A Multi-State Model of the Aging Process of Cyber-Physical Systems

Zhaojun Hao

Energy Department, Politecnico di Milano, Italy. E-mail: zhaojun.hao@polimi.it

Francesco Di Maio

Energy Department, Politecnico di Milano, Italy. E-mail: francesco.dimaio@polimi.it

Enrico Zio

Energy Department, Politecnico di Milano, Italy.
Centre de Recherche sur les Risques et les Crises (CRC), MINES ParisTech/PSL Université Paris, Sophia Antipolis, France
Department of Nuclear Engineering, Kyung Hee University, Seoul, South Korea. E-mail: enrico.zio@polimi.it

Cyber-Physical Systems (CPSs) are systems with tight integration and dependence between software and hardware parts, typically used as control systems and supervised by human operators. In this paper, we propose a multi-state model (commonly used to model the hardware degradation) to describe the aging of the cyber part of a CPS, where memory leakage is considered as the degradation process that leads to a service rate decrease, resulting in data jamming in the mission queue which, in turns, increases the memory request. The CPS is blocked and significantly increases the control delay when the amount of memory available cannot satisfy the demand of the mission queue. With control delay, the CPS may fail to control the system during transients: as an example, a typical Nuclear Power Plant (NPP) control rod system is taken as the CPS and its cyber degradation process modelled with the proposed multi-state degradation model. The unreliability of control rod system is quantified with respect to a transient of power step change, considering cyber degradation and hardware stochastic failure.

Keywords: Cyber-physical system, nuclear power plant, aging, multi-state model, memory leakage, data jamming, system blocking, control delay, control rod system.

1. Introduction

Cyber-Physical Systems (CPSs) are sytems with high integration and dependence between software and hardware parts, typically used as control systems and supervised by human operators (Baheti and Gill (2011)). CPSs are increasingly operated in energy production facilities (e.g. Nuclear Power Plants (NPPs)), transportation, medical, manufacturing applications, where high reliability of systems is strongly required, making the physical processes more and more dependent and interacting with the cyber process through feedback control loops (Lee et al. (2015)).

In all cases of applications, CPSs have to guarantee high reliability standards with respect to hardware components stochastic failures that can result in accidental scenarios leading the system towards unacceptable consequences, as well to cyber components failures (Wang et al. (2019)).

To target the requirement, dynamic reliability methods capable of capitalizing the simulation results of detailed dynamic models of CPSs that adequately represent components failure modes are being increasingly developed. The Dynamic Flowgraph Methodology (DFM) (Aldemir et al. (2006)), Petri Net (Lee et al. (2006)), Bayesian Network (Boudali and Dugan (2006)), Dynamic Fault Tree (DFT) (Dehlinger and Dugan (2008)), Fuzzy C-Means (FCM) (Di Maio et al. (2011)), Multi-State Physical Modeling (MSPM) (Di Maio et al. (2017)) have been used for NPP dynamic reliability assessment. In particular, MSPM is an efficient method to achieve quantitative reliability assessment results based on realistic degradation process assumptions, by integrating ”first principles” physical process dynamic models with hardware components degradation models. However, the focus is only on hardware stochastic failures, neglecting the contribution of the cyber components to CPSs failure due to degradation process such as memory leakage (Du et al. (2009)).

In this paper, we analyze the cyber components aging behavior and propose a multi-state model (commonly used to model the hardware degradation in MSPM) to describe the cyber aging, where memory leakage is considered as the degradation process that leads to a service rate decrease, resulting in data jamming in the mission queue which, in turns, increases the memory request. The CPS

Proceedings of the 30th European Safety and Reliability Conference and
the 15th Probabilistic Safety Assessment and Management Conference
Edited by Piero Baraldi, Francesco Di Maio and Enrico Zio
Copyright © ESREL2020-PSAM15 Organizers.*Published by* Research Publishing, Singapore.
ISBN: 978-981-14-8593-0; doi:10.3850/978-981-14-8593-0

is blocked and significantly increases the control delay when the amount of memory available cannot satisfy the demand of the mission queue. With control delay, the CPSs may fail to control the system during transients: as an example, a typical Nuclear Power Plant (NPP) control rod system is taken as the CPS and its cyber degradation process modelled with the proposed multi-state degradation model. The unreliability of control rod system is quantified with respect to a transient of power step change, considering cyber degradation and hardware stochastic failure.

The remaining of paper is as follows: Section 2 presents the cyber multi-state model; Section 3 presents the case study; in Section 4, conclusions are drawn.

2. Aging of Cyber Systems

CPSs normally consist of sensors, controllers, actuators, plant and communication network. A typical model of CPS with delay is shown in Fig. 1: the k-th sensor measurements y_k is taken at time t_k with a sampling interval $h[ms]$ and elaborated in $\tau_s[ms]$ before being transmitted through the network to the controller which processes the information and elaborates the action u_k to be done by the actuator in $\tau_c[ms]$, that reacts after $\tau_a[ms]$. Also, $\tau_{sc}[ms]$ and $\tau_{ca}[ms]$ are transmission delay through the network. The actuator holds the actuation signal output $u^*(t)$ as long as a new control command comes in.

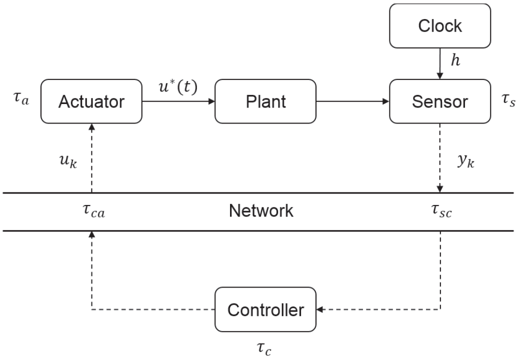

Fig. 1. Cyber-physical system with delay.

Aging of cyber systems manifests in performance degradation and failure rate increase of the software that drives the controller. Software aging is caused by some specific software faults/bugs, known as aging-related bugs (Grottke et al. (2008)) and activated by internal/external factors, causing errors that accumulate and propagate inside the system and finally lead to aging-related failures.

Memory leakage is an example of cyber aging process caused by internal errors, like unterminated processes that shrink the available physical memory amount (Grottke et al. (2008)).

With memory leakage, data jamming can occur due to the decreasing service rate, preventing the controller to process or deliver data and tasks in due time, which results in an accumulation of data in mission queue and an increase of the memory requests, and breaks the data packets flow (i.e, data packets loss) when the mission queue is full.

As a result, the system is blocked when the amount of memory available cannot satisfy the demand of the mission queue, significantly increasing the control delay (τ_c) in processing data of the controller (Cloosterman et al. (2009)) and reducing controllability and stability of the system (Åström and Wittenmark (2013)).

2.1. *Modeling the Aging of Cyber Systems*

2.1.1. *Memory leakage*

The system performance deteriorates stochastically and eventually reaches a blocking state when the available memory cannot satisfy the demand from the mission process queue; therefore, as shown in Fig. 2 (above), memory leakage degradation process can be modeled as a continuous time Markov Chain with state space $L = \{S_0, S_1, \ldots, S_n, B\}$, where state S_0 is the normal state, in which the system has the maximum memory capacity and performance; states $S_1 \sim S_n$ represent increasing degradation states referring to decreasing memory available; state B the blocking state; $\lambda_{i,i+1}\,(i = 0, 1, \ldots, n-1)$ the transition rates between degradation states S_i and S_{i+1}; $\lambda_{i,B}$ the system blocking transition rate from the i-th state S_i to blocking state B (if $i < j$, then $\lambda_{i,B} < \lambda_{j,B}$, which means that the worse the degradation state, the larger the transition rate to the blocking state).

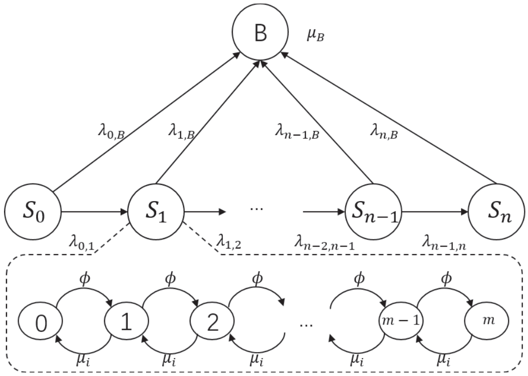

Fig. 2. Multiple degradation states of cyber aging.

2.1.2. *Data jamming*

For each degradation state S_i, assuming a data coming rate ϕ, an exponential service rate μ_i and a maximum capacity of task delivery queue equal to m, the continuous time Markov Chain of Fig. 2 (below) can be used to model data jamming, nested into the model of Fig. 2 (above), where μ_i denotes the different service rate under the different state $S_i\,(i = 0, 1, \ldots, k)$ (if $i < j$, then $\mu_i > \mu_j$), and the lowest service rate μ_B at blocking state B.

For each state S_i, the probability $P_{jam}(i, j)$ of j data jamming in the queue at state S_i is (Bolch et al. (2006)):

$$P_{jam}(i,j) = \frac{1-\phi/\mu_i}{1-(\phi/\mu_i)^{m+1}}(\frac{\phi}{\mu_i})^j$$
$$i = 1, 2, \ldots, n \quad j = 0, 1, \ldots, m \tag{1}$$

2.1.3. *Calculation of the system blocking transition rate*

As mentioned in Section 2.1.1, the probability of system blocking $P_{i,B}$ in state S_i, and, therefore, the blocking transition rate $\lambda_{i,B}$ depend on the current available memory $M(t)$ and on the memory request of the mission queue. $M(t)$ is estimated by assuming the transition time between degradation states (S_i and S_{i+1}) to be exponentially distributed with parameter $\lambda_{i,i+1}$ (Du et al. (2009)); with a Monte Carlo simulation that samples the transition time between S_i and S_{i+1} from S_0, the available memory corresponding to each state is recorded at each transition time; repeating N_{mc} times the simulation, the mean value of the collected available memory at each time is taken as the available memory $M(t)$ at time t. On the other hand, to estimate the memory request of the mission queue, we need to assume that each new data comes into the queue (with maximum capacity m) with a memory request which is a continuous random variable with density function $g(x)$ (Bao et al. (2005)): for any $0 < j \leqslant m$, let $g^{[j]}(x)$ be the density function for the total amount of j independent resource requests, is equal to the j-fold convolution of g (Trivedi (2001)).

$$\begin{aligned} &g^{[1]}(x) = g(x), \\ &g^{[j+1]}(x) = \int_0^x g^{[j]}(u)g(x-u)du, j \geqslant 1 \end{aligned} \tag{2}$$

Let $G^{[j]}(x)$ be the corresponding cumulative distribution funcion of $g^{[j]}(x)$, $G^{[j]}(x) = \int_0^x g^{[j]}(u)du$. The conditional probability $\xi[j, M]$ that the system blocks with j data in the queue and M memory available upon the arrival of a new request can be calculated considering the system blocking mechnism (i.e., the memory available cannot satisfy the memory request).

$$\begin{aligned} &\xi[0, M] = 1 - G^{[1]}(M), \\ &\xi[j, M] = 1 - \frac{G^{[j+1]}(M)}{G^{[j]}(M)}, 1 \leqslant j \leqslant m-1 \\ &\xi[m, M] = 1 - G^{[m]}(M) \end{aligned} \tag{3}$$

Combining the probability $P_{jam}(i, j)$ of j data jamming in the queue at state S_i shown in Section 2.1.2 and the conditional probability $\xi[j, M]$ of system blocking with j data, the probability $P_{i,B}(M)$ of system blocking at each state with M available memory can be calculated as in Eq. (4). Reminding that $M(t)$ is specific to each system, we can calculate $P_{i,B}(t)$ and the corresponding blocking transition rate $\lambda_{i,B}(t)$ as in Eq. (5) and (6).

$$P_{i,B}(M) = \sum_{j=0}^{m} P_{jam}(i,j) \cdot \xi[j, M]$$
$$i = 1, 2, \ldots, n \tag{4}$$

$$f_{i,B}(t) = \frac{dP_{i,B}(t)}{dt} \tag{5}$$

$$\lambda_{i,B}(t) = \frac{f_{i,B}(t)}{1 - P_{i,B}(t)} \tag{6}$$

2.1.4. *Calculation of the control delay*

The transmission delay τ_{sc} and τ_{ca} of Fig. 1 are usually assumed constant for a specific network structure, while τ_c is dependent with the system state (blocking or non-blocking), and consists in the waiting time $\tau_{waiting}$ necessary for a data packet to be processed, and in the calculating time $\tau_{calculating}$. When the cyber system is in a non-blocking state $S_0 \sim S_n$, the control delay equals to the sum of τ_{sc} and τ_{ca} (τ_c can be neglected); whereas, when the cyber system is in the blocking state B, the low service rate causes data packets accumulation in the mission queue, significantly increasing τ_c, which results in an increase of the total control delay τ.

$$\tau_c = \tau_{waiting} + \tau_{calculating} \tag{7}$$

$$\tau = \begin{cases} \tau_{sc} + \tau_{ca}, & S_0 \sim S_n \\ \tau_{sc} + \tau_{ca} + \tau_c, & B \end{cases} \tag{8}$$

The signal delay calculation in a degraded CPS is sketched in Fig. 3: the sensors sample the signals from the plant at the k-th sampling time; the control command signal finally reaches the actuator after delay τ: when the cyber system is in a non-blocking state $S_0 \sim S_n$, the total delay τ only accounts for τ_{sc} and τ_{ca}, commonly less than h; whereas, when the cyber system is in the blocking state B, τ also accounts for τ_c, making τ larger than h.

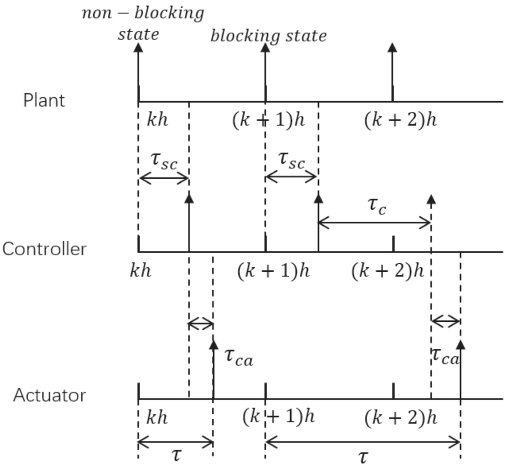

Fig. 3. Delay of the CPS control.

3. Case Study: The Control Rod System

With the aging model proposed in Section 2, the unreliability of a CPS considering cyber aging can be analyzed: to show this, a typical NPP control rod system is taken as a case study, the cyber degradation process described and the multi-state model built.

3.1. *The Control Rod System*

The control rod system (Fig. 4) is an important safety CPS employed in NPPs to adjust the position of control rods in the reactor core to control the thermal power and the electric power generated (Divandari et al. (2014); Yoritsune et al. (2002); Yuanqiang et al. (2002); Bakhri (2016)), with a closed-loop feedback control (Fig. 5), including a sensor, a controller, a connecting network, an actuator and a DC motor for rods movement, where r_k, u_k, y_k, e_k are the reference, control, output and error signals, respectively, at time $t_k = kh$ (Tipsuwan and Chow (2003)). τ is the total delay time of the feedback control loop, including the network transmission (τ_{sc}, τ_{ca}) and the controller processing (τ_c) delay time.

Without loss of generality, i) a typical digital Instrumental & Control (I&C) platform is chosen as hardware for the controller module in single controller mode with one CPU and, ii) the motor is a typical DC motor, used to control the position of the control rods (Divandari et al. (2015)). The transfer function of the controller and DC motor in Fig. 5 in Laplace domain is described given in Eq.(9) and (10) (Tipsuwan and Chow (2003)):

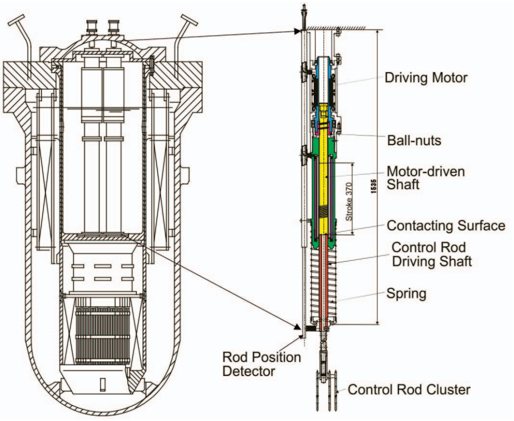

Fig. 4. Structure of control rod system.

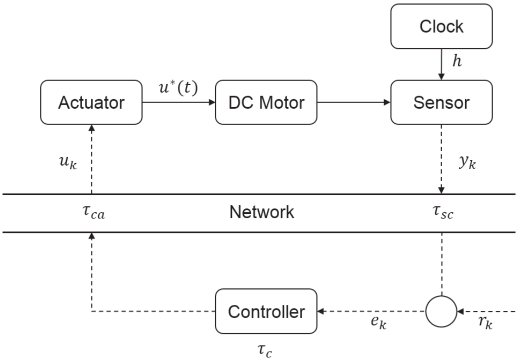

Fig. 5. Closed-loop control system of case study.

$$G_C(s) = \frac{\beta K_P(s + K_I/K_P)}{s}$$
$$K_P = 0.1701, \quad K_I = 0.378 \tag{9}$$

$$G_P(s) = \frac{2029.826}{(s + 26.29)(s + 2.296)} \tag{10}$$

For the convenience of simulation, we change the transfer function Eq.(9) and (10) into discrete form as shown below:

$$\begin{aligned} y_n = 1.944y_{n-1} - 0.944y_{n-2} \\ + 0.004u_{n-1} + 0.0039u_{n-2} \end{aligned} \tag{11}$$

$$u_n = u_{n-1} + 0.17e_n - 0.163e_{n-1} \tag{12}$$

3.2. *The Cyber Aging Model for the Control Rod System*

We assume the degradation model described in Section 2: with $n = 3$ degradation states ($S_1 \sim S_3$); memory available of $100, 75, 50, 25 Kbytes$ for S_0, S_1, S_2 and S_3, respectively; the transition rate $\lambda_{i,i+1}$ between degradation states is $2E-4/h$ (Butenko et al. (2014)).

We use Monte Carlo simulation to calculate the decreasing available memory curve $M(t)$: i) start from S_0 and sample the transition time between S_i and S_{i+1} from exponential distribution with transition rate $\lambda_{i,i+1}$ until the system blocks; ii) record the memory $(100, 75, 50, 25 Kbytes)$ of each state $(S_0 \sim S_3)$ corresponding to the transition time; iii) repeat procedures above N_{mc} times and calculate the mean value of the memory at each time to draw the memory decreasing curve $M(t)$. One random trial of simulation process (dash line) and the resulting available memory decreasing curve $M(t)$ (continuous line) are shown in Fig. 6.

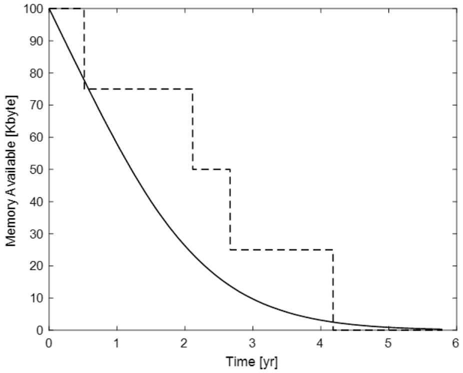

Fig. 6. Decreasing available memory $M(t)$ (continuous line).

As mentioned in Section 2.1.3, to calculate the system blocking transition rate $\lambda_B(t)$, we assume:

(i) the density function of each memory request to be uniform $g \sim U[2, 7]$ when the unit of the memory request is Kbyte.
(ii) the density function of total amount of $m = 10$ independent resource requests $g^{[1]} \sim g^{[10]}$ and the corresponding cumulative distribution fuction $G^{[1]} \sim G^{[10]}$ as calculated in Eq.(2).
(iii) the conditional probability $\xi[j, M]$ as calculated in Eq.(3).
(iv) the probability $P_{jam}(i, j)$ of jamming j data in the queue at degradation state S_i as calculated in Eq.(1), with service rate μ_i and the data coming rate ϕ shown in Table 1.
(v) the service rate at blocking state μ_B equal to the value in Table 1.

Table 1. Parameters for data jamming model

Parameter	Comment	Value
m	Maximum number of tasks	10
ϕ	Data coming rate	$50(s^{-1})$
μ_0	Service rate in state S_0	$100(s^{-1})$
μ_1	Service rate in state S_1	$85(s^{-1})$
μ_2	Service rate in state S_2	$70(s^{-1})$
μ_3	Service rate in state S_3	$55(s^{-1})$
μ_B	Service rate in state S_4	$30(s^{-1})$

Combining the probability $P_{jam}(i, j)$ of j data jamming in the queue at state S_i and the conditional probability $\xi[j, M]$ that the system blocks with j data, the resulting probability $P_{i,B}(M)$ of system blocking in each state is calculated with Eq. (4) and shown in Fig. 7;

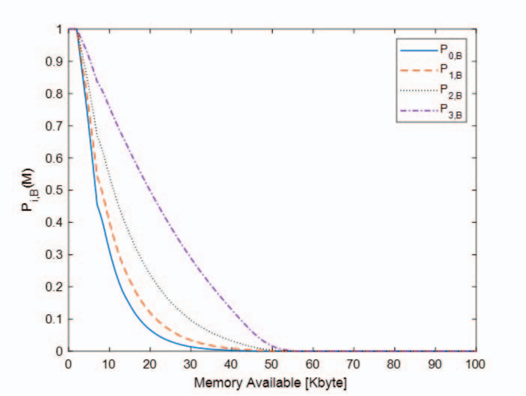

Fig. 7. Probability of system blocking in each state.

Since the memory curve $M(t)$ (Fig. 6) shows that the system sojourns in state S_0 for $0.58yr$, S_1 for $0.63yr$, S_2 for $0.84yr$ and S_3 for $2.3yr$, we can derive the system blocking transition rate $\lambda_B(t)$ as plotted in Fig. 8 by assigning, to each of the sojourn time of each state, the corresponding $\lambda_{i,B}(t)$ as calculated with Eq.(5) and (6), on the basis of the results shown in Fig. 7.

According to Section 2.1.4, when the system is in a blocking state, the total delay time τ consists of two parts: the transmission delay (τ_{sc}, τ_{ca}) and the controller processing delay (τ_c); whereas, when the system is in the non-blocking state, only the transmission delay is accounted for. Here we assume 1) the transmission delay obeys to $N(13.0985, 5.7005)[ms]$ (Cheng et al. (2007)).

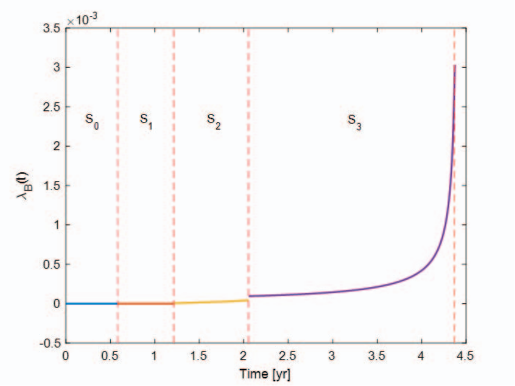

Fig. 8. Transition rate of system blocking.

2) the controller processing delay of a new coming data consists two parts: i) $\tau_{waiting}$ which is the sum of processing delay of the former data that still processing in the queue. ii) $\tau_{calculating}$ sampled from exponential distribution with corresponding service rate of current state in Table 1.

3.3. *Reliability Analysis of the Control Rod System*

With the model and parameters described in Section 3.2, in this Section we show how to calculate the reliability of the rod control system described in Section 3.1 when attempting to control, for example, a power step change. This transient is here chosen only for demonstration purpose of the methodology: it is worth pointing out that, since the NPP can experience a very large spectrum of transients during its life time, the result cannot be considered exhaustive and realistic, because an in-depth analysis of the response of the aging CPS to all the types of transients should be done to draw robust conclusions on its reliability when accounting for aging.

For the power step change, the control rod system is considered failed when the transient overshoot is out of the control safety boundary (assumed to be 20% above and below the reference value). The system response to a power step change considering cyber aging is simulated by the procedure, whose pseudocode is given in Fig. 9, with parameters in Table. 1. Fig. 10 shows an example of 10 simulations.

For comparison, we consider the reliability assessment of the same typical control rod system when only hardware stochastic failures, cyber aging and both two factors are accounted for, respectively. In particular, only the controller and DC motor stochastic failures are considered whose failure rates are listed in Table 2 (Chyou et al. (2004)) while the system blocking transition rate of cyber aging is calculated in Section 3.2.

```
1:  Initialization: Mission time $T_{miss} = 10s$, simulation step $dt = 0.002s$, sample interval $h = 0.02s$, sample time $t_k = kh, k = 0, 1, \ldots$
2:  Set: $t = 0s$ and the initial value of system output, reference, error and control signals: $y_0 = 0, r_0 = 1, e_0 = 1, u_0 = 0$
3:  $t = t + dt$
4:  Calculate $y_n$ according to Eq. (11)
5:  **if** $t = t_k$ (sensors sample new data $y_k$) **then**
6:      Sample delay $\tau$ according to Eq. (7) and (8)
7:  **end if**
8:  **if** $t > t_k + \tau$ **then**
9:      $e_n = r_n - y_k$
10:     Calculate $u_n$ according to Eq. (12)
11: **else**
12:     $e_n = e_{n-1}, u_n = u_{n-1}$
13: **end if**
14: **if** $|y_n - r_n| > 0.2$ **then**
15:     The system is out of control
16: **end if**
17: Repeat (3)~(13) until $t = T_{miss}$
```

Fig. 9. Pseudocode of power step change simulation.

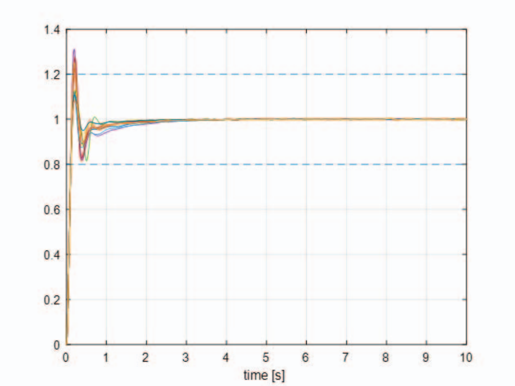

Fig. 10. Result of 10 simulations with power step change.

Table 2. Parameters for hardware stochastic failure

Parameter	Comment	Value$[h^{-1}]$
$\lambda_{controller}$	Controller failure rate	$8.01E - 6$
λ_{motor}	DC motor failure rate	$9.50E - 6$

To do this, we rely on the Monte Carlo simulation whose pseudocode is given in Fig. 11. According to Section 2, the system being out of control only happens when system is blocked. Therefore, we run the simulation described in Fig. 9 at each time step dt as long as the system is blocked, and record the system failure time (due to hardware stochastic failure or cyber aging caused being out of control). Finally by calculating, at each time, the mean value of simulations that have led the system out of control for N_c simulations, the system unreliability is estimated.

```
1:  Calculate the transition rate $\lambda_B$ as described in Section 2.1.3
2:  Initialization: Mission time $T_{miss} = 50000h$, simulation step $dt = 1h$
3:  **for** $i = 1 : N_c$ **do**
4:      Set $t = 0$
5:      Sample stochastic failure time $T_S$ using the parameters in Table. 2
6:      $t = t + dt$
7:      **if** $t > T_S$ (stochastic failure occurs) **then**
8:          $Ca(t : T_{miss}) = Ca(t : T_{miss}) + 1$
9:          Jump to (25)
10:     **else**
11:         **if** (system is not blocked) **then**
12:             Sample blocking transition time $T_B$ with blocking transition rate $\lambda_B(t)$
13:             **if** $T_B < dt$ **then**
14:                 system jumps to state $B$
15:             **end if**
16:         **end if**
17:         **if** (system is blocked) **then**
18:             Simulate system step power change with control delay using the pseudocode in Fig. 9
19:             **if** (system out of control) **then**
20:                 $Ca(t : T_{miss}) = Ca(t : T_{miss}) + 1$
21:                 Jump to (25)
22:             **end if**
23:         **end if**
24:     **end if**
25:     Repeat (6)$\sim$(24) until $t = T_{miss}$
26: **end for**
27: Calculate system unreliability during mission time: $Ca/N_c$
    * if hardware stochastic failure is not considered, skip step $5, 7 \sim 10$.
```

Fig. 11. Pseudocode of system unreliability considering cyber aging and stochastic failure.

The simulation result is shown in Fig. 12. The three curves are system unreliability of power step change considering stochastic failure (continuous line), cyber aging (dashed-dotted line) and both stochastic failure and cyber aging (dotted line): in the first year, only the stochastic failures would make the system accidentally fail; then, with the cyber aging gradually deteriorating the control rod system controllability, it starts contributing to the system unreliability, finally becoming the main contributor for the system failure.

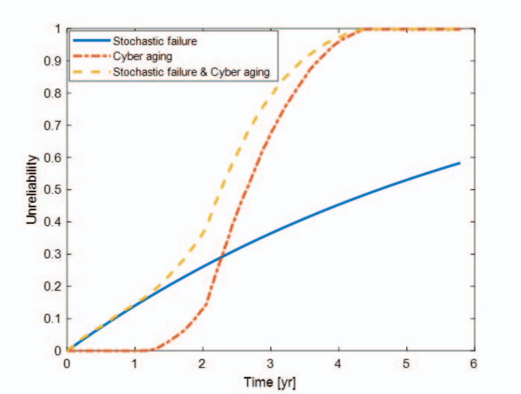

Fig. 12. Reliability simulation result.

4. Conclusion

In this paper, a multi-state degradation model that is commonly used to model the hardware degradation is proposed to describe the cyber degradation. Memory leakage is considered as the main degradation process while data jamming and control delay are the phenomena of aging and blocking. A Markov Chain model is used to describe the memory leakage process with different states and performance capabilities (service rate). A queueing model is used as the work load and embedded into each state to describe data jamming. When system blocks, the control delay significantly increases and damages the system stability and controllability which leads the system being out of control.

The feedback control system consists of a PI controller, a DC motor and connecting network, using for NPP control rod system is taken as the case study and the cyber aging model is built. The system power step change considering cyber aging and stochastic failure is simulated by Monte Carlo simulation. The result of simulation shows that in the first year, only the stochastic failures would make the system accidentally fail. On the other hand, as long as the cyber aging deteriorates the control rod system controllability, it exposes the system to hazard, finally becoming the main contributor for the system failure, that should not be neglected in practice.

References

Aldemir, T., D. Miller, M. Stovsky, J. Kirschenbaum, P. Bucci, A. Fentiman, and L. Mangan (2006). Current state of reliability modeling methodologies for digital systems and their acceptance criteria for nuclear power plant assessments. *US Nuclear Regulatory Commission, Washington, DC, Report No. NUREG/CR-6901*.

Åström, K. J. and B. Wittenmark (2013). *Computer-controlled systems: theory and design*. Courier Corporation.

Baheti, R. and H. Gill (2011). Cyber-physical systems. *The impact of control technology 12*(1), 161–166.

Bakhri, S. (2016). Investigation of rod control system reliability of pwr reactors. *KnE Energy*, 94–105.

Bao, Y., X. Sun, and K. S. Trivedi (2005). A workload-based analysis of software aging, and rejuvenation. *IEEE Transactions on Reliability 54*(3), 541–548.

Bolch, G., S. Greiner, H. De Meer, and K. S. Trivedi (2006). *Queueing networks and Markov chains: modeling and performance evaluation with computer science applications*. John Wiley & Sons.

Boudali, H. and J. B. Dugan (2006). A continuous-time bayesian network reliability modeling, and analysis framework. *IEEE transactions on reliability 55*(1), 86–97.

Butenko, V., V. Kharchenko, O. Odarushchenko, P. Popov, V. Sklyar, and E. Odarushchenko (2014). Markov's model and tool-based assessment of safety-critical i&c systems: Gaps of the iec 61508. *Proceedings of Probabilistic Safety Assessment and Management PSAM 12*.

Cheng, C.-W., C.-L. Lai, B.-C. Wang, and P.-L. Hsu (2007). The time-delay effect of multiple-network systems in ncs. In *SICE Annual Conference 2007*, pp. 929–934. IEEE.

Chyou, Y.-P., D.-D. Yu, and Y.-N. Cheng (2004). Performance validation on the prototype of control rod driving mechanism for the trr-ii project. *Nuclear engineering and design 227*(2), 195–207.

Cloosterman, M. B., N. Van de Wouw, W. Heemels, and H. Nijmeijer (2009). Stability of networked control systems with uncertain time-varying delays. *IEEE Transactions on Automatic Control 54*(7), 1575–1580.

Dehlinger, J. and J. B. Dugan (2008). Analyzing dynamic fault trees derived from model-based system architectures. *Nuclear Engineering and Technology 40*(5), 365–374.

Di Maio, F., D. Colli, E. Zio, L. Tao, J. Tong, et al. (2017). A multi-state physics modeling for estimating the size-and location-dependent loss of coolant accident initiating event probability. In *2017 International Topical Meeting on Probabilistic Safety Assessment and Analysis, PSA 2017*, Volume 2, pp. 1185–1192. American Nuclear Society.

Di Maio, F., P. Secchi, S. Vantini, and E. Zio (2011). Fuzzy c-means clustering of signal functional principal components for post-processing dynamic scenarios of a nuclear power plant digital instrumentation and control system. *IEEE Transactions on Reliability 60*(2), 415–425.

Divandari, M., M. Hashemi-Tilehnoee, B. Asgari-Ziarati, M. Hosseinkhah, and K. Sabagh (2015). Minimizing torque ripple in a brushless dc motor with fuzzy logic: applied to control rod driving mechanism of mnsr. *Nuclear science and techniques 26*(1), 10601–010601.

Divandari, M., M. Hashemi-Tilehnoee, M. Khaleghi, and M. Hosseinkhah (2014). A novel control-rod drive mechanism via electromagnetic levitation in mnsr. *Nukleonika 59*(2), 73–79.

Du, X., Y. Qi, D. Hou, Y. Chen, and X. Zhong (2009). Modeling and performance analysis of software rejuvenation policies for multiple degradation systems. In *2009 33rd Annual IEEE International Computer Software and Applications Conference*, Volume 1, pp. 240–245. IEEE.

Grottke, M., R. Matias, and K. S. Trivedi (2008). The fundamentals of software aging. In *2008 IEEE International Conference on Software Reliability Engineering Workshops (ISSRE Wksp)*, pp. 1–6. Ieee.

Lee, D.-Y., J.-G. Choi, and J. Lyou (2006). A safety assessment methodology for a digital reactor protection system. *International Journal of Control, Automation, and Systems 4*(1), 105–112.

Lee, J., B. Bagheri, and H.-A. Kao (2015). A cyber-physical systems architecture for industry 4.0-based manufacturing systems. *Manufacturing letters 3*, 18–23.

Tipsuwan, Y. and M.-Y. Chow (2003). Control methodologies in networked control systems. *Control engineering practice 11*(10), 1099–1111.

Trivedi, K. (2001). *Probability and Statistics with Reliability, Queuing and Computer Science Applications*. John Wiley & Sons.

Wang, W., F. Di Maio, and E. Zio (2019). Adversarial risk analysis to allocate optimal defense resources for protecting cyber–physical systems from cyber attacks. *Risk Analysis*.

Yoritsune, T., T. Ishida, and S. Imayoshi (2002). In-vessel type control rod drive mechanism using magnetic force latching for a very small reactor. *Journal of nuclear science and technology 39*(8), 913–922.

Yuanqiang, W., D. Xingzhong, Z. Huizhong, and H. Zhiyong (2002). Design and tests for the htr-10 control rod system. *Nuclear Engineering and design 218*(1-3), 147–154.