Bio-inspired System Identification Attacks in Noisy Networked Control Systems

Alan Oliveira de Sá[1,2],
António Casimiro[3],
Raphael Carlos Santos Machado[4,5], and
Luiz Fernando Rust da Costa Carmo[2,4]

[1] Admiral Wandenkolk Instruction Center, Brazilian Navy, RJ, Brazil
[2] Institute of Mathematics/NCE, Federal University of Rio de Janeiro, RJ, Brazil
[3] Department of Informatics, Faculty of Sciences of the University of Lisboa, Portugal.
[4] National Institute of Metrology, Quality and Technology, RJ, Brazil
[5] Rio de Janeiro Federal Center for Technological Education, RJ, Brazil
alan.oliveira.sa@gmail.com, casim@ciencias.ulisboa.pt,
{rcmachado,lfrust}@inmetro.gov.br

Abstract. The possibility of cyberattacks in Networked Control Systems (NCS), along with the growing use of networked controllers in industry and critical infrastructures, is motivating studies about the cybersecurity of these systems. The literature on cybersecurity of NCSs indicates that accurate and covert model-based attacks require high level of knowledge about the models of the attacked system. In this sense, recent works recognize that Bio-inspired System Identification (BiSI) attacks can be considered an effective tool to provide the attacker with the required system models. However, while BiSI attacks have obtained sufficiently accurate models to support the design of model-based attacks, they have demonstrated loss of accuracy in the presence of noisy signals. In this work, a noise processing technique is proposed to improve the accuracy of BiSI attacks in noisy NCSs. The technique is implemented along with a bio-inspired metaheuristic that was previously used in other BiSI attacks: the Backtracking Search Optimization Algorithm (BSA). The results indicate that, with the proposed approach, the accuracy of the estimated models improves. With the proposed noise processing technique, the attacker is able to obtain the model of an NCS by exploiting the noise as a useful information, instead of having it as a negative factor for the performance of the identification process.

Keywords: Security· Networked Control Systems· Cyber-Physical Systems· System Identification· Backtracking Search Algorithm· Bio-inspired Algorithm

1 Introduction

The use of communication networks to integrate controllers and physical processes in a Networked Control Systems (NCS), such as shown in Figure 1, aims to improve management and operational capabilities, as well as reduce costs [10]. However, this integration also exposes the physical plants to new threats originated in the cyber domain.

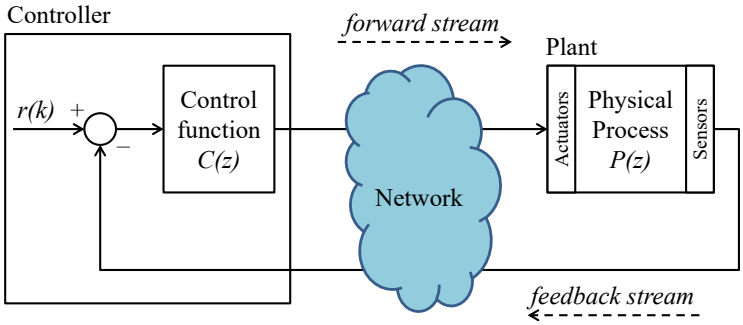


Fig. 1. Networked Control System.

The possibility of sophisticated and large impact attacks in Networked Control Systems (NCS) became unprecedentedly concrete after the launch of the Stuxnet worm [6]. The example of such cyber-physical attack – which is not unique –, along with the growing use of networked controllers in industry and critical infrastructures, has been motivating studies about the cybersecurity of NCSs. In this context, there is a research effort to characterize vulnerabilities, understand attack strategies, and propose security solutions for NCS [1, 3, 7–14].

The literature on cybersecurity of NCSs [1, 9–12, 14] indicates that accurate and covert offensives require high level of knowledge about the models of the attacked system. Exemples of covert attacks that agree with this statement are provided in [11, 12]. In these works the attacks are performed by a man-in-the-middle (MitM), where the attacker needs to know the model of the attacked plant to covertly manipulate the system by injecting false data in both forward and feedback streams. The covertness of the attacks shown in [11, 12] is analyzed from the perspective of the signals arriving to the controller, and depends on the difference between the actual model of the plant and the model known by the attacker. In [1], the authors demonstrate another covert offensive where the attacker, aware of the system's model, injects an attack signal in the NCS to steal water from a canal system located in Southern France.

However, in [1, 11, 12, 14], where the attacks intrinsically require knowledge about the NCS models, it is not described how such knowledge is obtained by the attacker. It is just stated that a model is previously known to subsidize the design of those attacks. More recently, in [9, 10], the authors propose two Bio-inspired System Identification (BiSI) attacks to fill this gap. They demonstrate how the data required to design Denial-of-Service (DoS) or Service Degradation (SD) attacks may be obtained using bio-inspired metaheuristics. Specifically, the attacks proposed in [9, 10] are used to obtain the linear time-invariant (LTI) transfer functions of NCS devices – be it a controller [10], a plant [10], or both in a open loop transfer function [9].

While BiSI attacks have obtained sufficiently accurate models to support the design of model-based attacks, they have demonstrated loss of accuracy in the presence of noisy signals [9]. To overcome this constraint, this work proposes

a noise processing technique to improve the accuracy of BiSI attacks in noisy NCSs. With the proposed strategy, an attacker is able to obtain the model of an NCS by exploiting the noise as a useful information, instead of having it as a negative factor for the performance of the identification process. In this paper, the BiSI attack is implemented using the bio-inspired metaheuristic called Backtracking Search Optimization Algorithm (BSA) [2]. It is worth mentioning that the purpose of this work is not to facilitate cyber-attacks in NCSs. With this study, we aim to encourage the research for techniques capable to enhance the security of NCSs against advanced attacks. Moreover, from the NCS owner perspective, it is worth knowing how an attacker can obtain valuable information about the NCS in case of a lack of confidentiality.

The next sections of this work are organized as follows. Section 2 provides a brief description about the BSA. Section 3 explains the novel noise processing strategy for BiSI attacks. Section 4 shows the results obtained when the noise processing strategy herein proposed is used to support a BiSI attack. Finally, Section 5 brings the conclusions of this work.

2 Backtracking Search Algorithm

This section describes the basic concepts of the BSA, in order to provide a clear understanding about the algorithm parameters that are adjusted when implementing a BSA-based BiSI attack. The BSA is a bio-inspired metaheuristic that searches for solutions of optimization problems using the information obtained by past generations [2] – or iterations. According to [2], its search process is metaphorically analogous to the behavior of a social group of animals that, at random intervals returns to hunting areas previously visited for food foraging. The general, evolutionary like, concept of the BSA is shown in Algorithm 1.

Algorithm 1: BSA

```
begin
    Initialization;
    repeat
        Selection-I;
        Generate new population
            Mutation;
            Crossover;
        end
        Selection-II;
    until Stopping Condition;
end
```

At the Initialization stage, the algorithm generates and evaluates the initial population $\mathcal{P}_0$ and sets the historical population $\mathcal{P}_{hist}$. The latter acts as the memory of the BSA.

In the first selection stage (Selection-I), the algorithm randomly determines, based on an uniform distribution U, whether the current population $\mathcal{P}$ should be kept as the new historical population and, thus, replace $\mathcal{P}_{hist}$ (*i.e.* if $a < b \mid a, b \sim U(0,1)$, then $P_{hist} = P$). After that, it shuffles the individuals of $\mathcal{P}_{hist}$.

The mutation operator creates $\mathcal{P}_{mod}$, which is the preliminary version of the new population $\mathcal{P}_{new}$. The computation of $\mathcal{P}_{mod}$ is performed according to (1):

$$\mathcal{P}_{mod} = \mathcal{P} + \eta \cdot \Gamma(\mathcal{P}_{hist} - \mathcal{P}), \tag{1}$$

wherein η is empirically adjusted through simulations and $\Gamma \sim N(0,1)$, with N being a normal standard distribution. Thus, $\mathcal{P}_{mod}$ is the result of the movement of $\mathcal{P}$'s individuals in the directions established by vector $(\mathcal{P}_{hist} - \mathcal{P})$. In order to create the final version of $\mathcal{P}_{new}$, the crossover operator randomly combines individuals from $\mathcal{P}_{mod}$ and $\mathcal{P}$, also following a uniform distribution.

In the second selection stage (Selection-II), the algorithm evaluates the elements of $\mathcal{P}_{new}$ using a fitness function f, selects the elements of $\mathcal{P}_{new}$ with better fitness than the ones in $\mathcal{P}$, and replaces them in $\mathcal{P}$. Hence, $\mathcal{P}$ includes only new individuals that have evolved. The algorithm iterates until the stopping condition is met. When it occurs, the BSA returns the best solution found.

Note that the algorithm has two parameters that are empirically adjusted: the size $|\mathcal{P}|$ of its population $\mathcal{P}$; and η, that establishes the amplitude of the movements of the individuals of $\mathcal{P}$. The parameter η must be adjusted to assign to the algorithm both good exploration and exploitation capabilities. With this parameters set, the BSA is used to search for the global minimum of the fitness function f described in Section 3.

3 Noise Processing Technique for BiSI attacks

The purpose of the technique presented in this section is to use the white gaussian noise that may be present in an NCS – such as in [9] – in favor of a BiSI attack. With this technique, an attacker is able to accurately estimate the models of an NCS by exploiting the noise as a useful information, instead of having it as a negative factor for the performance of the identification process – which happened in previous implementations of BiSI attacks [9].

The first step of the attack is to eavesdrop the input $i(k)$ and output $o(k)$ signals of the device to be identified, represented in Figure 2. The device can be a controller or a plant. The signals are captured during a monitoring period containing T samples.

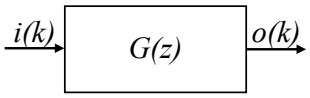


Fig. 2. Device to be identified.

After that, the attacker selects every sample of the eavesdropped input signal $i(k)$ that exceeds a predefined threshold Ω, *i.e.* if (2) is satisfied:

$$i(k) > \Omega, \tag{2}$$

Each sample selected from $i(k)$ according to (2) is referred to as i_n, wherein $n \in \mathbb{Z}_+^*$ is a sequential index number for each selected sample, as exemplified in Figure 3. Additionally, every time that (2) is satisfied, the attacker also stores a portion $o_n(k)$ of the output signal $o(k)$. As represented in Figure 3, each portion $o_n(k)$ selected from $o(k)$ starts when its respective i_n occurs. Each portion $o_n(k)$ encompasses a sequence of τ samples.

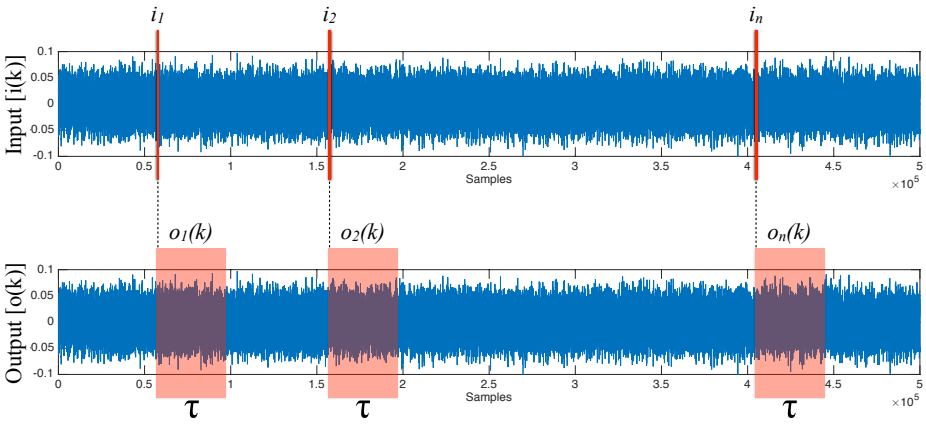


Fig. 3. Selection of noise portions.

After selecting all i_n and $o_n(k)$ from the eavesdropped signals, the attacker computes $\mathcal{I}$ and $\mathcal{O}(k)$ according to (3) and (4), respectively:

$$\mathcal{I} = \frac{\sum_{n=1}^{\mathcal{N}} i_n}{\mathcal{N}}, \tag{3}$$

$$\mathcal{O}(k) = \frac{\sum_{n=1}^{\mathcal{N}} o_n(k)}{\mathcal{N}}, \tag{4}$$

wherein $\mathcal{N}$ is the index number of the last sample i_n obtained from $i(k)$ based on (2). In the present approach, $\mathcal{I}$ corresponds to the amplitude of an impulse signal $\mathcal{I}(k)$ (5) that, when applied to $G(z)$, produces $\mathcal{O}(k)$ – the impulse response function of $G(z)$.

$$\mathcal{I}(k) = \mathcal{I}\delta(k). \tag{5}$$

Now, to estimate $G(z)$, the attacker applies $\mathcal{I}(k)$ to the input of an estimated model $G_e(z)$ defined by (6):

$$G_e(z) = \frac{\mathcal{Z}[\hat{\mathcal{O}}(k)]}{\mathcal{Z}[\mathcal{I}(k)]} = \frac{\alpha_p z^p + \alpha_{p-1} z^{p-1} + ... + \alpha_1 z^1 + \alpha_0}{z^q + \beta_{q-1} z^{q-1} + ... + \beta_1 z^1 + \beta_0}, \tag{6}$$

wherein $\hat{\mathcal{O}}(k)$ is the output provided by the estimated model $G_e(z)$, and $\mathcal{Z}$ represents the Z-transform operation. See that, $[\alpha_p, \alpha_{p-1}, ..., \alpha_1, \alpha_0, \beta_{q-1}, \beta_{q-2}, ...\beta_1, \beta_0]$ is the set of coefficients of $G(z)$ that the BiSI attack aims to discover, wherein p and q represent the order of the numerator and denominator, respectively. Therefore, to obtain the model of the actual device $G(z)$, the parameters of the estimated model $G_e(z)$ are modified and adapted until the output $\hat{\mathcal{O}}(k)$ of $G_e(z)$ converges to $\mathcal{O}(k)$. To do so, the BSA iteratively adjusts the parameters of $G_e(z)$ by minimizing a fitness function f, until $G_e(z)$ meets $G(z)$. The coordinates $x_j = [\alpha_{p,j}, \alpha_{p-1,j}, ...\alpha_{1,j}, \alpha_{0,j}, \beta_{q-1,j}, \beta_{q-2,j}, ...\beta_{1,j}, \beta_{0,j}]$ of each individual j of the BSA are assigned as the coefficients of an estimated model $G_e(z)$. The fitness f_j of each individual j of the BSA is computed according to (7):

$$f_j = \frac{\sum_{k=1}^{\tau} \left[\mathcal{O}(k) - \hat{\mathcal{O}}_j(k)\right]^2}{\tau}. \tag{7}$$

Recall, from Figure 3, that τ is the number of samples contained in each portion $o_n(k)$ of $o(k)$, and, therefore, is also the number of samples contained in $\mathcal{O}(k)$ and $\hat{\mathcal{O}}_j(k)$. The signal $\hat{\mathcal{O}}_j(k)$ is the output of $G_e(z)$ (6) when its coefficients are defined as x_j. From (7) it is possible to see that $\min f_j = 0$ if $\mathcal{O}(k) = \hat{\mathcal{O}}_j(k)$. This result is achieved whenever $[\alpha_{p,j}, \alpha_{p-1,j}, \ldots, \alpha_{1,j}, \alpha_{0,j}, \beta_{q-1,j}, \beta_{q-2,j}, \ldots, \beta_{1,j}, \beta_{0,j}]$ $= [\alpha_p, \alpha_{p-1}, \ldots, \alpha_1, \alpha_0, \beta_{q-1}, \beta_{q-2}, \ldots, \beta_1, \beta_0]$ or, in other words, when $G_e(z) = G(z)$.

Algorithm 2: BiSI attack with the noise processing strategy

```
begin
    Eavesdrop i(k) and o(k) during T samples;
    Noise Processing
        Select all i_n and the respective o_n(k), ∀i(k) > Ω;
        Compute I(k) and O(k) according to (3), (4) and (5);
    end
    Execute BSA, using I(k) and O(k) to find G(z).
end
```

The Algorithm 2 briefly describes the complete BiSI attack with the proposed noise processing strategy. Albeit the BiSI attack herein proposed uses the same bio-inspired metaheuristic used in [9] (*i.e.*, the BSA, concisely described in Section 2 as in [9]), its is worth mentioning the differences from the present attack and the BiSI attack of [9]:

– In [9] the attacker injects an attack signal in the system to identify its transfer function. In that approach, the presence of noise affects the ability of the

attack to learn the system model from the outputs caused by the attack signal. On the other hand, in the present work, the attacker does not injects an attack signal in the system. Conversely, the attacker passively collects the noisy signals and use them to estimate the system transfer function.
– The approach presented in [9] does not use the Noise Processing technique herein proposed.

4 Results

This section presents an evaluation on the performance of the BiSI attack with the noise processing strategy presented in Section 3. The model of the attacked device – *i.e.*, the device to be identified – is represented by (8). In practice, such second order transfer function can represent, for instance, a DC motor [4] or a lighting system [5] (among other systems). However, it is worth mentioning that, depending on the system characteristics, the coefficients of such plants can be different from the example defined by (8).

$$G(z) = \frac{\mathcal{Z}[o(k)]}{\mathcal{Z}[i(k)]} = \frac{2}{z - 0.9}. \tag{8}$$

The sample rate is 50 samples/s, and the noise measured in the input of $G(z)$ is a white gaussian noise $w(k) \sim N(\mu, \sigma)$, wherein N is a normal distribution with mean $\mu = 0$ and standard deviation $\sigma = 0.005$. This way, 95% of the amplitudes of $w(k)$ are within ± 0.01 (2σ).

The results of this section were obtained through simulations using MATLAB/SIMULINK. The evaluate the benefits – in terms of accuracy – provided by the noise processing technique described in Section 3, two BiSI attacks are implemented for comparison:

(I) a BiSI attack using the noise processing technique along with the BSA optimization process, such as described in Section 3;
(II) a BiSI attack using only the BSA optimization process (*i.e.*, without the noise processing stage). In this case, the eavesdropped signals $i(k)$ and $o(k)$ are directly used – without treatment – by the BSA to estimate the parameters of $G_e(z)$. To do so, equations (6) and (7) – used to compute the fitness of BSA individuals – are rewritten as (9) and (10), and the BiSI attack is simply represented by Algorithm 3.

$$G_e(z) = \frac{\mathcal{Z}[\hat{o}(k)]}{\mathcal{Z}[i(k)]} = \frac{\alpha_p z^p + \alpha_{p-1} z^{p-1} + ... + \alpha_1 z^1 + \alpha_0}{z^q + \beta_{q-1} z^{q-1} + ... + \beta_1 z^1 + \beta_0}, \tag{9}$$

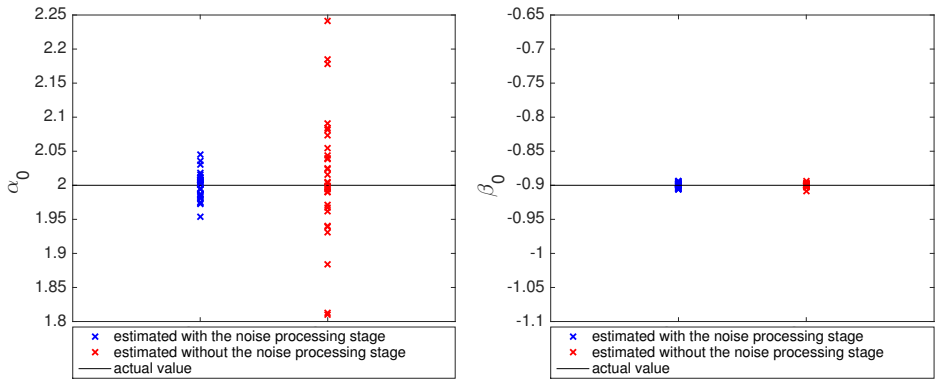
$$f_j = \frac{\sum_{k=1}^{\tau} \left[o(k) - \hat{o}_j(k)\right]^2}{\tau}. \tag{10}$$

Algorithm 3: BiSI attack without the noise processing strategy

```
begin
    Eavesdrop i(k) and o(k) during τ samples;
    Execute BSA, using i(k) and o(k) to find G(z).
end
```

As previously discussed, the BiSI attack aims to estimate the coefficients of the LTI transfer function of an NCS device. Therefore, in the present simulations, the parameters to be identified – according to (8) – are $\alpha_0 = 2$ and $\beta_0 = 0.9$. The BSA configurations in this paper are the same as those used in [9, 10]: the lower and upper limits of each search space dimension are -10 and 10, respectively; the number of individuals in the BSA population is 100; $\eta = 1$; and the stopping criteria is 600 iterations. Moreover, $T = 0,5Msamples$, $\tau = 100samples$ and $\Omega = 0.01$.

Each of the BiSI attack implementations – (I) and (II) – are evaluated through 31 different simulations. Each simulation uses a different white gaussian noise signal, randomly generated. Figure 4 shows the 31 values of α_0 and β_0 estimated by the two BiSI attack implementations (*i.e.*, with and without the noise processing stage). Additionally, Table 1 shows the statistics of the results presented in Figure 4. From Figure 4 and Table 1, it is possible to verify that the accuracy of the BiSI attack with the noise processing stage is better than the accuracy of the BiSI attack without the proposed technique. Figure 4(b) indicates that the two implementations have similar performance when estimating β_0. In both implementations, all estimated β_0 are close to the actual β_0 and, according to Table 1, the standard deviations are similarly low. On the other hand, Figure 4(a) demonstrates that implementation (I) has better performance than implementation (II) when estimating α_0. With the noise processing stage, the estimated values of α_0 are closer to the actual α_0 – *i.e.*, less spread than without the noise processing stage. The statistics shown in Table 1 ratifies the better performance provided by the noise processing stage when the BiSI attack estimates α_0. In this case, the mean of the estimated values is closer to the actual α_0, with lower standard deviation.



(a) Estimations of α_0 (b) Estimations of β_0

Fig. 4. Estimations of α_0 and β_0 with and without the noise processing stage.

Table 1. Statistics of the BiSI attacks

Coefficient	BiSI attack Implementation	Mean	Standard Deviation
α_0	(I)	1.9997	0.0189
	(II)	2.0119	0.0911
β_0	(I)	-0.8999	0.0034
	(II)	-0.8998	0.0024

Figure 5, obtained from one example of BiSI attack using implementation (I), compares the impulse response function $\mathcal{O}(k)$ of $G(z)$ – computed by the noise processing stage – with the impulse response function $\hat{\mathcal{O}}(k)$ of the estimated model $G_e(z)$. Note that, this figure demonstrates the product of the work done by the noise processing stage: a clear impulse response function, extracted from a white gaussian noise, that is better handled by the bio-inspired identification process performed by the BSA. It is possible to see how close $\hat{\mathcal{O}}(k)$ is from $\mathcal{O}(k)$, which demonstrates the high accuracy of the estimated model $G_e(z)$ when the BSA-based identification uses the signals provided by the proposed noise processing stage.

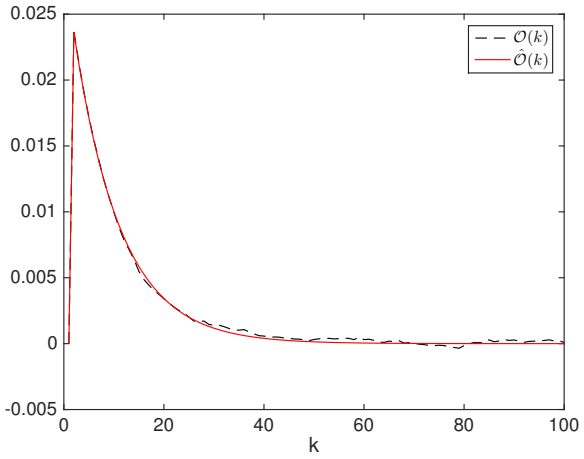


Fig. 5. Evaluation of the performance of the identification process – comparison between $\mathcal{O}(k)$ and $\hat{\mathcal{O}}(k)$.

5 Conclusion

In this work we propose a noise processing technique to improve the accuracy of bio-inspired system identification algorithms. The simulation results indicate that when the proposed technique is performed prior to the BSA-based system identification process, the accuracy of the estimated model increases. Therefore, the present technique represent a useful tool to make BiSI attacks effective in

noisy NCSs. The proposed technique overcomes the constraint presented in other implementations of BiSI attacks, where the accuracy of estimated models used to be degraded by noise. The outcomes indicates that, with this approach, noise may not be a problem for a BiSI attack. Instead, noise can represent a meaningful and useful information for an attacker if he/she uses the approach described in this paper.

For future work, we plan to investigate techniques to mitigate BiSI attacks, by hindering the identification process in situations where an attacker has access to the data flowing in the NCS. Moreover, we plan to investigate the use of the proposed algorithm as a defense tool to identify possible model-based attacks in noisy NCSs. In this sense, we believe that this algorithm can be used to provide the NCS with information regarding the model of an eventual attack, in order to allow the autonomous reconfiguration of the control function to compensate the presence of the attack.

Acknowledgements

This work was partially sponsored by the EU-BR SecureCloud project (MCTI/ RNP 3rd Coordinated Call), by the Coordination for the Improvement of Higher Education Personnel (CAPES), grant 99999.008512/2014-0, and by FCT through project LaSIGE (UID/CEC/00408/2013).

References

1. Amin, S., Litrico, X., Sastry, S., Bayen, A.M.: Cyber security of water scada systems part i: analysis and experimentation of stealthy deception attacks. IEEE Transactions on Control Systems Technology **21**(5), 1963–1970 (2013)
2. Civicioglu, P.: Backtracking search optimization algorithm for numerical optimization problems. Applied Mathematics and Computation **219**(15), 8121–8144 (2013)
3. Farooqui, A.A., Zaidi, S.S.H., Memon, A.Y., Qazi, S.: Cyber security backdrop: A scada testbed. In: Computing, Communications and IT Applications Conference (ComComAp), 2014 IEEE. pp. 98–103. IEEE (2014)
4. Ferrari, P., Flammini, A., Rizzi, M., Sisinni, E.: Improving simulation of wireless networked control systems based on wirelesshart. Computer Standards & Interfaces **35**(6), 605–615 (2013)
5. Ji, K., Wei, D.: Resilient control for wireless networked control systems. International Journal of Control, Automation and Systems **9**(2), 285–293 (2011)
6. Langner, R.: Stuxnet: Dissecting a cyberwarfare weapon. Security & Privacy, IEEE **9**(3), 49–51 (2011)
7. Long, M., Wu, C.H., Hung, J.Y.: Denial of service attacks on network-based control systems: impact and mitigation. Industrial Informatics, IEEE Transactions on **1**(2), 85–96 (2005)
8. de Sá, A.O., da Costa Carmo, L.F., Machado, R.C.: A controller design for mitigation of passive system identification attacks in networked control systems. Journal of Internet Services and Applications **9**(1), 2 (2018)

9. de Sá, A.O., Carmo, L.F.d.C., Machado, R.C.: Bio-inspired active system identification: a cyber-physical intelligence attack in networked control systems. Mobile Networks and Applications pp. 1–14 (2017)
10. de Sá, A.O., da Costa Carmo, L.F.R., Machado, R.C.: Covert attacks in cyber-physical control systems. IEEE Transactions on Industrial Informatics **13**(4), 1641–1651 (2017)
11. Smith, R.: A decoupled feedback structure for covertly appropriating networked control systems. In: Proceedings of the 18th IFAC World Congress 2011. vol. 18. IFAC-PapersOnLine (2011)
12. Smith, R.S.: Covert misappropriation of networked control systems: Presenting a feedback structure. Control Systems, IEEE **35**(1), 82–92 (2015)
13. Snoeren, A.C., Partridge, C., Sanchez, L.A., Jones, C.E., Tchakountio, F., Schwartz, B., Kent, S.T., Strayer, W.T.: Single-packet ip traceback. IEEE/ACM Transactions on Networking (ToN) **10**(6), 721–734 (2002)
14. Teixeira, A., Shames, I., Sandberg, H., Johansson, K.H.: A secure control framework for resource-limited adversaries. Automatica **51**, 135–148 (2015)