A Message Passing Approach for Decision Fusion of Hidden-Markov Observations in the Presence of Synchronized Attacks

Andrea Abrardo*†, Mauro Barni*, Kassem Kallas*, Benedetta Tondi*
*Department of Information Engineering and Mathematics, University of Siena
Via Roma 56, 53100 - Siena, ITALY
†CNIT - Consorzio Nazionale Interuniversitario per le Telecomunicazioni
abrardo@diism.unisi.it, barni@dii.unisi.it, k_kallas@hotmail.com, benedettatondi@gmail.com

Abstract—**We consider a setup in which a Fusion Center (FC) makes a binary decision on the sequence of system states by relying on local observations provided by both honest and byzantine nodes, i.e., nodes that deliberately alter the result of the local decision to induce an error at the fusion center. In this setting, we assume a Markovian information model for the status with a given transition probability that can be perfectly estimated at the FC. Hence, we consider an attacking strategy where the byzantine nodes can coordinate their attacks by producing correlated reports, with the aim of mimicking the behavior of the original information and at the same time minimizing the information conveyed to the FC about the sequence of states. In this scenario, we derive a nearly-optimal fusion scheme based on message passing (MP) and factor graphs. Experimental results show that, although the proposed detector is able to mitigate the effect of Byzantines, the coordination of the efforts is very harmful and significantly impairs the detection performance.**

Keywords*–*Decision Fusion in Adversarial Settings; Adversarial Signal Processing; Byzantine attacks; Message Passing Algorithm; Markovian Sources.

I. Introduction

We address a decision problem in which a Fusion Center (FC) is required to make a decision about the status of an observed system by relying on the information provided by the nodes of a sensor network. In the adversarial version of this problems, some of the nodes, commonly referred to as Byzantines, malevolently alter their reports to induce a decision error [1]. This is a recurrent problem in many scenario wherein the nodes may take advantage from a decision error, e.g., in cognitive radio networks [2] or online reputation systems [3]. In this paper, we focus on a binary version of the fusion problem, wherein the system can assume only two states. Specifically, the nodes observe the system over an observation window of m time instants and make a local decisions about the sequence of system states. Honest nodes send their decisions to the FC, while Byzantines try to induce a decision error by flipping their observations with a certain probability. When the FC makes its decision on the system state at a certain time instant j by relying only on the corresponding report, the Bayesian optimal fusion rule for the non-adversarial version of this case has been derived in [4] and it is known as Chair-Varshney. In the presence of Byzantines, Chair-Varshney rule requires the knowledge of Byzantines' positions along with their flipping probability P_{mal}. However, this information is rarely available and then the FC needs to resort to suboptimal fusion strategies. In order to improve the estimation of the system states, the FC can gather a sequence of reports and make a global decision. In this way, it is possible for the FC to perform *isolation* of the Byzantines by identifying the malevolent nodes and discarding their reports [5][6]. Isolation is achieved by counting the mismatches between the reports and the global decision. In [7], a soft isolation scheme is proposed where the reports from suspect byzantine nodes are given a lower importance rather than being discarded.

In [8], the optimum fusion rule under a bunch of observations is first derived assuming to know the malicious probability P_{mal} of the Byzantines along with the probability that a node is Byzantine. Then, the knowledge of P_{mal} at the FC is relaxed as it is strategically chosen in a game-theoretic framework. In this work, the authors show that, differently from what commonly expected, always flipping the local decision is not necessarily the best option for the Byzantines. In fact, in some cases, in order to prevent identification, it is better for the Byzantines to minimize the mutual information between the reports submitted to the FC and the system states. One of the main inconvenience of the optimal fusion rule proposed in [8] is that the computational cost grows exponentially with the size of the observation window. A nearly-optimum fusion scheme based on message passing (MP) and factor graphs is proposed in [9], where an iterative algorithm based on the so called Generalised Distributive Law (GLD, [10]), permits to achieve a linear complexity. Besides, whereas in [8] the analysis is limited to the case of independent system states, in [9] it is extended to the case of sequences with Markovian distribution, which is rather common model in many practical scenarios; for instance, in cognitive radio networks the primary user occupancy of the spectrum is often modelled as a Hidden Markov Model (HMM), e.g., [11][12].

In this paper, by focusing on the case of Markovian system states, we consider the scenario in which the Byzantines can cooperate by synchronizing their efforts to push forth more powerful attacks. Specifically, the contribution of this paper is twofold: we first propose two types of synchronized attacks; then, we refine the detection scheme based on message passing proposed in [9] and devise the nearly-optimal decision rule for the synchronized case. Finally, we demonstrate the effectiveness of the proposed scheme by means of numerical simulations. The results show that, upon knowing the attacking strategy, the new detector can mitigate the effect of the Byzantines. Nevertheless, synchronization among Byzantines is very harmful and significantly impairs the detection performance with respect to the non-synchronized case.

The rest of this paper is organized as follows: in Section II, we formalize the problem at hand and we propose the synchronized attack models, while in Section III the message passing algorithm is proposed. In Section IV we use simulations to analyze the performance of the synchronized Byzantine attacks and the message passing algorithm. The paper is concluded in Section V.

II. Problem Formulation

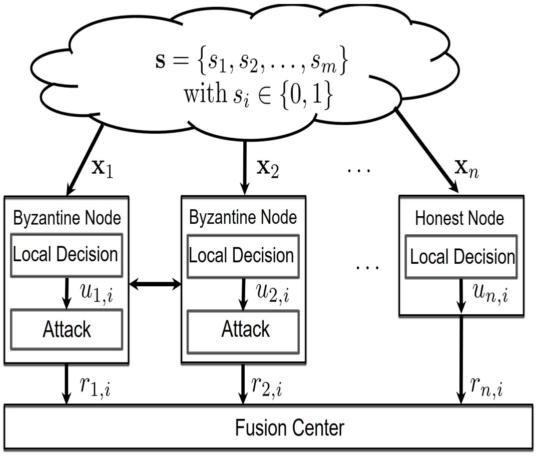

Figure 1. Sketch of the adversarial decision fusion scheme.

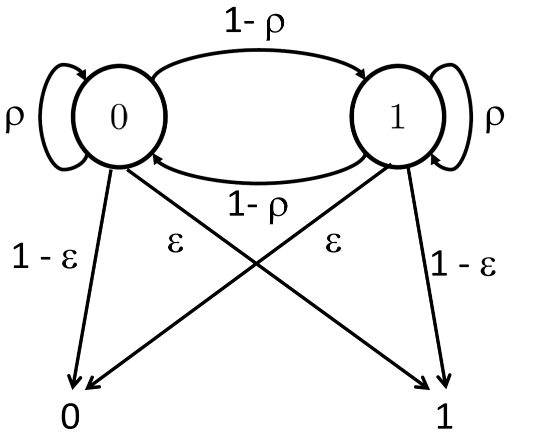

Figure 2. Hidden-Markov model for the local decisions.

1) Problem Setup: The adversarial decision fusion scheme considered in this paper is depicted in Figure 1. We let $\mathbf{s} = \{s_1, s_2, \dots, s_m\}$ with $s_i \in \{0,1\}$ indicate the sequence of system states over an observation window of length m. We assume that the sequence of states $\mathbf{s}$ follows a *Markov model* of order 1, with transition probabilities $p(s_i|s_{i-1}) = 1-\rho$ if $s_i = s_{i-1}$ and $p(s_i|s_{i-1}) = \rho$ when $s_i \neq s_{i-1}$. Then, the probability of a sequence is given by $p(\mathbf{s}) = \prod_i p(s_i|s_{i-1})$, where for $i=1$ we have $p(s_1|s_0) = p(s_1) = 0.5$.

The nodes collect information about the system through the vectors $\mathbf{x}_1, \mathbf{x}_2 \dots \mathbf{x}_n$, with $\mathbf{x}_j$ indicating the observations available at node j. Based on such observations, a node j makes a local decision $u_{i,j}$ about system state s_i. We assume that the local error probability is $p(u_{i,j} \neq s_i) = \varepsilon$, which does not depend on either i or j. Then, the sequence of the local decisions follows a Hidden-Markov distribution [13], as shown in Figure 2. The state of the nodes in the network is given by the vector $\mathbf{h} = \{h_1, h_2, \dots, h_n\}$ with $h_j = 1/0$ indicating that node j is honest or Byzantine, respectively. Finally, the matrix $\mathbf{R} = \{r_{i,j}\}$, $i = 1, \dots, m$, $j = 1, \dots, n$ contains all the reports received by the FC. Specifically, $r_{i,j}$ is the report sent by node j relative to s_i. For honest nodes we have $u_{i,j} = r_{i,j}$ while, for Byzantines, possibly $u_{i,j} \neq r_{i,j}$. Then, by assuming an error-free transmission between nodes and FC, according to the local decision error model, for honest nodes we have:

$$\begin{aligned} p\left(r_{i,j}|s_i, h_j = 1\right) = (1-\varepsilon)\delta(r_{i,j} - s_i) \\ + \varepsilon(1 - \delta(r_{i,j} - s_i)), \end{aligned} \tag{1}$$

where $\delta(a)$ is equal to 1 when its argument is 0 and 0 otherwise. On the other hand, the probability that the FC receives a wrong report from a Byzantine depends on the attack strategy and is discussed in the following section.

2) The Attacks Model: In the general context of synchronized attacks, we consider two different strategies. In the first case, the Byzantines generate a fake states sequence $\hat{\mathbf{s}}$ and decide to flip the reports only when $\hat{s}_i = 0$. The rationale of this attack is to reduce the mutual information conveyed by the Byzantines towards the FC with respect to the classical $P_{mal} = 1$ case, thus reducing the identification probability. The generation of the fake sequence can be achieved for instance by using a pseudo random generator with a common seed to synchronize the local clocks of the sensors.

In the second attack strategy, the Byzantines generate a fake sequence which follows the statistic of the original sequence, namely a Markovian sequence $\hat{\mathbf{s}}$ with transition probability $\hat{\rho}$. Then, they introduce some intentional i.i.d errors with probability ε thus mimicking the behavior of the honest nodes. In this case, the mutual information between the system states and the malicious reports is completely canceled. To elaborate, for the first attack, we have

$$\begin{aligned} & p\left(r_{i,j}|s_i, \hat{s}_i, h_j = 0\right) = \\ & \hat{s}_i[(1-\varepsilon)\delta(r_{i,j} - s_i) + \varepsilon(1 - \delta(r_{i,j} - s_i))] \\ & \quad - (\hat{s}_i - 1)[\varepsilon\delta(r_{i,j} - s_i) + (1-\varepsilon)(1 - \delta(r_{i,j} - s_i))] \end{aligned} \tag{2}$$

where ε is the error probability of the local decisions at the nodes. For the second case, the report conditional probabilities depend on the fake states only:

$$\begin{aligned} & p\left(r_{i,j}|\hat{s}_i, h_j = 0\right) = \\ & \quad (1-\varepsilon)\delta(r_{i,j} - \hat{s}_i) + \varepsilon(1 - \delta(r_{i,j} - \hat{s}_i)), \end{aligned} \tag{3}$$

where this time ε is the probability of the i.i.d. errors introduced intentionally.

Eventually, we consider that nodes' state are independent of each other and the state of each node is a Bernoulli random variable with parameter α, that is $p(h_j = 0) = \alpha, \forall j$. In this way, the number of byzantine nodes in the network is a random variable following a binomial distribution, corresponding to the maximum entropy case [8] with $p(\mathbf{h}) = \prod_j p(h_j)$, where $p(h_j) = \alpha(1-h_j) + (1-\alpha)h_j$.

III. MP-based decision fusion with synchronized Byzantines

Given the sequence of reports, the optimum decision at the FC can be taken by looking at the *bitwise* Maximum A

Posteriori Probability (MAP) estimation of the system states $\{s_i\}$ which reads as follows:

$$\begin{aligned}
s_i^* &= \arg\max_{s_i\in\{0,1\}} \; p\left(s_i|\mathbf{R}\right) \\
&= \arg\max_{s_i\in\{0,1\}} \sum_{\{\mathbf{s},\hat{\mathbf{s}},\mathbf{h}\}\backslash s_i} p\left(\mathbf{s},\hat{\mathbf{s}},\mathbf{h}|\mathbf{R}\right) \\
&= \arg\max_{s_i\in\{0,1\}} \sum_{\{\mathbf{s},\hat{\mathbf{s}},\mathbf{h}\}\backslash s_i} p\left(\mathbf{R}|\mathbf{s},\hat{\mathbf{s}},\mathbf{h}\right) p(\mathbf{s})p(\hat{\mathbf{s}})p(\mathbf{h}) \\
&= \arg\max_{s_i\in\{0,1\}} \sum_{\{\mathbf{s},\mathbf{q},\mathbf{h}\}\backslash s_i} \prod_{i,j} p\left(r_{i,j}|s_i,\hat{s}_i,h_j\right) \prod_i p(s_i|s_{i-1}) \\
&\prod_i p(\hat{s}_i|\hat{s}_{i-1}) \prod_j p(h_j)
\end{aligned} \tag{4}$$

where the notation $\sum_{\backslash}$ denotes a summation over all the variables contained in the expression except the one listed after the operator. For a given $\mathbf{h}$, the matrix of the observations $\mathbf{R}$ at the FC follows a HMM.

The objective function in the optimal fusion rule expressed in (4) can be seen as a marginalization of a sum product of functions of binary variables, and, as such, it falls within the MP framework [14]. Specifically, in our problem, the variables are the system states s_i, the fake system states $\hat{s}_i$, and the status of the nodes h_j, while the functions are the probabilities of the reports $p\left(r_{i,j}|s_i,\hat{s}_i,h_j\right)$, the conditional probabilities $p(s_i|s_{i-1})$, $p(\hat{s}_i|\hat{s}_{i-1})$, and the a-priori probabilities $p(h_j)$. The resulting bipartite graph along with all the messages exchanged are shown in Figure 3. These messages are exchanged to parallely estimate each state s_i in the vector s. Specifically, we have:

$$\begin{aligned}
&\tau_i^{(l)}(s_i) = \varphi_i^{(l)}(s_i) \prod_{j=1}^{n} \nu_{i,j}^{(u)}(s_i) \\
&\qquad i = 1,\ldots,m \\
&\tau_i^{(r)}(s_i) = \varphi_i^{(r)}(s_i) \prod_{j=1}^{n} \nu_{i,j}^{(u)}(s_i) \\
&\qquad i = 1,\ldots,m \\
&\varphi_i^{(l)}(s_i) = \sum_{s_{i+1}=0,1} p\left(s_{i+1}|s_i\right) \tau_{i+1}^{(l)}(s_{i+1}) \\
&\qquad i = 1,\ldots,m-1 \\
&\varphi_i^{(r)}(s_i) = \sum_{s_{i-1}=0,1} p\left(s_i|s_{i-1}\right) \tau_{i-1}^{(r)}(s_{i-1}) \\
&\qquad i = 2,\ldots,m \\
&\varphi_1^{(r)}(s_1) = p(s_1) \\
&\nu_{i,j}^{(u)}(s_i) = \sum_{h_j=0,1} \sum_{\hat{s}_i=0,1} p\left(r_{i,j}\,|s_i,\hat{s}_i,h_j\,\right) \lambda_{j,i}^{(u)}(h_j)\hat{\nu}_{i,j}^{(d)}(\hat{s}_i) \\
&\qquad i = 1,\ldots,m, \quad j = 1,\ldots,n \\
&\nu_{i,j}^{(d)}(s_i) = \varphi_i^{(r)}(s_i)\varphi_i^{(l)}(s_i) \prod_{\substack{k=1\\k\neq j}}^{n} \nu_{i,k}^{(u)}(s_i) \\
&\qquad i = 1,\ldots,m-1, \quad j = 1,\ldots,n \\
&\nu_{m,j}^{(d)}(s_m) = \varphi_i^{(r)}(s_m) \prod_{\substack{k=1\\k\neq j}}^{n} \nu_{m,k}^{(u)}(s_m)
\end{aligned}$$

$$\begin{aligned}
&\qquad j = 1,\ldots,n \\
&\hat{\tau}_i^{(l)}(\hat{s}_i) = \hat{\varphi}_i^{(l)}(\hat{s}_i) \prod_{j=1}^{n} \hat{\nu}_{i,j}^{(u)}(\hat{s}_i) \\
&\qquad i = 1,\ldots,m \\
&\hat{\tau}_i^{(r)}(\hat{s}_i) = \varphi_i^{(r)}(\hat{s}_i) \prod_{j=1}^{n} \nu_{i,j}^{(u)}(\hat{s}_i) \\
&\qquad i = 1,\ldots,m \\
&\hat{\varphi}_i^{(l)}(\hat{s}_i) = \sum_{\hat{s}_{i+1}=0,1} p\left(\hat{s}_{i+1}|\hat{s}_i\right) \hat{\tau}_{i+1}^{(l)}(\hat{s}_{i+1}) \\
&\qquad i = 1,\ldots,m-1 \\
&\hat{\varphi}_i^{(r)}(\hat{s}_i) = \sum_{\hat{s}_{i-1}=0,1} p\left(\hat{s}_i|\hat{s}_{i-1}\right) \hat{\tau}_{i-1}^{(r)}(\hat{s}_{i-1}) \\
&\qquad i = 2,\ldots,m \\
&\hat{\varphi}_1^{(r)}(s_1) = p(\hat{s}_1) \\
&\hat{\nu}_{i,j}^{(u)}(\hat{s}_i) = \sum_{h_j=0,1} \sum_{s_i=0,1} p\left(r_{i,j}\,|s_i,\hat{s}_i,h_j\,\right) \lambda_{j,i}^{(u)}(h_j)\nu_{i,j}^{(d)}(s_i) \\
&\qquad i = 1,\ldots,m, \quad j = 1,\ldots,n \\
&\hat{\nu}_{i,j}^{(d)}(\hat{s}_i) = \hat{\varphi}_i^{(r)}(\hat{s}_i)\hat{\varphi}_i^{(l)}(s_i) \prod_{\substack{k=1\\k\neq j}}^{n} \hat{\nu}_{i,k}^{(u)}(s_i) \\
&\qquad i = 1,\ldots,m-1, \quad j = 1,\ldots,n \\
&\hat{\nu}_{m,j}^{(d)}(\hat{s}_m) = \hat{\varphi}_i^{(r)}(\hat{s}_m) \prod_{\substack{k=1\\k\neq j}}^{n} \hat{\nu}_{m,k}^{(u)}(\hat{s}_m) \\
&\qquad j = 1,\ldots,n \\
&\lambda_{j,i}^{(d)}(h_j) = \sum_{s_i=0,1} \sum_{\hat{s}_i=0,1} p\left(r_{i,j}\,|s_i,\hat{s}_i,h_j\,\right) \nu_{i,j}^{(d)}(s_i)\hat{\nu}_{i,j}^{(d)}(\hat{s}_i) \\
&\qquad i = 1,\ldots,m, \quad j = 1,\ldots,n \\
&\lambda_{j,i}^{(u)}(h_j) = \omega_j^{(u)}(h_j) \prod_{\substack{q=1\\q\neq i}}^{m} \lambda_{j,q}^{(d)}(h_j) \\
&\qquad i = 1,\ldots,m, \quad j = 1,\ldots,n \\
&\omega_j^{(d)}(h_j) = \prod_{i=1}^{m} \lambda_{j,i}^{(d)}(h_j) \\
&\qquad j = 1,\ldots,n \\
&\omega_j^{(u)}(h_j) = p(h_j) \\
&\qquad j = 1,\ldots,n
\end{aligned} \tag{5}$$

As for the scheduling policy, the MP procedure starts by initializing the messages $\lambda_{j,i}^{(u)}(h_j) = p(h_j)$ and $\hat{\nu}_{i,j}^{(d)}(\hat{s}_i) = 1$ and sending them to all $p\left(r_{i,j}\,|s_i,\hat{s}_i,h_j\,\right)$ factors, and by sending the messages $p(s_1)$ and $p(\hat{s}_1)$ to the variable nodes s_1 and $\hat{s}_1$, respectively. Hence, the MP proceeds according to the general message passing rules, until all variable nodes are able to compute the respective marginals, thus concluding the first iteration. Successive iterations are carried out by starting from leaf nodes and by taking into account the messages received at the previous iteration for the evaluation of new messages. The algorithm stops when convergence of messages is achieved, or after a maximum number of iterations.

This version of the MP algorithm described above is an extension of the one proposed in [9], which does not take into account the possibility of synchronized attacks. More specifically, in the attack model considered in [9], the Byzantines independently flip the observations with a given probability P_{mal}, thus yielding

$$\begin{aligned} &p\left(r_{i,j}|s_i, h_j = 0\right) = \\ &(1-\eta)\delta(r_{i,j} - s_i) + \eta(1 - \delta(r_{i,j} - s_i)) \end{aligned} \tag{6}$$

where $\eta = \varepsilon(1 - P_{mal}) + (1 - \varepsilon)P_{mal}$ is the probability of receiving a wrong report from a Byzantine. For the honest nodes, the probability model was the same as in Equation (1).

In order to evaluate the complexity of the algorithm shown in Figure 3, we consider the number of operations performed to estimate the vector of system states $\mathbf{s}$. By number of operations we mean the number of additions, substractions, multiplications and divisions done at the FC for the state estimation.

By looking at equation (5), we see that running the message passing algorithm requires the following number of operations:

- $n+1$ operations for each of $\tau_i^{(l)}(s_i)$, $\tau_i^{(r)}(s_i)$, $\nu_{i,j}^{(d)}(s_i)$, $\hat{\tau}_i^{(l)}(\hat{s}_i)$, $\hat{\tau}_i^{(r)}(\hat{s}_i)$, and $\hat{\nu}_{i,j}^{(d)}(\hat{s}_i)$.
- 3 operations for each of $\varphi_i^{(l)}(s_i)$, $\varphi_i^{(r)}(s_i)$, $\hat{\varphi}_i^{(l)}(\hat{s}_i)$ and $\hat{\varphi}_i^{(r)}(\hat{s}_i)$.
- n operations for each of $\nu_{m,j}^{(d)}(s_m)$ and $\hat{\nu}_{m,j}^{(d)}(\hat{s}_m)$.
- 8 operations for each of $\nu_{i,j}^{(u)}(s_i)$, $\hat{\nu}_{i,j}^{(u)}(\hat{s}_i)$ and $\lambda_{j,i}^{(d)}(h_j)$.
- m operations for each of $\lambda_{j,i}^{(u)}(h_j)$ and $\omega_j^{(u)}(h_j)$.

summing up to $8n+2m+41$ operations for each iteration over the factor graph. Therefore, we can argue that the complexity of the algorithm increases linearly with both n and m in contrast to the complexity the optimum fusion rule presented in [8] which grows exponentially with n.

IV. Simulation Results and Discussion

In this section, we evaluate the performance of the proposed synchronized attacks. We denote the two attack strategies described in Section II-2 as ATTACK_SYNC_FLIP and ATTACK_SYNC_FAKE, respectively. We also compare the performance of these attacks with the unsynchronized attack considered in [8] where the Byzantines act independently from each other and flip the decisions with a given P_{mal}. Specifically, we consider the two cases $P_{mal} = 1.0$ and $P_{mal} = 0.5$, which are the most meaningful cases, as shown in [8]. Simulation results are provided for both the MP-based detector proposed in [9] (referred to as MP_UN) and the MP-based detector proposed in this paper (referred to as MP_SYNC).

We consider the following settings: a sensor network with $n = 20$ nodes, transition probability of the Markovian states $\rho = 0.95$, an observation window $m = 10$, local error probability $\varepsilon = 0.15$, the fraction of Byzantines in the network $\alpha \in [0, 0.45]$ and $\hat{\rho} = \{0.5, 0.95\}$. To evaluate the performance of the MP algorithm, we consider three performance metrics: the probability of decision error P_e, the probability of correct identification of byzantines nodes $P(B|B)$, and the probability of mis-identifying a byzantine node as honest $P(B|H)$. The performance metrics are estimated over 20000 simulations.

Figure 4 shows the performance of the detectors subject to different attacks. As first observation, we can note that both the synchronized attacks have a much more detrimental effect on the system performance than the un-synchronized attacks (bottom-most curves displayed in Figure 4). Moreover, the worst performance is provoked by the ATTACK_SYNC_FAKE strategy with perfect information model estimation, i.e., $\hat{\rho} = \rho$ (upper-most curves displayed in Figure 4). The rationale is twofold: on one side, the sequence of reports sent from the Byzantines does not convey any information to the FC concerning the true states' values (zero-mutual information case); on the other side, in the ATTACK_SYNC_FAKE case, when the fake sequence $\hat{s}$ perfectly matches the state model, the identification of the byzantine nodes become very difficult at the FC. When instead $\hat{\rho} \neq \rho$, the effectiveness of the attack decreases. Indeed, since the Byzantines's reports do not follow exactly the same model as that of the honest nodes, the identification becomes easier. As an example, in Figure 4, it is shown that for $\hat{\rho} = 0.5$, the efficiency of the ATTACK_SYNC_FAKE is considerably reduced and it gives almost the same results of the ATTACK_SYNC_FLIP. Finally, it is worth noting from Figure 4 that the MP_SYNC significantly outperforms the MP_UN in the presence of synchronized attacks. In Figure 5, we report the performance of the MP_SYNC in terms of $P(B|B)$ and $P(B|H)$ to understand how well the MP algorithm can correctly identify the nodes' status. Upon inspection of the figure, we see that identification of the Byzantines is quite good when they adopt the ATTACK_SYNC_FLIP strategy ($P(B|B)$ is around 0.9 and $P(B|H)$ is lower than 0.1). Similar results are obtained for the mismatched ATTACK_SYNC_FAKE case with $\hat{\rho} = 0.5$ and that is why the curves of both cases are superposed on each other. When instead the Byzantines adopt the ATTACK_SYNC_FAKE strategy with perfect estimation of the model, the mission of the detector as expected becomes harder than before (for $\alpha = 0.45$ we have $P(B|B) = 0.7$ and $P(B|H) = 0.25$).

V. Conclusion

We presented two types of synchronized attacks capable to affect the performance of decision fusion in sensor networks. Then, we propose a nearly-optimum detector for coping with synchronized attacks by extending the message passing approach proposed in [9]. Experimental results show that, although the proposed detector is able to mitigate the effect of Byzantines, the coordination of the efforts is very harmful and significantly impairs the detection performance.

References

[1] A. Vempaty, T. Lang, and P. Varshney, "Distributed inference with byzantine data: State-of-the-art review on data falsification attacks," IEEE Signal Processing Magazine, vol. 30, no. 5, Sept 2013, pp. 65–75.

[2] A. S. Rawat, P. Anand, H. Chen, and P. K. Varshney, "Collaborative spectrum sensing in the presence of byzantine attacks in cognitive radio networks," IEEE Transactions on Signal Processing, vol. 59, no. 2, February 2011, pp. 774–786.

[3] Y. Sun and Y. Liu, "Security of online reputation systems: The evolution of attacks and defenses," IEEE Signal Processing Magazine, vol. 29, no. 2, March 2012, pp. 87–97.

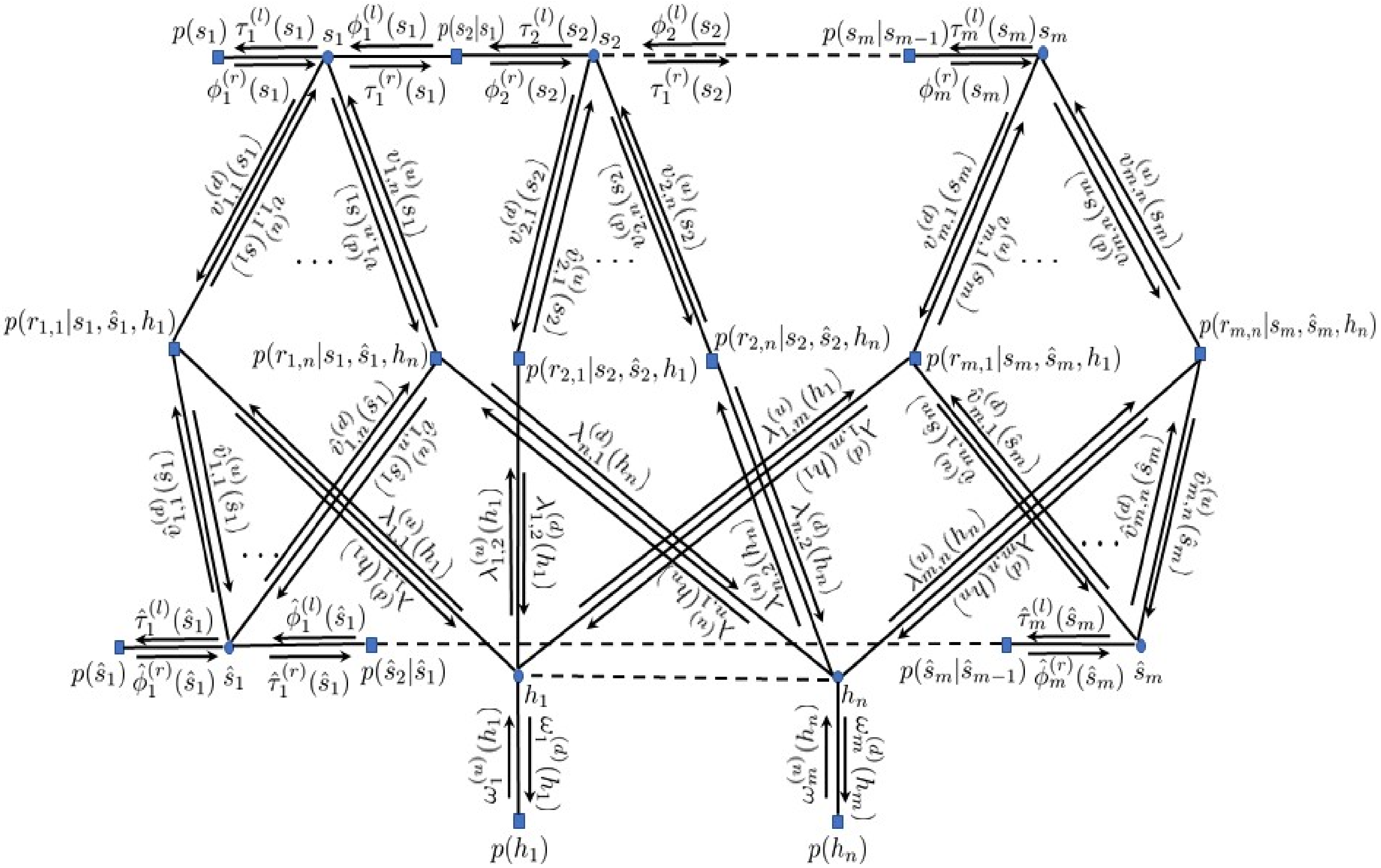
$p(s_1)$
$\tau_1^{(l)}(s_1)$
s_1
$\phi_1^{(l)}(s_1)$
$p(s_2|s_1)$
$\tau_2^{(l)}(s_2)$
s_2
$\phi_2^{(l)}(s_2)$
$p(s_m|s_{m-1})$
$\tau_m^{(l)}(s_m)$
s_m
$\phi_1^{(r)}(s_1)$
$\tau_1^{(r)}(s_1)$
$\phi_2^{(r)}(s_2)$
$\tau_1^{(r)}(s_2)$
$\phi_m^{(r)}(s_m)$
$v_{1,1}^{(d)}(s_1)$
$v_{1,1}^{(u)}(s_1)$
$v_{1,n}^{(u)}(s_1)$
$v_{1,n}^{(d)}(s_1)$
$v_{2,1}^{(d)}(s_2)$
$v_{2,1}^{(u)}(s_2)$
$v_{2,n}^{(d)}(s_2)$
$v_{2,n}^{(u)}(s_2)$
$v_{m,1}^{(d)}(s_m)$
$v_{m,1}^{(u)}(s_m)$
$v_{m,n}^{(u)}(s_m)$
$v_{m,n}^{(d)}(s_m)$
$p(r_{1,1}|s_1,\hat{s}_1,h_1)$
$p(r_{1,n}|s_1,\hat{s}_1,h_n)$
$p(r_{2,1}|s_2,\hat{s}_2,h_1)$
$p(r_{2,n}|s_2,\hat{s}_2,h_n)$
$p(r_{m,1}|s_m,\hat{s}_m,h_1)$
$p(r_{m,n}|s_m,\hat{s}_m,h_n)$
$\hat{v}_{1,1}^{(d)}(\hat{s}_1)$
$\hat{v}_{1,1}^{(u)}(\hat{s}_1)$
$\hat{v}_{1,n}^{(d)}(\hat{s}_1)$
$\hat{v}_{1,n}^{(u)}(\hat{s}_1)$
$\lambda_{1,1}^{(u)}(h_1)$
$\lambda_{1,1}^{(d)}(h_1)$
$\lambda_{n,1}^{(d)}(h_n)$
$\lambda_{n,1}^{(u)}(h_n)$
$\lambda_{1,2}^{(u)}(h_1)$
$\lambda_{1,2}^{(d)}(h_1)$
$\lambda_{1,m}^{(u)}(h_1)$
$\lambda_{1,m}^{(d)}(h_1)$
$\lambda_{n,2}^{(d)}(h_n)$
$\lambda_{n,2}^{(u)}(h_n)$
$\hat{v}_{m,1}^{(d)}(\hat{s}_m)$
$\hat{v}_{m,1}^{(u)}(\hat{s}_m)$
$\hat{v}_{m,n}^{(d)}(\hat{s}_m)$
$\hat{v}_{m,n}^{(u)}(\hat{s}_m)$
$\lambda_{m,n}^{(u)}(h_n)$
$\lambda_{m,n}^{(d)}(h_n)$
$\hat{\tau}_1^{(l)}(\hat{s}_1)$
$\hat{\phi}_1^{(l)}(\hat{s}_1)$
$\hat{\tau}_m^{(l)}(\hat{s}_m)$
$p(\hat{s}_1)$
$\hat{\phi}_1^{(r)}(\hat{s}_1)$
$\hat{s}_1$
$\hat{\tau}_1^{(r)}(\hat{s}_1)$
$p(\hat{s}_2|\hat{s}_1)$
$p(\hat{s}_m|\hat{s}_{m-1})$
$\hat{\phi}_m^{(r)}(\hat{s}_m)$
$\hat{s}_m$
h_1
h_n
$\omega_1^{(u)}(h_1)$
$\omega_1^{(d)}(h_1)$
$\omega_m^{(u)}(h_n)$
$\omega_m^{(d)}(h_m)$
$p(h_1)$
$p(h_n)$
. . .

Figure 3. Factor graph for the problem at hand with the illustration of all the exchanged messages.

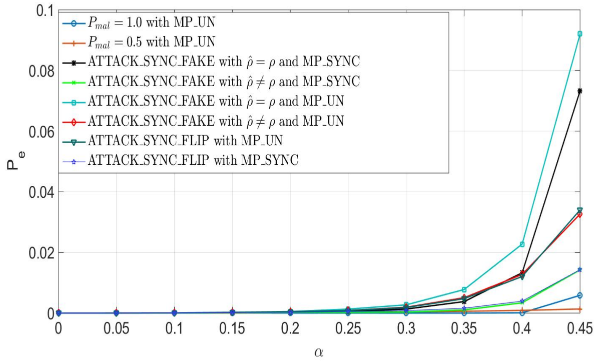

Figure 4. P_e vs. α for various attacks with MP_SYNC and MP_UN for $n = 20$, $\varepsilon = 0.15$, $\rho = 0.95$, $\hat{\rho} = \{0.5, 0.95\}$, and $m = 10$.

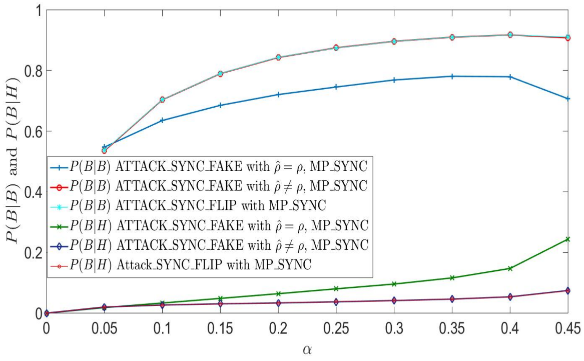

Figure 5. $P(B|B)$ and $P(B|H)$ vs. α for various attacks with MP_SYNC for $n = 20$, $\varepsilon = 0.15$, $\rho = 0.95$, $\hat{\rho} = \{0.5, 0.95\}$, and $m = 10$.

[4] P. K. Varshney, Distributed Detection and Data Fusion. Springer-Verlag, 1997.

[5] S. Marano, V. Matta, and L. Tong, “Distributed detection in the presence of byzantine attacks,” IEEE Transactions on Signal Processing,, vol. 57, no. 1, 2009, pp. 16–29.

[6] A. S. Rawat, P. Anand, H. Chen, and P. K. Varshney, “Collaborative spectrum sensing in the presence of byzantine attacks in cognitive radio networks,” IEEE Transactions on Signal Processing, vol. 59, no. 2, Feb 2011, pp. 774–786.

[7] A. Abrardo, M. Barni, K. Kallas, and B. Tondi, “Decision fusion with corrupted reports in multi-sensor networks: A game-theoretic approach,” in 53rd IEEE Conference on Decision and Control, Dec 2014, pp. 505–510.

[8] A. Abrardo, M. Barni, K. Kallas, and B. Tondi, “A game-theoretic framework for optimum decision fusion in the presence of byzantines,” IEEE Transactions on Information Forensics and Security, vol. 11, no. 6, June 2016, pp. 1333–1345.

[9] A. Abrardo, M. Barni, K. Kallas, and B. Tondi, “A Message Pass-

ing Approach for Decision Fusion in Adversarial Multi-Sensor Networks," submitted to the Information Fusion Journal. ArXiv e-prints 1702.08357, Feb 2017.

[10] S. M. Aji and R. J. McEliece, "The generalized distributive law," IEEE Transactions on Information Theory, vol. 46, no. 2, Mar 2000, pp. 325–343.

[11] K. W. Choi and E. Hossain, "Estimation of primary user parameters in cognitive radio systems via hidden markov model," IEEE Transactions on Signal Processing, vol. 61, no. 3, Feb 2013, pp. 782–795.

[12] I. A. Akbar and W. H. Tranter, "Dynamic spectrum allocation in cognitive radio using hidden markov models: Poisson distributed case," in IEEE Proceedings of SoutheastCon, March 2007, pp. 196–201.

[13] L. Rabiner and B. Juang, "An introduction to hidden markov models," IEEE ASSP Magazine, vol. 3, no. 1, Jan 1986, pp. 4–16.

[14] F. R. Kschischang, B. J. Frey, and H.-A. Loeliger, "Factor graphs and the sum-product algorithm," IEEE Transactions on Information Theory, vol. 47, no. 2, 2001, pp. 498–519.