statistics for $\eta(k)$.

III. Outlier Removal Techniques

The problem outlined in Section II and summarized in Problem 1 can be seen as the task of removing outliers from the set of measurements or the need to average the noise of each batch of data points. Before introducing the proposed algorithm, we first detail possible solutions based on this concept and show in simulation that their performance is subpar due to the dynamical nature of the underlying process. This analysis will serve as a motivation and a rationale to understand the current proposal. For the comparison, we consider a case of $m = 10$ sensors that with a 0.5 probability will gather a noisy measurement and send it to the estimator. The process is moving in a straight line but this information is not made available. Noises for the measured positions are drawn uniformly in a square $[-a, a]$ with the values of a being $\{10^{-2}, 0.1733, 0.3367, 0.5, 45, 56, 67, 78, 89, 100\}$, simulating 4 nodes with small errors whereas the remaining almost work as faulty equipment.

A. Average estimator

A first option to get $\tilde{x}(k)$ could be the idea of taking averages of the measurements in each time instant to reduce the effect of the noise. Naturally, this solution is ill-behaved when the average of the expected values of the distributions for each noise signal is nonzero. Nevertheless, lets us define this mechanism and simulate it to assess whether posing the assumption of zero mean noise is sufficient for accurate estimates. We remark that the average estimator corresponds to the least squares solution of the estimation problem for a given set of measurements. The average method defines the estimate as:

$$\tilde{x}(k) = \frac{1}{|\mathcal{M}(k)|} \sum_{z_i \in \mathcal{M}(k)} z_i$$

The mean error from the initial time up to the last iteration is 13.61. This approach presents poor performance as it cannot take advantage of the contributors with small noise values to eliminate the bad data points. In the next section, we present a method in the literature that improves by removing extreme points. We remark to the reader that statistical methods such

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/LCSYS.2020.3048098, IEEE Control Systems Letters

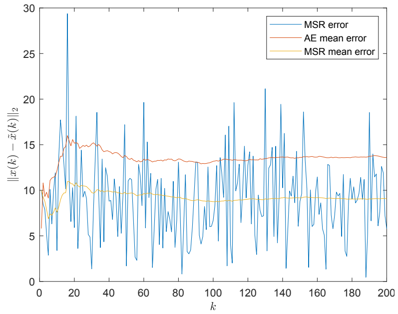

Fig. 1. Error in each iteration and mean error of the Mean Subsequence Reduce (MSR) estimator against the mean error of the Average Estimator (AE).

as the ones using the quantiles distance or a number of standard deviations from the average are hard to generalize to the dynamical case as measurements are taken at different time instants. Using them by considering the data set to be the collections of points gathered into a single instant could suffer from the low dimension expected for the number of measurements.

B. Mean Subsequence Reduce Estimator

In the literature, whenever designing a resilient consensus system, it is often used the concept of mean subsequence that aims at removing the extreme points from the ones received by all the neighbors [17]. Given that the state belongs to $\mathbb{R}^2$ in the running example, we discard the points with the largest and smallest value for each of the coordinates. If the function returning the accepted points after discarding f values in each coordinate is denoted by $\mathrm{MSR}(\mathcal{M}(k), f)$, then the estimate is given by:

$$\tilde{x}(k) = \frac{1}{|\mathrm{MSR}(\mathcal{M}(k), f)|} \sum_{z_i \in \mathrm{MSR}(\mathcal{M}(k), f)} z_i$$

As observed in Figure 1, the MSR method outperforms the AE given that it removes the nodes with extreme coordinates as to avoid the influence of large noise values. The mean error for MSR over the entire simulation was 9.08 and larger values of f improve the quality of this method. However, given the dynamical nature of the underlying state, we could be removing data points relevant to the estimation in a blind manner. Therefore, the proposed solution lies on eliminating data points using a provided metric based on ratings in reputation systems.

IV. Proposed Solution

In this section, we draw inspiration from rating/reputation in ranking systems to provide suggestions and propose a rating function (applied to the data points) and reputation (scores for the agents/sensors) in order to decide which data points and contributors are relevant. Our method also uses a discarding parameter f similar to MSR. Increasing f allows to discard more data points and reduce the possible effect of bad data at the expenses of less capability to average out the noise. The design of f corresponds to the designer choice of the trade-off between removal of bad data points and losing the ability to reduce the error by the averaging operator. In the proposed rating and reputation method an optimal design would force the assumption of knowing either the covariances of the noises or the number of bad sensors

A. Rating System

The first step in our proposal is to design a rating system for the acquired measurements $\mathcal{M}(k)$ based on previous and current data points. Therefore, we assumed that there is a sliding window $\mathcal{H}$ of size h storing past values. When there exists a cluster of data points, these are more significant than isolated measurements and should have a better score. As a consequence, we resort to the Euclidean distance to define the rating function:

$$\mathrm{rating}(p, \mathcal{H}) = \sum_{v_i \in \mathcal{H}} \|p - v_i\|_2^2 \tag{3}$$

thus, equating a small rating to a desirable point p. The algorithm main step is rating all new data points and discard values that have the largest score. However, in order to avoid having all the new measurements being discarded because the dynamics changed the state to a value very different from past measurements, we included the $\mathrm{update}(\cdot)$ function that removes an older value from the sliding window before adding the new measurements. In doing so, it is guaranteed that at least one new measurement is added to the set. The algorithm is summarized in Algorithm 1.

Algorithm 1 Rating-based estimator.

```
/* Initialize sliding window */
H(0) = ∅
for each k > 0 do
    /* Receive data points */
    Receive M(k)
    /* Update sliding window */
    H(k) = update(H(k − 1), M(k))
    /* Rate all points */
    for each z_i ∈ H(k) do
        r_i = rating(z_i, H(k))
    end for
    /* Discard f points */
    for each j ≤ f do
        H(k).discard(arg max_i r_i)
    end for
    /* Compute estimate */
    x̃(k) = mean(z_i ∈ M(k) ∩ H(k))
end for
```

The rating-based algorithm ignores the fact that "consistently" good sensors data points should be taken in higher consideration in the rating methodology than those of agents with poor performance. In the next section, we present an algorithm that can take advantage of the quality of the sender and not only on the received data.

B. Reputation System

The concept of reputation translates the quality of the provided information and, in turn, influences how measurements are rated. Let us introduce the reputation of a sensor as:

$$w_j = 1 - \frac{d_j}{s_j}, \forall j \leq m$$

where w_j is the weight (reputation) associated with sensor j using the number of discarded d_j messages and the total s_j messages of sensor j over all time instants. We remark that the removed data points at the beginning of each time instant due to the update function do not count towards the discarded messages (variable d_j only counts points removed due to a bad rating). Having defined the reputation of a node, the rating function in (3) can be generalized for asymmetric reputations in the following manner:

$$\text{rating}(p, \mathcal{H}, w) = \sum_{v_{ij} \in \mathcal{H}} w_j \|p - v_{ij}\|_2^2$$

where we made clear that the messages $v_{ij} \in \mathcal{H}$ are the message with id i of node j in the sliding window. The general algorithm is similar to that in Algorithm 1 with the exception that prior to the calculation of the rating, the reputation has to be updated, which is summarized in Algorithm 2.

Algorithm 2 Reputation-based estimator.

/* *Initialize sliding window and reputation vector* */
$\mathcal{H} = \emptyset$
$r = 0_m$
for each $k > 0$ **do**
  /* *Receive data points* */
  Receive $\mathcal{M}(k)$
  /* *Increment total data points per sensor* */
  $s(k) = \text{increment}(s(k-1), \mathcal{M}(k))$
  /* *Update sliding window* */
  $\mathcal{H}(k) = \text{update}(\mathcal{H}(k-1), \mathcal{M}(k))$
  /* *Discard points until* $\mathcal{H}(k)$ *is of size* h */
  while $|\mathcal{H}(k)| > h$ **do**
    /* *Compute reputations* */
    for each $j \leq m$ **do**
      $w_j(k) = 1 - \frac{d_j(k)}{s_j(k)}$
    end for
    /* *Rate all points using the reputations* */
    for each $z_i \in \mathcal{H}(k)$ **do**
      $r_i = \text{rating}(z_i, \mathcal{H}(k), w(k))$
    end for
    $\mathcal{H}(k).\text{discard}(\arg\max_i r_i)$
    /* *Update number of discarded points* */
    $d_j = d_j + 1$, for discarded z_{ij}
  end while
  /* *Compute estimate* */
  $\tilde{x}(k) = \text{mean}(z_i \in \mathcal{M}(k) \cap \mathcal{H}(k))$
end for

V. Theoretical Results

In this section, we evaluate the theoretical guarantees of the proposed rating and reputation mechanisms. The first result states that noisy measurements with larger variance will receive a higher expected rating, i.e., on average they will be removed more times. This is given in Theorem 2.

Theorem 2 (Expected Rating): Assume two measurements $z_1(k)$ and $z_2(k)$ corresponding to the full state corrupted by zero-mean independent noise signals with diagonal covariance matrices Γ_1 and Γ_2, respectively. If $\Gamma_1 < \Gamma_2$ holds element-wise, then $\mathbb{E}[r_1(k)] < \mathbb{E}[r_2(k)]$.

Proof: Given that the noise is independent for all entries of the measurements, we can prove the conclusion for a scalar measurement and the result follows for the vector case by applying to each entry the same reasoning. From (2), we have:

$$\begin{aligned} z_1(k) &= x(k) + \eta_1(k) \\ z_2(k) &= x(k) + \eta_2(k) \end{aligned}$$

with $\eta_1(k)$ and $\eta_2(k)$ satisfying $\text{Var}(\eta_1(k)) < \text{Var}(\eta_2(k))$. To lighten the notation, we will omit the dependence of all variables on the time variable k. One can then compute the expected value for r_1 using its definition (3):

$$\begin{aligned} \mathbb{E}[r_1] &= \mathbb{E}\left[\sum_{v_i \in \mathcal{H}} (z_1 - v_i)^2\right] \qquad (4) \\ &= \mathbb{E}\left[\sum_{v_i \in \mathcal{H}} (x + \eta_1 - x - \eta_i)^2\right] \\ &= \mathbb{E}\left[\sum_{v_i \in \mathcal{H}} (\eta_1 - \eta_i)^2\right] \\ &= \sum_{v_i \in \mathcal{H}} \mathbb{E}\left[(\eta_1 - \eta_i)^2\right] \end{aligned}$$

where the last equality is due to the linearity of the expected value operator. Given that η_1 and all η_i are independent random variables, the variance of the subtraction satisfies:

$$\begin{aligned} \text{Var}(\eta_1 - \eta_i) &= \text{Var}(\eta_1) + \text{Var}(\eta_i) \\ &= \mathbb{E}\left[\eta_1^2\right] + \mathbb{E}\left[\eta_i^2\right] \end{aligned}$$

and on the other hand, given its definition:

$$\text{Var}(\eta_1 - \eta_i) = \mathbb{E}\left[(\eta_1 - \eta_i)^2\right].$$

Therefore, the expected value of r_1 in (4) can be further simplified to:

$$\begin{aligned} \mathbb{E}[r_1] &= \sum_{v_i \in \mathcal{H}} \mathbb{E}\left[\eta_1^2\right] + \mathbb{E}\left[\eta_i^2\right] \\ &= h\mathbb{E}\left[\eta_1^2\right] + \sum_{v_i \in \mathcal{H}} \mathbb{E}\left[\eta_i^2\right]. \end{aligned}$$

We can now compute:

$$\begin{aligned} \mathbb{E}[r_1] - \mathbb{E}[r_2] &= h\mathbb{E}\left[\eta_1^2\right] + \sum_{v_i \in \mathcal{H}} \mathbb{E}\left[\eta_i^2\right] - h\mathbb{E}\left[\eta_2^2\right] - \sum_{v_i \in \mathcal{H}} \mathbb{E}\left[\eta_i^2\right] \\ &= h\mathbb{E}\left[\eta_1^2\right] - h\mathbb{E}\left[\eta_2^2\right] \\ &< 0 \end{aligned}$$

and the conclusion follows. ■

The result in Theorem 2 demonstrates that our proposed rating function rates on average consistently with the added noise. In the next theorem, we provide a similar analysis for

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/LCSYS.2020.3048098, IEEE Control Systems Letters

the reputation system with an additional assumption that the probability distribution of the noise causes the rating of a node to be higher than the other with a probability of at least 0.5. In the following, we will denote by $P[\cdot]$ the probability operator whereas $\mathbb{E}[a|b]$ stands for the conditional expected value of a knowing b.

Theorem 3 (Expected Reputation): Assume two sensors sending the same number of messages and providing measurements $z_1(k)$ and $z_2(k)$ corresponding to the full state corrupted by zero-mean independent noise signals. If $w_1(k-1) \geq w_2(k-1)$ and $P[r_1(k) < r_2(k)] > 0.5$, then $\mathbb{E}[w_1(k)|w(k-1)] > \mathbb{E}[w_2(k)|w(k-1)]$.

Proof: Let us denote $p = 1 - P[r_1(k) < r_2(k)] < 0.5$. Since both nodes send the same number of messages, $s = s_1 = s_2$. One can compute the expected value of the next reputation value conditional to the previous one for sensor 1 as:

$$\begin{aligned}\mathbb{E}[w_1(k)|w(k-1)] &= p\left(1 - \frac{d_1+1}{s+1}\right) + (1-p)\left(1 - \frac{d_1}{s+1}\right)\\ &= 1 - \frac{d_1+p}{s+1}\end{aligned}$$

and similarly for sensor 2:

$$\begin{aligned}\mathbb{E}[w_2(k)|w(k-1)] &= (1-p)\left(1 - \frac{d_1+1}{s+1}\right) + p\left(1 - \frac{d_1}{s+1}\right)\\ &= 1 - \frac{d_2+(1-p)}{s+1}.\end{aligned}$$

Combining both equations we have:

$$\begin{aligned}\mathbb{E}[w_1(k) - w_2(k)|w(k-1)] &= 1 - \frac{d_1+p}{s+1} - \left(1 - \frac{d_2+(1-p)}{s+1}\right)\\ &= \frac{1-2p+d_2-d_1}{s+1}\\ &> 0.\end{aligned}$$

The last inequality comes from the facts that i) $p < 0.5$ implies $1-2p > 0$ and ii) from the statement of the theorem, $w_1(k-1) \geq w_2(k-1)$ implies that $d_1 < d_2$. Thus, the conclusion follows. ■

Please note that the assumption of Theorem 3 regarding the probability density function is satisfied for noise signals with similar distributions. For example, if both noise values follow a Gaussian distribution with covariance matrices as in Theorem 2, the probability of the rating of z_1 being smaller than z_2 is at least 0.5. However, this condition fails for example when sensors have very small noise levels but at some time instants they will introduce very large values. Such cases, even though they can satisfy conditions in Theorem 2 will fail the ones in Theorem 3.

VI. Simulations

In this section, we present simulation results in comparison with the state-of-the-art in resilient mechanisms for linear iterative algorithms. We first recover the example of a straight line to illustrate the performance of the proposed methods with respect to the MSR algorithm. The scenario consists of $m = 10$ sensors that with some probability will gather a noisy measurement and send it to the estimator. Noises are again uniformly selected with parameters $\{10^{-2}, 0.1733, 0.3367, 0.5, 45, 56, 67, 78, 89, 100\}$.

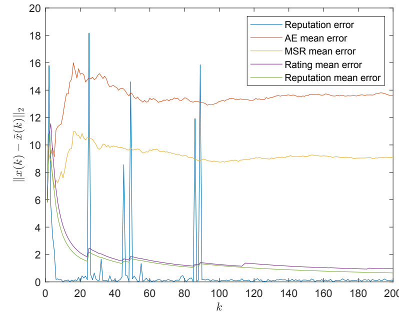

Fig. 2. Error in each iteration and mean error of the Reputation algorithm in Algorithm 2 in comparison with Mean Subsequence Reduce (MSR) estimator, Average Estimator (AE), and rating algorithm mean errors for $p = 0.5$.

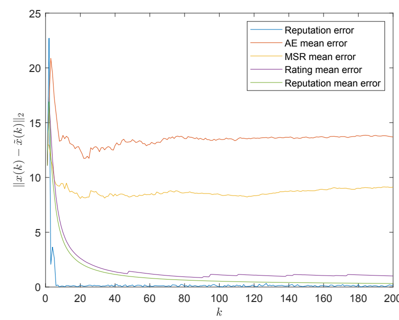

Fig. 3. Error in each iteration and mean error of the Reputation algorithm in Algorithm 2 in comparison with Mean Subsequence Reduce (MSR) estimator, Average Estimator (AE), and rating algorithm mean errors for $p = 1$.

When using the reputation-based method in Algorithm 2, the estimation is improved as reported in Figure 2 where the final mean error of both strategies approach 0.99. For this simulation, once the reputations converge, nodes providing bad data points are disregarded given that the rating of their measurements will be worse than those with higher scores.

Intuitively, a rating/reputation mechanism should work better if given more data. The second simulation uses the same parameters except all nodes send their information in all time instants. Figure 3 presents the results for the reputation algorithm with the clear distinction that it is never affected by the erroneous sensors. The rating mechanism presents a worse mean error given that points in the sliding window might be better rated given that the dynamical nature of the state can cause some bad points to be closer to the rest of the window. As a consequence, the rating mechanism in Algorithm 1 may add newer noisier data points while removing older (and better) measurements, which degrades the estimation performance. By keeping track of the reputation of each sensor, this effect is removed as it can be observed in Figure 3. The algorithm will assign a worse reputation and prevent bad points from being added to the window at the end of the iteration cycle.

The previous simulations were all based on the underlying state moving in a straight line, which will not be similar to

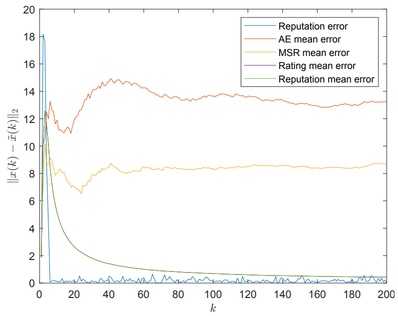

Fig. 4. Error in each iteration and mean error of the Reputation algorithm in Algorithm 2 in comparison with Mean Subsequence Reduce (MSR) estimator, Average Estimator (AE), and rating algorithm mean errors for $p = 1$ in a random walk.

real data. The second simulation defined a dynamical system composed of an integrator with stochastic input (generating a random walk trajectory). Figure 4 shows the mean errors and the iteration error for the reputation algorithm. The results are very similar to the ones in Figure 3 given that it is the noise levels and available data that has the most influence on the reputation and rating.

VII. Conclusions and Future Work

In this paper, we tackle the problem of deciding which data points from a group of sensors should be used in the estimation task. Given that the underlying dynamical system is unknown and possibly complex, it precludes the use of system identification techniques to resort to resilient state estimation filter available in the literature. Moreover, outlier removal based on the quartile range does not cope well with a small number of data points and is not appropriate for time-varying variables. On the other hand, the current state-of-the-art in resilient consensus presents large estimation errors. The proposed solution is based on a rating function for the data points and a sliding window given the dynamical nature of the underlying state. Additionally, by keeping track of the reputation of the sensors, it is possible to achieve better estimations. The expected values of the ratings are computed and shown to map the variances of the noise signals.

In simulation, it is shown that the proposed methods a better performance in comparison with the state-of-the-art. However, there are two main avenues of research to pursue. The first one is validating the methodology using real data that is expected to occur in the next two years. The second one is on how to deal with users that intentionally try to undermine the objective of the application. Two strategies can be investigated: i) validate reputation with verified data from the authorities; and, ii) redefine the reputation system to have a reset operation whenever the ratings of the data points without reputation present a conflict. Strategy i) is highly dependent on the availability of verified data whereas option ii) could work for general application by having the algorithm self check its detection. Another possibility is to investigate different weighting mechanisms for the reputation of the contributors.

References

[1] D. Silvestre, J. P. Hespanha, and C. Silvestre, "Broadcast and gossip stochastic average consensus algorithms in directed topologies," *IEEE Transactions on Control of Network Systems*, vol. 6, no. 2, pp. 474–486, June 2019.

[2] D. Silvestre, "Optool—an optimization toolbox for iterative algorithms," *SoftwareX*, vol. 11, p. 100371, 2020.

[3] R. Ribeiro, D. Silvestre, and C. Silvestre, "Decentralized control for multi-agent missions based on flocking rules," in *CONTROLO 2020*, J. A. Gonçalves, M. Braz-César, and J. P. Coelho, Eds. Cham: Springer International Publishing, 2021, pp. 445–454.

[4] R. Ribeiro, D. Silvestre, and C. Silvestre, "A rendezvous algorithm for multi-agent systems in disconnected network topologies," in *2020 28th Mediterranean Conference on Control and Automation (MED)*, Sep. 2020, pp. 592–597.

[5] D. Silvestre, J. Hespanha, and C. Silvestre, "A pagerank algorithm based on asynchronous gauss-seidel iterations," in *American Control Conference (ACC)*, June 2018, pp. 484–489.

[6] D. Silvestre, J. Hespanha, and C. Silvestre, "Desynchronization for decentralized medium access control based on gauss-seidel iterations," in *American Control Conference (ACC)*, July 2019, pp. 4049–4054.

[7] K. Manandhar, X. Cao, F. Hu, and Y. Liu, "Detection of faults and attacks including false data injection attack in smart grid using kalman filter," *IEEE Transactions on Control of Network Systems*, vol. 1, no. 4, pp. 370–379, Dec 2014.

[8] A. Cristofaro and T. Johansen, "Fault tolerant control allocation using unknown input observers," *Automatica*, vol. 50, no. 7, pp. 1891 – 1897, 2014.

[9] P. Menon and C. Edwards, "Robust fault estimation using relative information in linear multi-agent networks," *IEEE Transactions on Automatic Control*, vol. 59, no. 2, pp. 477–482, Feb 2014.

[10] D. Silvestre, J. P. Hespanha, and C. Silvestre, "Resilient desynchronization for decentralized medium access control," *IEEE Control Systems Letters*, vol. 5, no. 3, pp. 803–808, July 2021.

[11] D. Silvestre, P. Rosa, J. P. Hespanha, and C. Silvestre, "Fault detection for LPV systems using set-valued observers: A coprime factorization approach," *Systems & Control Letters*, vol. 106, pp. 32 – 39, 2017.

[12] ——, "Stochastic and deterministic fault detection for randomized gossip algorithms," *Automatica*, vol. 78, pp. 46 – 60, 2017.

[13] J. K. Scott, D. M. Raimondo, G. R. Marseglia, and R. D. Braatz, "Constrained zonotopes: A new tool for set-based estimation and fault detection," *Automatica*, vol. 69, pp. 126 – 136, 2016.

[14] J. K. Scott, R. Findeisen, R. D. Braatz, and D. M. Raimondo, "Input design for guaranteed fault diagnosis using zonotopes," *Automatica*, vol. 50, no. 6, pp. 1580 – 1589, 2014.

[15] G. Gan and M. K.-P. Ng, "k-means clustering with outlier removal," *Pattern Recognition Letters*, vol. 90, pp. 8 – 14, 2017.

[16] W. Liu, G. Hua, and J. R. Smith, "Unsupervised one-class learning for automatic outlier removal," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2014.

[17] S. M. Dibaji and H. Ishii, "Consensus of second-order multi-agent systems in the presence of locally bounded faults," *Systems & Control Letters*, vol. 79, pp. 23 – 29, 2015.

[18] G. Ramos, D. Silvestre, and C. Silvestre, "A general discrete-time method to achieve resilience in consensus algorithms," in *to appear in 59th IEEE Conference on Decision and Control (CDC)*, December 2020.

[19] R.-H. Li, J. Xu Yu, X. Huang, and H. Cheng, "Robust reputation-based ranking on bipartite rating networks," in *Proceedings of the SIAM international conference on data mining*. SIAM, 2012, pp. 612–623.

[20] G. Ramos and L. Boratto, "Reputation (in)dependence in ranking systems: Demographics influence over output disparities," in *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, ser. SIGIR '20, 2020, p. 2061–2064.

[21] A. Jøsang, R. Ismail, and C. Boyd, "A survey of trust and reputation systems for online service provision," *Decision support systems*, vol. 43, no. 2, pp. 618–644, 2007.

[22] F. Hendrikx, K. Bubendorfer, and R. Chard, "Reputation systems: A survey and taxonomy," *Journal of Parallel and Distributed Computing*, vol. 75, pp. 184–197, 2015.