SCIENCE CHINA
Information Sciences

• RESEARCH PAPER •

June 2022, Vol. 65 162201:1–162201:9
https://doi.org/10.1007/s11432-020-3027-0

Deep reinforcement learning for optimal denial-of-service attacks scheduling

Fangyuan HOU[1], Jian SUN[1*], Qiuling YANG[1] & Zhonghua PANG[2]

[1]*State Key Laboratory of Intelligent Control and Decision of Complex Systems, Beijing Institute of Technology, Beijing* 100081, *China;*
[2]*Key Laboratory of Fieldbus Technology and Automation of Beijing, North China University of Technology, Beijing* 100144, *China*

Received 24 February 2020/Revised 30 April 2020/Accepted 29 June 2020/Published online 18 April 2022

Abstract We consider an optimal denial-of-service (DoS) attack scheduling problem of N independent linear time-invariant processes, where sensors have limited computational capability. Sensors transmit measurements to the remote estimator via a communication channel that is exposed to DoS attackers. However, due to limited energy, an attacker can only attack a subset of sensors at each time step. To maximally degrade the estimation performance, a DoS attacker needs to determine which sensors to attack at each time step. In this context, a deep reinforcement learning (DRL) algorithm, which combines Q-learning with a deep neural network, is introduced to solve the Markov decision process (MDP). The DoS attack scheduling optimization problem is formulated as an MDP that is solved by the DRL algorithm. A numerical example is provided to illustrate the efficiency of the optimal DoS attack scheduling scheme using the DRL algorithm.

Keywords optimal denial-of-service attack, scheduling, optimization, limited energy, deep reinforcement learning

Citation Hou F Y, Sun J, Yang Q L, et al. Deep reinforcement learning for optimal denial-of-service attacks scheduling. Sci China Inf Sci, 2022, 65(6): 162201, https://doi.org/10.1007/s11432-020-3027-0

1 Introduction

Cyber physical systems (CPSs) that integrate sensing, computation, control, and communication [1–3] have a wide range of applications, such as smart grids, smart transportation, health care, and other critical infrastructures [4–6]. In CPSs, sensor measurements are transmitted over networks [7, 8], which are susceptible to be corrupted by an attacker [9]. For example, Stuxnet has been reported to attack the supervisory control and data acquisition in the nuclear power generation systems [10]. Such security incidents inspired researchers to study security issues in CPSs.

Malicious cyber attacks have diverse forms, and various cyber attacks in the attack space have been demonstrated [11]. Typical network attacks can be classified as deception attacks and denial-of-service (DoS) attacks. In a deception attack, without being detected, an attacker transmits false data to the controller or the actuator [12]. A false data injection attack, a particular type of deception attack, in a control system has been designed [13]. A worst-case innovation-based linear attack strategy has also been proposed [14]. In addition, an optimal switching false data injection attack in the actuator under energy-constraints has been designed [15]. Replay attacks are another type of deception attacks. In replay attacks, the attacker first records a sufficient number of sensor measurements from $k = k_0$ to k_T, and then begins replaying the recorded data at time $k = k_T + 1$ until the end of the attack [16]. A zero-dynamics attack, a particular type of deception attacks, corresponds to the input sequence that makes the outputs of a system identically zero [17], which requires perfect model knowledge. However, another type of deception attack, i.e., a covert attack, can be designed based on imperfect model knowledge [18].

Compared to deception attacks, a DoS attack requires much less knowledge about the target system. DoS attacks attempt to block the communication channel and prevent legitimate access between system

* Corresponding author (email: sunjian@bit.edu.cn)

components [19,20]. DoS attacks can significantly deteriorate the performance of system state estimation in smart grids [21]. The stability analysis of CPSs in the presence of a DoS attack has been investigated [22]. An optimal DoS attack in a two-hop network has also been designed [23]. How a mobile ad hoc network can be affected by distributed DoS (DDoS) attacks has been studied [24], and a novel solution to handle DDoS attacks in a mobile ad hoc network was proposed. In [25], a novel flow-table sharing approach was proposed to protect the software defined network (SDN)-based cloud from table overloading DDoS attacks. The comparison between single tier and three tier architectures against DDoS attacks has been studied [26]. In that study, it has demonstrated that the three tier architecture has significantly more resilience against DDoS attacks. An online intrusion detection system based on the NeuCube algorithm was proposed to detect malicious attacks in cloud computing, especially the zero-day attack [27]. Practically, the attacker cannot attack all sensors simultaneously over time due to limited energy.

Therefore, various efforts have focused on studying the optimal scheduling of energy-constrained DoS attacks in CPSs. An optimal energy-constrained DoS attack scheduling that maximized the linear quadratic Gaussian control of a wireless networked control system has been proposed [28]. In [29], an attack power allocation algorithm was proposed to maximize the performance of the DoS attack. The optimal DoS attack energy management problem in a signal-to-interference-plus-noise ratio-based network has also been investigated [30]. In addition, an optimal attack scheduling scheme in a packet-dropping network was derived [31]. In that study, some countermeasures against DoS attacks were proposed, and the optimal defense strategy was discussed. In [32], the authors addressed an optimal DoS attack problem considering a scenario where the observations are transmitted through a standard block fading communication channel. In previous studies, the authors had assumed that the sensor had sufficient computation capability and that the local state estimate was transmitted to the remote estimator. Unfortunately, it is a strong assumption in real-world CPSs. In [33], the authors analyzed the DoS attack in sensor measurements with limited energy and obtained a suboptimal solution of the optimization problem based on convex relaxation, while incurring high computational complexities. The solution of the optimization problem in [33] depends on the initializations of attack variables. If the initializations were not selected well, there may be no feasible solutions. Differing from [33], in this paper, a deep reinforcement learning (DRL) algorithm, which is independent of the initializations of attack variables, is introduced to solve the optimal DoS attack scheduling problem in an iterative model-free manner.

Motivated by the above discussions of previous studies, in this paper sensors with limited computation only transmit the sensor measurements to the remote estimator via a communication channel that is exposed to an attacker. A DoS attacker attempts to destroy the estimation error covariance by jamming the communication channel. However, due to the limited energy, a DoS attacker cannot attack all sensors simultaneously all the time. To maximally degrade the estimation performance, a DoS attacker needs to determine which sensors to attack at each time step. It is NP-hard to find a feasible solution for the optimization problem that involves binary constraint. Therefore, a DRL algorithm that combines Q-learning with a deep neural network is introduced to solve the optimal DoS attack scheduling problem which is modeled as a Markov decision process (MDP).

The primary contributions of this paper are summarized as follows.

• An optimal DoS attack scheduling scheme of the sensor measurements that maximizes the estimation error covariance is designed. This approach differs from most existing studies that focus on scheduling the local state estimate.

• A DRL algorithm is introduced to solve the optimal DoS attack scheduling problem to find the suboptimal policy. Again, this approach differs from most previous studies that obtain a suboptimal solution based on convex relaxation, which depends on the initializations of attack variables.

The rest of the paper is organized as follows. Section 2 formulates the system model and DoS attacks. In Section 3, a DRL algorithm for optimal DoS attacks scheduling scheme is presented. Section 4 illustrates the efficiency of the optimal DoS attack scheduling scheme by a numerical example. The conclusion is drawn in Section 5.

Notation. $\mathbb{R}$ denotes the set of real numbers, and $\mathbb{R}^n$ denotes n-dimensional Euclidean space. $\mathrm{E}[\cdot]$ refers to mathematical expectation, and $\mathrm{E}(X|Y)$ refers to the mathematical expectation of random variable X conditioned on Y. X^{T} denotes the transpose of matrix X.

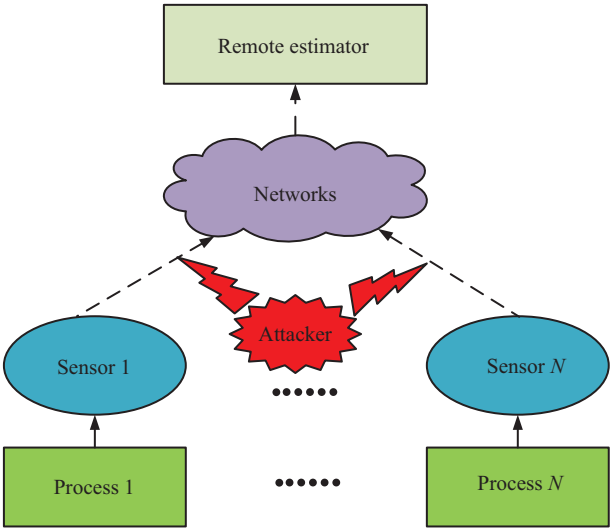

Figure 1 (Color online) System architecture.

2 Problem formulation

2.1 System model

Consider N independent discrete-time processes (see Figure 1)

$$x_{i,k+1} = A_i x_{i,k} + \omega_{i,k}, \quad i = 1, \ldots, N, \tag{1}$$

where $x_{i,k} \in \mathbb{R}^{n_i}$ is the state vector of subsystem i at time k; the initial state of subsystem i, $x_{i,0}$, is i.i.d. Gaussian random variable with $x_{i,0} \sim \mathcal{N}(\bar{x}_{i,0}, \Sigma_{i,0})$; $\omega_{i,k}$ is zero mean i.i.d. Gaussian random variable with $\omega_{i,k} \sim \mathcal{N}(0, Q_i)$.

Each sensor measurement is available, and

$$y_{i,k} = C_i x_{i,k} + \upsilon_{i,k}, \tag{2}$$

where $y_{i,k}$ is sensor measurement vector of subsystem i at time k; $\upsilon_{i,k}$ is zero mean i.i.d. Gaussian random variable with $\upsilon_{i,k} \sim \mathcal{N}(0, R_i)$. Assume that $(A_i, \sqrt{Q_i})$ is controllable and (A_i, C_i) is observable.

The remote estimator is equipped with a Kalman filter computing the minimum mean-squared error (MMSE) estimate of $x_{i,k}$ by the sensor measurement $y_{i,k}$ which is transmitted via networks. The prior MMSE estimate and its corresponding error covariance are defined by

$$\hat{x}_{i,k|k-1} \triangleq \mathrm{E}[x_{i,k}|y_{i,0}, \ldots, y_{i,k-1}], \tag{3}$$

$$P_{i,k|k-1} \triangleq \mathrm{E}[(x_{i,k} - \hat{x}_{i,k|k-1})(x_{i,k} - \hat{x}_{i,k|k-1})^{\mathrm{T}}|y_{i,0}, \ldots, y_{i,k-1}]. \tag{4}$$

It is obtained that $\hat{x}_{i,k|k-1}$ and $P_{i,k|k-1}$ are as follows:

$$\hat{x}_{i,k|k-1} = A_i \hat{x}_{i,k-1|k-1}, \tag{5}$$

$$P_{i,k|k-1} = A_i P_{i,k-1|k-1} A_i^{\mathrm{T}} + Q_i, \tag{6}$$

where $\hat{x}_{i,k-1|k-1}$ is the posteriori MMSE estimate and $P_{i,k-1|k-1}$ is the corresponding error covariance. The posteriori MMSE estimate and its corresponding error covariance are defined by

$$\hat{x}_{i,k|k} \triangleq \mathrm{E}[x_{i,k}|y_{i,0}, \ldots, y_{i,k}], \tag{7}$$

$$P_{i,k|k} \triangleq \mathrm{E}[(x_{i,k} - \hat{x}_{i,k|k})(x_{i,k} - \hat{x}_{i,k|k})^{\mathrm{T}}|y_{i,0}, \ldots, y_{i,k}]. \tag{8}$$

$\hat{x}_{i,k|k}$ and $P_{i,k|k}$ are computed by

$$\hat{x}_{i,k|k} = \hat{x}_{i,k|k-1} + K_{i,k}(y_{i,k} - C_i \hat{x}_{i,k|k-1}), \tag{9}$$

$$P_{i,k|k} = P_{i,k|k-1} - K_{i,k}C_iP_{i,k|k-1}, \tag{10}$$

where $K_{i,k}$ is Kalman filter gain and computed as follows:

$$K_{i,k} = P_{i,k|k-1}C_i^{\mathrm{T}}(C_iP_{i,k|k-1}C_i^{\mathrm{T}} + R_i)^{-1}. \tag{11}$$

2.2 DoS attacks

We assume that a DoS attacker can only attack a subset of sensors due to the limited energy. In other words, only $M < N$ sensors can be attacked at each time step. To maximally degrade the estimation performance, a DoS attacker needs to determine which sensors to attack at each time step. Define the attacker's decision variable at time k

$$\gamma_{i,k} = \begin{cases} 1, & \text{if sensor } i \text{ is attacked,} \\ 0, & \text{otherwise.} \end{cases} \tag{12}$$

Therefore, at time k, the received sensor measurements in the remote estimator are

$$\tilde{y}_{i,k} = (1 - \gamma_{i,k})y_{i,k}. \tag{13}$$

In the remote estimator, the state estimate and estimation error covariance are computed based on the attacked sensor measurements

$$K_{i,k} = P_{i,k|k-1}\tilde{C}_i^{\mathrm{T}}(\tilde{C}_iP_{i,k|k-1}\tilde{C}_i^{\mathrm{T}} + \tilde{R}_i)^{\dagger}, \tag{14}$$

$$\hat{x}_{i,k|k} = \hat{x}_{i,k|k-1} + K_{i,k}(\tilde{y}_{i,k} - \tilde{C}_i\hat{x}_{i,k|k-1}), \tag{15}$$

$$P_{i,k|k} = P_{i,k|k-1} - K_{i,k}\tilde{C}_iP_{i,k|k-1}, \tag{16}$$

where $\tilde{C}_i = (1 - \gamma_{i,k})C_i$, $\tilde{R}_i = (1 - \gamma_{i,k})R_i$, $\dagger$ denotes the Moore-Penrose pseudo-inverse.

At time k, a DoS attacker determines which M sensors among N sensors to attack. The optimal DoS attack scheduling problem can be written as

$$\begin{aligned} &\max_{\gamma} \lim_{T\to\infty} \frac{1}{T}\sum_{k=1}^{\mathrm{T}}\sum_{i=1}^{N} \mathrm{trace}(P_{i,k|k}), \\ &\text{s.t.} \sum_{i=1}^{N} \gamma_{i,k} \leqslant M, \quad \gamma_{i,k} \in \{0, 1\}. \end{aligned} \tag{17}$$

It should be noted that the last constraint in (17) involves binary constraint. Therefore the optimization problem (17) is nonconvex optimization and NP-hard to find a feasible solution. In previous studies, they obtain a suboptimal solution based on convex relaxation while increasing computational complexity and depending on the initializations of attack variables. To solve this challenging problem, a DRL algorithm is introduced in the next section to solve the optimal DoS attack scheduling problem.

3 Deep reinforcement learning for DoS attacks scheduling

A DRL algorithm, which combines Q-learning with a deep neural network, is introduced to solve the associated MDP in an iterative model-free manner [34]. The optimal DoS attack scheduling problem can be formulated as an MDP. At each time step k, an attacker makes decisions about which sensors to attack, that is, the decision maker chooses an action. At the next time step, the process responds to a new average estimation error covariance, and provides a corresponding reward to the attacker. An MDP $\mathcal{M}$ is formulated as the following tuple $(\mathcal{S}, \mathcal{A}, r, \alpha)$ problem.

State space $\mathcal{S}$. At time k, the state space $\mathcal{S}$ contains all subsystem estimation error covariance. The state space $\mathcal{S}$ is defined as the finite set $(P_{1,k-1}, \ldots, P_{N,k-1})$, where the size of state space is N.

Action space $\mathcal{A}$. At each decision epoch, a DoS attacker chooses action a from the action space $\mathcal{A}$. The size of the action space is C_N^M, which means selecting M sensors out of N sensors to attack.

Reward r. $r_k = \sum_{i=1}^{N} \mathrm{trace}(P_{i,k|k})$ is the one stage reward function, and specifies the reward for a DoS attacker choosing an action in current state.

Discount factor α. The discount factor $\alpha \in [0, 1]$ trades off between the current and future costs, which hastens the rate of convergence.

Note that at each time step, a DoS attacker selects an action a_k from the allowable actions $\mathcal{A}$; the action is taken in the current estimation error covariance. Then a DoS attacker receives a new estimation error covariance and a corresponding reward r_k representing the feedback of the estimation error covariance.

To maximize the future reward by selecting an optimal policy, consider the future discounted return at time k_0 as

$$R(k_0) = \lim_{T\to\infty} \mathrm{E}\left[\sum_{k=k_0}^{T} \alpha^{k-k_0} r_k\right]. \tag{18}$$

Given the current state and action, the optimal action-value function $Q^*(s, a)$ is defined as the maximum expected return under the control policy π [34],

$$Q^*(s, a) = \max_{\pi} \mathrm{E}[R_k | s_k = s, a_k = a, \pi], \tag{19}$$

where π is a policy that maps estimation error covariance to action. The optimal action-value function $Q^*(s, a)$ represents the expected future return associated with taking action a' maximizing $r+\alpha Q^*(s', a')$. The Bellman equation for the optimal action-value function is given by

$$Q^*(s, a) = \mathrm{E}\left[r + \alpha \sum_{s'\in\mathcal{S}} \mathbb{P}_{ss'} \max_{a'\in\mathcal{A}} Q^*(s', a') | s, a\right], \tag{20}$$

where s' denotes the next time step estimation error covariance given by the current estimation error covariance s and action a, $\mathbb{P}_{ss'}$ denotes the state transition probability matrix from the current estimation error covariance s to the next time step estimation error covariance s'. The optimal DoS attack scheduling policy can be obtained by

$$\pi^*(s) = \operatorname*{argmax}_{a} Q^*(s, a). \tag{21}$$

If the state transition probabilities are known to the attacker, the action-value function will converge to the optimal action-value function $Q^*(s, a)$ and obtain the optimal policy π'. Unfortunately, the transition probabilities in a real-world CPS are impossible to obtain. To obtain the optimal policy π' for large MDP without having the knowledge of transition probabilities, a deep neural network function approximator to estimate the action-value function $Q(s, a; \theta) \approx Q^*(s, a)$ was introduced in [34]. A DRL algorithm has gained popularity because it performs remarkably well in solving an MDP in an iterative model-free manner and dealing with high-dimensional state and action spaces such as in the playing Atari [35]. To estimate the costs for all possible DoS attacks scheduling schemes, the deep Q-network (DQN) needs to be trained by iteratively updating θ_k [36]. A DQN can be trained by adjusting the parameters θ_k at iteration k to reduce the mean-squared error in the Bellman equation. The lossy function $\mathcal{L}_k(\theta_k)$ used in Q-learning updates at iteration k is given by

$$\mathcal{L}_k(\theta_k) = \mathrm{E}_{s,a,r,s'}\left[\left(r + \alpha \max_{a'} Q(s', a'; \theta_k^-) - Q(s, a; \theta_k)\right)^2\right], \tag{22}$$

where θ_k^- is the target network parameter. The target network, with parameters θ_k^-, is the same as the online network except that its parameters are updated every c steps from the online network, so that $\theta_k^- = \theta_k$, and kept fixed on all other steps [37, 38].

Stochastic gradient descent (SGD) is used to optimize the loss function to get the weight parameters of the DQN, which is given by

$$\theta_{k+1} = \theta_k - \beta \nabla_{\theta_k} \mathcal{L}_k(\theta_k), \tag{23}$$

where β is the learning rate, and $\nabla_{\theta_k} \mathcal{L}_k(\theta_k)$ denotes the gradient of the loss function with respect to the weights

$$\nabla_{\theta_k} \mathcal{L}_k(\theta_k) = \mathrm{E}_{s,a,r,s'}\left[r + \alpha \max_{a'} Q(s', a'; \theta_k^-) - Q(s, a; \theta_k) \nabla_{\theta_k} Q(s, a; \theta_k)\right]. \tag{24}$$

To break the correlations among data sequences, experience replay [35] technique is adopted. The k-th experience is defined as $e_k = (s_k, a_k, r_k, s_{k+1})$. Different experiences are stored into a replay memory $\mathcal{D}_k = \{e_1, \ldots, e_k\}$. During learning, we uniformly draw a minibatch of experiences at random from the replay memory $\mathcal{D}_k$ to update the network parameter θ_k. The optimal DoS attack scheduling scheme using a DRL algorithm is summarized as Algorithm 1.

Algorithm 1 Optimal DoS attacks scheduling scheme

Initialize the replay memory $\mathcal{D}$ to capacity D;
Initialize the action-value function Q with random weights θ_0;
Initialize the target action-value function $\hat{Q}$ with weights $\theta_0^- = \theta_0$;
Initialize the state s_1;
for $k = 1, 2, \ldots, T$ **do**
  Take action a_k through exploration-exploitation

$$a_k = \begin{cases} \text{random} \quad a \in \mathcal{A}, & \text{w.p. } \epsilon, \\ \arg\max_a Q(s_k, a; \theta), & \text{w.p. } 1-\epsilon; \end{cases}$$

  Execute action a_k using $r_k = \sum_{i=1}^{N} \text{trace}(P_{i,k|k})$ to obtain reward r_k and observe s_{k+1};
  Update s_k;
  Store experience (s_k, a_k, r_k, s_{k+1}) into the replay memory $\mathcal{D}$;
  Uniformly draw a minibatch of experiences at random from the replay memory $\mathcal{D}$;
  Form the loss function using (22);
  Updata θ_k using (23);
  Every c steps update the target network $\theta_k^- = \theta_k$;
end for

4 Numerical example

In this section, an F-18 aircraft team example is provided to demonstrate the efficiency of the proposed optimal DoS attack scheduling scheme [39]. Considering the number of F-18 aircraft in the example is 6, which is modeled by

$$\begin{bmatrix} \dot{\alpha} \\ \dot{q} \end{bmatrix} = \begin{bmatrix} Z_\alpha & Z_q \\ M_\alpha & M_q \end{bmatrix} \begin{bmatrix} \alpha \\ q \end{bmatrix} + \begin{bmatrix} Z_{\delta E} & Z_{\delta \text{PTV}} \\ M_{\delta E} & M_{\delta \text{PTV}} \end{bmatrix} \begin{bmatrix} \delta_E \\ \delta_{\text{PTV}} \end{bmatrix} = A_s \begin{bmatrix} \alpha \\ q \end{bmatrix} + B_s \begin{bmatrix} \delta_E \\ \delta_{\text{PTV}} \end{bmatrix},$$

where α is the angle of attack, q is the pitch rate, Z_α, Z_q, M_α, and M_q are longitudinal stability derivatives, $Z_{\delta E}$, $Z_{\delta \text{PTV}}$, $M_{\delta E}$, and $M_{\delta \text{PTV}}$ are longitudinal control derivatives. Choose the following discrete dynamics of each F-18 aircraft based on the obtained data from wind tunnel and flight test data:

$$A_1 = \begin{bmatrix} 0.177 & 0.589 \\ -0.004 & 0.829 \end{bmatrix}, \ A_2 = \begin{bmatrix} 0.087 & -0.09 \\ 0.003 & 0.022 \end{bmatrix}, \ A_3 = \begin{bmatrix} 0.07 & 0.109 \\ -2.96 & 1.30 \end{bmatrix},$$
$$A_4 = \begin{bmatrix} -0.617 & 1.137 \\ -1.066 & 1.643 \end{bmatrix}, \ A_5 = \begin{bmatrix} 0.752 & -0.09 \\ 0.081 & 0.42 \end{bmatrix}, \ A_6 = \begin{bmatrix} -0.022 & 0.029 \\ -1.24 & 0.953 \end{bmatrix},$$
$$C_1 = [0.923 \ 0.293], \ C_2 = [0.367 \ 0.599], \ C_3 = [0.426 \ 0.549],$$
$$C_4 = [0.098 \ 0.039], \ C_5 = [0.659 \ 0.343], \ C_6 = [0.358 \ 0.892].$$

The positive semidefinite matrix Q_i and the positive matrix R_i are generated randomly. Consider the number of F-18 aircraft attacked by the attacker is 2.

In the DQN, two hidden layers, each of which has 1024 units, are used in the simulation. The rectified linear unit (ReLU) activation functions are used in the hidden layers, and a fully connected layer is used in the output layer. The discount factor is fixed at 0.97. The experience replay memory size D is set to 20000. The control horizon T is set to 500, which refers to an episode. The minibatch size is set to 32. The target network was updated every $c = 100$ iterations. The relationship between the combinations of attacked channels and the action is shown in Table 1.

The empirical average cost $\frac{1}{T}\sum_{k=1}^{\text{T}}\sum_{i=1}^{N}\text{trace}(P_{i,k|k})$ over the first 100 episodes is plotted in Figure 2. The action a_k over the first 5 episodes is plotted in Figure 3. Figure 2 shows the comparison between DRL

Table 1 The relationship between the combinations of attacked channels and the action

The combinations of attacked channels	Action
(1, 2)	0
(1, 3)	1
(1, 4)	2
(1, 5)	3
(1, 6)	4
(2, 3)	5
(2, 4)	6
(2, 5)	7
(2, 6)	8
(3, 4)	9
(3, 5)	10
(3, 6)	11
(4, 5)	12
(4, 6)	13
(5, 6)	14

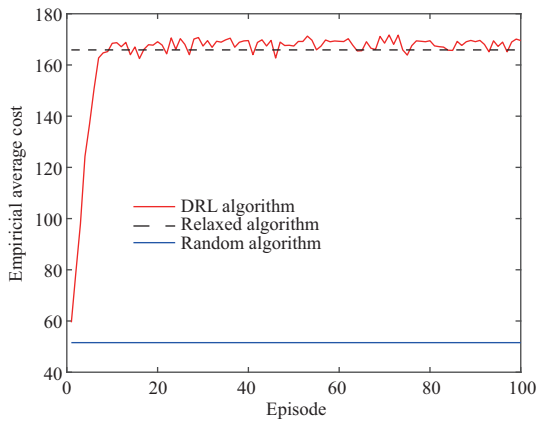

Figure 2 (Color online) Convergence of the empirical average cost.

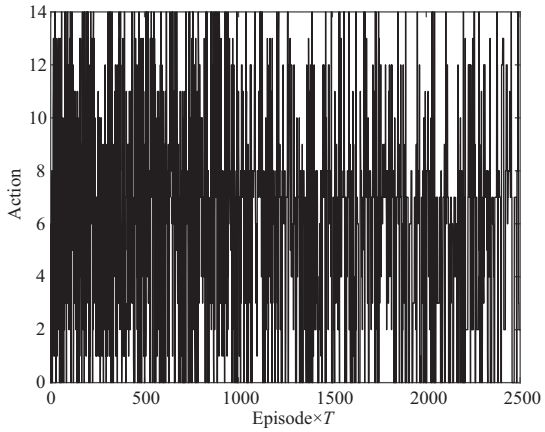

Figure 3 Attack scheduling strategy.

algorithm, convex relaxation algorithm in [33] and random algorithm for optimal DoS attacks scheduling. Convex relaxation algorithm in [33] depends on the initializations of attack variables. If the initializations were not selected well, there may be no feasible solutions, while a DRL algorithm is independent of the initializations of attack variables. Figure 3 shows the attack scheduling strategies made by the attacker. As we can see from Figure 2, the empirical average cost of DRL algorithm is higher than the empirical average cost of the convex relaxation algorithm, which means that the proposed DRL algorithm performs significantly better than the convex relaxation algorithm for the optimal DoS attack scheduling problem.

5 Conclusion

In this paper, we have examined the problem of optimal DoS attack scheduling schemes. Under limited computation, sensors only transmit measurements to the remote estimator via a communication channel that is exposed to DoS attackers. Due to the limited energy, a DoS attacker cannot attack all the sensors. Therefore, a DoS attacker must determine which sensors to attack in order to maximally degrade the estimation performance. A DRL algorithm has been introduced to solve the optimal DoS attack scheduling problem which is formulated as an MDP in an iterative model-free manner. The comparison in the simulation results demonstrated that the proposed DRL algorithm performs significantly better than the convex relaxation algorithm for the optimal DoS attack scheduling problem.

Acknowledgements This work was supported by National Natural Science Foundation of China (Grant Nos. U1613225, 61925303, 62088101, 61673023).

References

1 Wu G Y, Wang G, Sun J, et al. Optimal partial feedback attacks in cyber-physical power systems. IEEE Trans Autom Control, 2020, 65: 3919–3926
2 Pang Z H, Liu G P, Zhou D H, et al. Data-based predictive control for networked nonlinear systems with network-induced delay and packet dropout. IEEE Trans Ind Electron, 2016, 63: 1249–1257
3 Lu A Y, Yang G H. Input-to-state stabilizing control for cyber-physical systems with multiple transmission channels under denial of service. IEEE Trans Autom Control, 2018, 63: 1813–1820
4 Wang G, Giannakis G B, Chen J, et al. Distribution system state estimation: an overview of recent developments. Front Inf Technol Electron Eng, 2019, 20: 4–17
5 Wu C W, Hu Z R, Liu J X, et al. Secure estimation for cyber-physical systems via sliding mode. IEEE Trans Cybern, 2018, 48: 3420–3431
6 Pang Z H, Liu G P, Zhou D, et al. Two-channel false data injection attacks against output tracking control of networked systems. IEEE Trans Ind Electron, 2016, 63: 3242–3251
7 Chen J, Wang G, Sun J. Power scheduling for Kalman filtering over lossy wireless sensor networks. IET Control Theory Appl, 2016, 11: 531–540
8 Yang G, Zhou X S. Intelligent CPS: features and challenges. Sci China Inf Sci, 2016, 59: 050102
9 Gupta B, Brij B. Computer and Cyber Security: Principles, Algorithm, Applications, and Perspectives. Boca Raton: CRC Press, 2018
10 Vijayan J. Stuxnet renews power grid security concerns. Computerworld, 2010. http://www.computerworld.com/artical/2519574/security0/stuxnet-renews-power-grid-security-concerns.html
11 Teixeira A, Shames I, Sandberg H, et al. A secure control framework for resource-limited adversaries. Automatica, 2015, 51: 135–148
12 Kwon C, Liu W Y, Hwang I. Security analysis for cyber-physical systems against stealthy deception attacks. In: Proceedings of American Control Conference, Washington, 2013. 3344–3349
13 Mo Y, Chabukswar R, Sinopoli B. Detecting integrity attacks on SCADA systems. IEEE Trans Control Syst Technol, 2014, 22: 1396–1407
14 Guo Z Y, Shi D W, Johansson K H, et al. Worst-case stealthy innovation-based linear attack on remote state estimation. Automatica, 2018, 89: 117–124
15 Wu G Y, Sun J, Chen J. Optimal data injection attacks in cyber-physical systems. IEEE Trans Cybern, 2018, 48: 3302–3312
16 Mo Y, Sinopoli B. Secure control against replay attacks. In: Proceedings of Annual Allerton Conference, Illinois, 2009. 911–918
17 Teixeira A, Shames I, Sandberg H, et al. Revealing stealthy attacks in control systems. In: Proceedings of Annual Allerton Conference, Illinois, 2012. 1806–1813
18 Hou F Y, Sun J. Covert attacks against output tracking control of cyber-physical systems. In: Proceedings of the 43rd Annual Conference of the IEEE Industrial Electronics Society, Beijing, 2017. 5743–5748
19 Cárdenas A, Amin S, Sastry S. Research challenges for the security of control systems. In: Proceedings of Conference Hot Topics Security, California, 2018
20 Yang Y, Li Y F, Yue D. Event-trigger-based consensus secure control of linear multi-agent systems under DoS attacks over multiple transmission channels. Sci China Inf Sci, 2020, 63: 150208
21 Li W T, Wen C K, Chen J C, et al. Location identification of power line outages using PMU measurements with bad data. IEEE Trans Power Syst, 2016, 31: 3624–3635
22 Amin S, Cárdenas A, Sastry S. Safe and secure networked control systems under denial-of-service attacks. In: Proceedings of International Workshop on Hybrid Systems: Computation and Control, 2009. 31–45
23 Wang J Z, Shi L. Optimal DoS attacks on remote state estimation with a router. In: Proceedings of IEEE Conference on Decision and Control, Florida, 2018. 6384–6389
24 Chhabra M, Gupta B, Almomani A. A novel solution to handle DDOS attack in MANET. J Inform Secur, 2013, 4: 165–179
25 Bhushan K, Gupta B B. Distributed denial of service (DDoS) attack mitigation in software defined network (SDN)-based cloud computing environment. J Ambient Intell Human Comput, 2019, 10: 1985–1997
26 Bhardwaj A, Goundar S. Comparing single tier and three tier infrastructure designs against DDoS attacks. Int J Cloud Appl Comput, 2017, 7: 59–75
27 Almomani A, Alauthman M, Albalas F, et al. An online intrusion detection system to cloud computing based on NeuCube algorithms. Intern J Cloud App Comp, 2018, 8: 96–112
28 Zhang H, Cheng P, Shi L, et al. Optimal DoS attack scheduling in wireless networked control system. IEEE Trans Contr Syst Technol, 2016, 24: 843–852
29 Zhang H, Qi Y F, Wu J F, et al. DoS attack energy management against remote state estimation. IEEE Trans Control Netw Syst, 2018, 5: 383–394

30 Qin J H, Li M L, Wang J, et al. Optimal denial-of-service attack energy management against state estimation over an SINR-based network. Automatica, 2020, 119: 109090
31 Qin J H, Li M L, Shi L, et al. Optimal denial-of-service attack scheduling with energy constraint over packet-dropping networks. IEEE Trans Autom Control, 2018, 63: 1648–1663
32 Zhang H, Zhu Y Z, Li Z J, et al. DoS attack power allocation against remote state estimation via a block fading channel. In: Proceedings of Chinese Control Conference, Wuhan, 2018. 6441–6446
33 Yang C, Yang W, Shi H B. DoS attack in centralised sensor network against state estimation. IET Control Theory Appl, 2018, 12: 1244–1253
34 Mnih V, Kavukcuoglu K, Silver D, et al. Playing atari with deep reinforcement learning. 2013. ArXiv:1312.5602v1
35 Mnih V, Kavukcuoglu K, Silver D, et al. Human-level control through deep reinforcement learning. Nature, 2015, 518: 529–533
36 Yang Q L, Wang G, Sadeghi A, et al. Two-timescale voltage control in distribution grids using deep reinforcement learning. IEEE Trans Smart Grid, 2020, 11: 2313–2323
37 Leong A, Ramaswamy A, Duevedo D, et al. Deep reinforcement learning for wireless sensor scheduling in cyber-physical systems. Automatica, 2020, 113: 108759
38 Sadeghi A, Wang G, Giannakis G B. Deep reinforcement learning for adaptive caching in hierarchical content delivery networks. IEEE Trans Cogn Commun Netw, 2019, 5: 1024–1033
39 Lian J, Li C, Xia B. Sampled-data control of switched linear systems with application to an F-18 aircraft. IEEE Trans Ind Electron, 2017, 64: 1332–1340