An Emergency Management System for Government Data Security Based on Artificial Intelligence

Haining Luo

State Information Center, Beijing 100045, China

Corresponding Author Email: lhn@cegn.gov.cn

ABSTRACT

This paper designs a novel emergency management system for government data security based on artificial intelligence (AI). Firstly, a novel concept was put forward government data resilience chain (GDRC) based on the AI technology. Next, several AI techniques were designed to ensure the security of government data on data sharing and exchange platforms, including intelligent government data security monitoring technology, artificial neural network (ANN) and fuzzy identification technology, and firewall security technology based fuzzy clustering. On this basis, an emergency management system was established for government data security, and verified through simulation of fault tolerance of multi-agent formation. The results show that the GDRC helps solve emergency management problems in the sharing and exchange of government data; the proposed system realized deep learning (DL), and real-time updates of dynamic features of multiple agents. The research results provide effective decision supports for improving the fault tolerance of multi-agent system.

1. INTRODUCTION

In recent years, many countries have established data sharing and exchange platforms, which meet the data demand of the government and the public. However, the government data are not sufficiently open or standardized [1]. There is a severe lacking of disclosure standards and regulatory mechanisms for data on national emergency management. To make matters worse, the government's preparation and utilization of open data have not been verified, in the light of the key issues in emergency management.

Artificial intelligence (AI) offers a possible solution to the above problem. Currently, the AI techniques combine the merits of deep learning (DL) algorithms, and the information processing strategies of big data [2-6]. Machine learning (ML), a subset of the DL, makes full use of the superior computing accuracy and data processing capabilities of machines. Many scholars have explored emergency management, using ML methods for information flow, edge computing, secure delegation of private data, etc. [7-10].

This paper attempts to ensure the security of data sharing and exchange platforms in the area of emergency management. For this purpose, a novel concept called government data resilience chain (GDRC) was put forward based on the AI techniques for government data security. Based on the GDRC, an emergency management system was established to ensure the security of government data. The application in multi-agent formation proves that our system guarantees the robustness of the business at the location of government data, and safeguards the sustainable and smooth operation of important business platforms.

2. SECURITY MODEL

Drawing on traditional data security models, this section constructs an AI-based security model for government data in emergency management systems. The purpose of the model is to enhance the reliability of the decision supports to emergency response [11]. The mainstay of the security model is the GDRC, which consists of four key aspects: continuous monitoring, efficient response, fast recovery, and repeated verification. The structure of the GDRC-centered security model is illustrated in Figure 1 below.

Specifically, continuous monitoring, the core of the GDRC, fully utilizes DL and other AI techniques to extract the features of various existing or potential security threats, classifies the extracted features, and combines the classification results into a feature library. Efficient response refers to the efficient research and actions in response to the contents of continuous monitoring. Fast recovery protects and restores government data based on the responses. Repeated verification relies on techniques, personnel, and processes to verify the latest system state on a continuous basis, offering positive feedbacks for continuous monitoring.
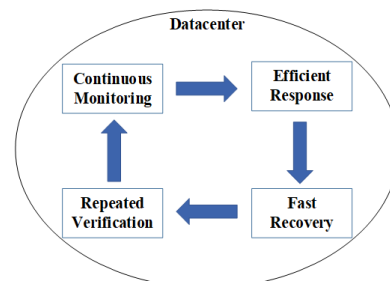
Figure 1. The GDRC-centered security model

3. AI-BASED SECURITY TECHNIQUES

In the area of government data security, the main application value of the AI lies in its strong ability to gather information resources, analyze and process fuzzy information, make effective learning and reasoning, as well as cooperate with multiple agents. Thanks to these advantages, the AI boasts strong applicable value in protecting the security of government data. This section brief introduces the AI techniques adopted to ensure government data security.

3.1 Government data security monitoring technology

Based on AI techniques like the DL and big data, government departments at all levels can establish big data security centers to collect, organize, and analyze various types of security events, and then build a data security detection and response platform covering such dimensions as region, industry, and type. The multi-dimensional platform will provide each datacenter with network security services from the cloud. With the aid of AI techniques, the government departments will be able to respond quickly to unexcepted security incidents, e.g. fast-tracking incident analysis, alert release, and information processing. The security strategy of the multi-dimensional platform will be updated synchronously, creating a complete and efficient data security system.

3.2 Artificial neural network (ANN) and fuzzy recognition technology

The ANN is a highly self-adaptive network capable of identifying noises or distortions. Mimicking the information interaction principle of the biological neural network, the ANN acquires the learning and comprehension abilities to quickly identify and store various information. Based on the ANN, a time series prediction model can be established to improve the detection efficiency of intruding viruses. In addition, the fuzzy recognition technology, also an AI technique, can be introduced to screen and pinpoint viruses rapidly in each datacenter, allowing the government to determine the type of viruses and adopt the correct security policies for the computer network.

3.3 Firewall security technology based on fuzzy clustering

The firewall security is an AI technology with advanced intelligence and decision-making capabilities. This technique helps to further enhance the robustness of network security. There are multiple sub-techniques of the firewall security, including but not limited to fuzzy clustering, packet filtering, and state monitoring. Among them, fuzzy clustering can predict unknown risks based on the current feature library, and detect the intruding viruses in an accurate manner. Packet filtering mainly performs nonlinear monitoring of data packets n the network layer, and integrates the destination address and source information.

4. SYSTEM DESIGN BASED ON MULTI-AGENT FORMATION

In the era of the big data, it is of great importance for government departments to ensure the security of data sharing and exchange platforms. From the perspective of multi-agent

collaboration, this paper builds up an emergency management system for government data security. In this system, each agent represents a datacenter. Dividing the datacenters into masters and slaves, the following analysis only focuses on the agents of master datacenters.

For our problem, three agents are sufficiently representative, as the situation of additional agents can be determined by analogy. To ensure the security of datacenters, it is critical to find the formation of agents that maintain the fault tolerance and persistence of the emergency management system [12,13]. Here, the fault tolerance is only considered under the situation where only one connection of agent is cut off.

4.1 Preliminaries for graph persistence

The previous studies have shown that formation can fulfil various tasks in different ways, and the formation should be controlled by directed graph rather than undirected graph [14-17]. Hence, it is necessary to express the underlying rigid formations as an undirected graph.

Let $G = (V, E)$ be an undirected graph with n points, $p = (p_1, p_2, \dots p_n) \in R^{2n}$ as the composite vector, and (G, p) be a pair of points. Note that $p_i (i \in \{1, 2, \dots, n\})$ is the coordinates of the i-th point on the plane. Further, a constant parameter $d_{ij} = ||p_i - p_j||$ was introduced as the Euclidean distance between a pair of points (p_i, p_j). In addition, the rigidity matrix was defined as [18]:

$$\langle (p_i - p_j), (p_i - p_j) \rangle = d_{ij}^2 \quad i, j \in \{1, 2, \dots, n\} \tag{1}$$

Assuming that the trajectory is smooth, the following can be derived from formula (1):

$$2 \langle (p_i - p_j), (\dot{p}_i - \dot{p}_j) \rangle = 0,$$
$$i, j \in \{1, 2, \dots, n\}, t \geq 0 \tag{2}$$

where, $\dot{p}_i$ is the velocity at point p_i. The following homogenous function can be obtained by combining formula (2) at different points:

$$R \dot{p} = 0 \tag{3}$$

where, $\dot{p} = column\{\dot{p}_1, \dot{p}_2, \dots \dot{p}_n\}$; R is a rigidity matrix with dimension $m \times 2n [m = n(n-1)/2]$ on the plane.

By another way of definition, the rigidity function can also be described as follows:

$$\hat{g}_G(p) = [|| p_{e_1}^{in} - p_{e_1}^{out} ||^2, || p_{e_2}^{in} - p_{e_2}^{out} ||^2, \dots || p_{e_m}^{in} - p_{e_m}^{out} ||^2]^T \tag{4}$$

where, the i-th component $||p_{e_k}^{in} - p_{e_k}^{out}||^2$ corresponds to the edge e_k of E ($k \in \{1, 2, \dots, m\}$ $E = \{e_1, e_2, \dots e_m\}$).

Definition 4.1 [19, 20] A framework (G, p) is *infinitesimally rigid* on the plane if $dim(\ker J_{\hat{g}_G}(p)) = 3$, or if

$$\text{rank} J_{\hat{g}_G}(p) = 2n - 3 \tag{5}$$

where, $J_{\hat{g}_G}(p)$ is the *Jacobian* matrix of $\hat{g}_G(p)$ and takes the same form as the rigidity matrix R.

Rigidity is the property of the undirected graph, and persistence is the corresponding property of the directed graph.

Lemma 4.1 For a directed graph with more than one point, it is minimally persistent if and only if the underlying undirected graph is minimally rigid, and of no point has more than two outgoing edges [21].

It is known that the number of the points in minimally persistent graph is always $2n$-3, where n is the number of points.

4.2 Formation control laws

Figure 2 gives an example of the minimally persistent formation of three co-leaders (agents 4, 5, and 6). The three leaders in the cycle all have one degree of freedom (DOF). The DOF of followers (agents 1, 2, and 3) is zero. The followers are directly or indirectly controlled by the leaders through the distance constraint. Specifically, agent 1 is controlled by agents 4 and 6; agent 2 is controlled by agent 4 and follower 1.
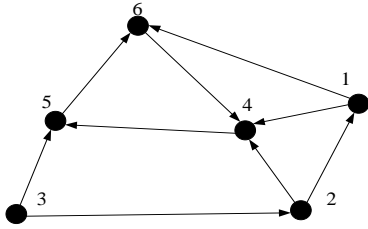
Figure 2. A minimally persistent formation of three co-leaders

In this paper, the first-order kinematic model is adopted for every agent:

$$u_i = \dot{p}_i \qquad (6)$$

Suppose the adjustment of leaders is so minor that the followers can catch up with the two neighboring leaders in due course. The formation motions were detailed as follows:

Before time 0, the agents are in the persistent statement that every distance and direction constraints are working, fulfilling the requirements on rigidity and constraint consistence. Then, one leader discovers the object, moves to it, and stops moving for a while. The other two leaders will quickly follow suit. After that, all followers will correct their positions based on the distance constraint with their neighboring leaders. Next, the first leader will move again. The above process finishes in a very short time, and will be repeated on and one. However, the control laws for leaders and followers are different, for the two types of agents follow diverse motion principles.

4.2.1 Control laws for leaders with fault tolerance

Suppose three leaders *n-2, n-1,* and *n* lead each other, and any of them responds to the formation assignment randomly. As shown in Figure 3, leaders 4, 5, and 6 are equivalent. It is assumed that leader 6 is the first to respond to the assignment. Thus, leaders 4 and 5 need to respond to leader 6. The dashed lines are specially designed spare lines (ineffective in normal motion) for the fault tolerance of the leaders in the cycle.
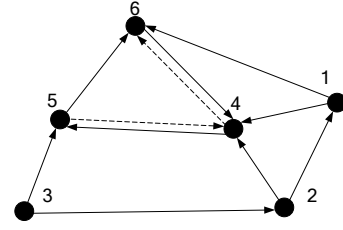

Figure 3. A minimally persistent formation of three co-leaders with fault tolerance

There are three different formations of co-leaders for fault tolerance:

(A) Disconnection at edge (6, 4)

First, it is assumed that the edge between leaders 6 and 4 is broken (Figure 4). As shown in Figure 5, when edge (6, 4) is cut off, leaders 4 and 5 still know the information from leader 6, and the added edge (4, 6) or (5, 4) does not take a part in the control laws.
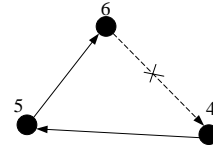
Figure 4. Disconnection at edge (6, 4)

(a) Edge (4, 6) is added   (b) Edge (5, 4) is added
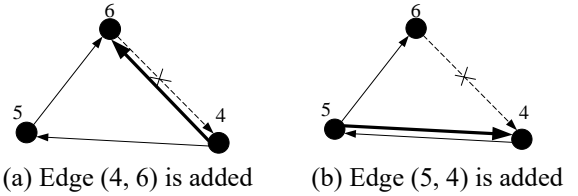
Figure 5. Type A fault tolerance

Hence, the detailed control laws can be expressed as:

$$\dot{p}_{n-2} = k_{n-2} \frac{\| p_{n-1} - p_{n-2} \| - d_{(n-2,n-1)\_o}}{\| p_{n-1} - p_{n-2} \|} (p_{n-1} - p_{n-2})$$

$$\dot{p}_{n-1} = k_{n-1} \frac{\| p_n - p_{n-1} \| - d_{(n-1,n)\_o}}{\| p_n - p_{n-1} \|} (p_n - p_{n-1}) \qquad (7)$$

where, k_n, k_{n-1}, and k_{n-2} are real gains; $d_{(n-2,n-1)\_o}, d_{(n-1,n)\_o}, d_{(n,n-2)\_o}$ are the specific distances.

Then, we have:

$$\begin{bmatrix} \delta \dot{x}_{n-2} \\ \delta \dot{y}_{n-2} \end{bmatrix} = k_{n-2} I_2 R_{e\_n-2} R_{((n-2)(n-1),00)} \begin{bmatrix} \delta x_{n-2} \\ \delta y_{n-2} \\ \delta x_{n-1} \\ \delta y_{n-1} \end{bmatrix} \qquad (8)$$

where,

$$R_{((n-2)(n-1),00)} = \begin{bmatrix} \bar{x}_{n-2} - \bar{x}_{n-1} & \bar{y}_{n-2} - \bar{y}_{n-1} \\ 0 & 0 \\ \bar{x}_{n-1} - \bar{x}_{n-2} & \bar{y}_{n-1} - \bar{y}_{n-2} \\ 0 & 0 \end{bmatrix} \qquad (9)$$

Then, R_{e_n-2} can be described as:

$$R_{e\_n-2} = \begin{bmatrix} \bar{x}_{n-1} - \bar{x}_{n-2} & \bar{y}_{n-1} - \bar{y}_{n-2} \\ -(\bar{y}_{n-1} - \bar{y}_{n-2}) & -(\bar{x}_{n-1} - \bar{x}_{n-2}) \end{bmatrix} \quad (10)$$

Since only one point p_{n-1} is related to p_{n-2}, the second row is another expression for the relationship with p_{n-1} and p_{n-2}. Hence, the control law for p_{n-2} can be described as:

$$\delta \dot{p}_{n-2}(t) = k_{n-2} I_2 R_{e\_n-2} R_{((n-2)(n-1),00)} \delta p_{n-2}(t) \quad (11)$$

Similarly, we have:

$$\delta \dot{p}_{n-1}(t) = k_{n-1} I_2 R_{e\_n-1} R_{((n-1)n,00)} \delta p_{n-1}(t) \quad (12)$$

where,

$$R_{((n-1)n,00)} = \begin{bmatrix} \bar{x}_{n-1} - \bar{x}_n & \bar{y}_{n-1} - \bar{y}_n \\ 0 & 0 \\ \bar{x}_n - \bar{x}_{n-1} & \bar{y}_n - \bar{y}_{n-1} \\ 0 & 0 \end{bmatrix} \quad (13)$$

$$R_{e\_n-1} = \begin{bmatrix} \bar{x}_n - \bar{x}_{n-1} & \bar{y}_n - \bar{y}_{n-1} \\ -(\bar{y}_n - \bar{y}_{n-1}) & -(\bar{x}_n - \bar{x}_{n-1}) \end{bmatrix} \quad (14)$$

and

$$\delta \dot{p}_n(t) = k_n I_2 R_{e\_n} R_{(n(n-2),00)} \delta p_n(t) \quad (15)$$

$$R_{(n(n-2),00)} = \begin{bmatrix} \bar{x}_n - \bar{x}_{n-2} & \bar{y}_n - \bar{y}_{n-2} \\ 0 & 0 \\ \bar{x}_{n-2} - \bar{x}_n & \bar{y}_{n-2} - \bar{y}_n \\ 0 & 0 \end{bmatrix} \quad (16)$$

$$R_{e\_n} = \begin{bmatrix} \bar{x}_{n-2} - \bar{x}_n & \bar{y}_{n-2} - \bar{y}_n \\ -(\bar{y}_{n-2} - \bar{y}_n) & -(\bar{x}_{n-2} - \bar{x}_n) \end{bmatrix} \quad (17)$$

(B) Disconnection at edge (5, 6)

Next, it is assumed that the edge between leaders 5 and 6 is broken (Figure 6).

As shown in Figure 7, when edge (5, 6) is cut off, leaders 4 and 5 should change their orientation to receive information from leader 6. In this case, two spare lines must be utilized at the same time.
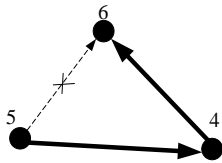

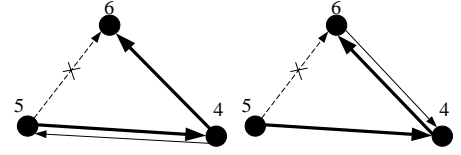**Figure 6.** Disconnection at edge (5, 6)

Figure 7. Type B fault tolerance (Both edges (5, 4) and (4, 6) are added)

Hence, the detailed control laws can be expressed as:

$$\dot{p}_{n-2} = k_{n-2} \frac{\| p_n - p_{n-2} \| - d_{(n-2,n)\_o}}{\| p_n - p_{n-2} \|} (p_n - p_{n-2}) \quad (18)$$

$$\dot{p}_{n-1} = k_{n-1} \frac{\| p_{n-2} - p_{n-1} \| - d_{(n-1,n-2)\_o}}{\| p_{n-2} - p_{n-1} \|} (p_{n-2} - p_{n-1}) \quad (19)$$

$$\delta \dot{p}_{n-2}(t) = k_{n-2} I_2 R_{e\_n-2} R_{((n-2)(n-1),00)} \delta p_{n-2}(t) \quad (20)$$

$$R_{((n-2)n,00)} = \begin{bmatrix} \bar{x}_{n-2} - \bar{x}_n & \bar{y}_{n-2} - \bar{y}_n \\ 0 & 0 \\ \bar{x}_n - \bar{x}_{n-2} & \bar{y}_n - \bar{y}_{n-2} \\ 0 & 0 \end{bmatrix} \quad (21)$$

$$R_{e\_n-2} = \begin{bmatrix} \bar{x}_n - \bar{x}_{n-2} & \bar{y}_n - \bar{y}_{n-2} \\ -(\bar{y}_n - \bar{y}_{n-2}) & -(\bar{x}_n - \bar{x}_{n-2}) \end{bmatrix} \quad (22)$$

$$\delta \dot{p}_{n-1}(t) = k_{n-1} I_2 R_{e\_n-1} R_{((n-1)(n-2),00)} \delta p_{n-1}(t) \quad (23)$$

$$R_{((n-1)(n-2),00)} = \begin{bmatrix} \bar{x}_{n-1} - \bar{x}_{n-2} & \bar{y}_{n-1} - \bar{y}_{n-2} \\ 0 & 0 \\ \bar{x}_{n-2} - \bar{x}_{n-1} & \bar{y}_{n-2} - \bar{y}_{n-1} \\ 0 & 0 \end{bmatrix} \quad (24)$$

$$R_{e\_n-1} = \begin{bmatrix} \bar{x}_{n-2} - \bar{x}_{n-1} & \bar{y}_{n-2} - \bar{y}_{n-1} \\ -(\bar{y}_{n-2} - \bar{y}_{n-1}) & -(\bar{x}_{n-2} - \bar{x}_{n-1}) \end{bmatrix} \quad (25)$$
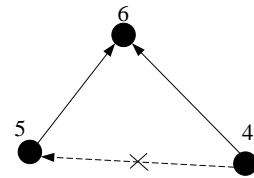
(C) Disconnection at edge (4, 5)

Figure 8. Disconnection at edge (4, 5)

Finally, it is assumed that the edge between leaders 4 and 5 is broken (Figure 8). In this case, leaders 4 and 5 should change their orientation to receive information from leader 6. Hence, the detailed control laws can be expressed as:

$$\dot{p}_{n-2} = k_{n-2} \frac{\parallel p_n - p_{n-2} \parallel - d_{(n-2,n)\_o}}{\parallel p_n - p_{n-2} \parallel}(p_n - p_{n-2}) \qquad (26)$$

$$\dot{p}_{n-1} = k_{n-1} \frac{\parallel p_n - p_{n-1} \parallel - d_{(n-1,n)\_o}}{\parallel p_n - p_{n-1} \parallel}(p_n - p_{n-1}) \qquad (27)$$

$$\delta \dot{p}_{n-2}(t) = k_{n-2} I_2 R_{e\_n-2} R_{((n-2)(n-1),00)} \delta p_{n-2}(t) \qquad (28)$$

$$R_{((n-2)n,00)} = \begin{bmatrix} \overline{x}_{n-2} - \overline{x}_n & \overline{y}_{n-2} - \overline{y}_n \\ 0 & 0 \\ \overline{x}_n - \overline{x}_{n-2} & \overline{y}_n - \overline{y}_{n-2} \\ 0 & 0 \end{bmatrix} \qquad (29)$$

$$R_{e\_n-2} = \begin{bmatrix} \overline{x}_n - \overline{x}_{n-2} & \overline{y}_n - \overline{y}_{n-2} \\ -(\overline{y}_n - \overline{y}_{n-2}) & -(\overline{x}_n - \overline{x}_{n-2}) \end{bmatrix} \qquad (30)$$

$$\delta \dot{p}_{n-1}(t) = k_{n-1} I_2 R_{e\_n-1} R_{((n-1)n,00)} \delta p_{n-1}(t) \qquad (31)$$

$$R_{((n-1)n,00)} = \begin{bmatrix} \overline{x}_{n-1} - \overline{x}_n & \overline{y}_{n-1} - \overline{y}_n \\ 0 & 0 \\ \overline{x}_n - \overline{x}_{n-1} & \overline{y}_n - \overline{y}_{n-1} \\ 0 & 0 \end{bmatrix} \qquad (32)$$

$$R_{e\_n-1} = \begin{bmatrix} \overline{x}_n - \overline{x}_{n-1} & \overline{y}_n - \overline{y}_{n-1} \\ -(\overline{y}_n - \overline{y}_{n-1}) & -(\overline{x}_n - \overline{x}_{n-1}) \end{bmatrix} \qquad (33)$$

4.2.2 Control laws for the whole formation

The straightforward translation for the object is difficult to be discussed about the three co-leaders persistent formation. Without loss of generality, the general translation can be completed in two steps: x-axis translation and y-axis translation. Thus, the straightforward translation was replaced with x-axis and y-axis translations for analysis.

For x-axis translation, if the leader that discovers the object is point n, then the values of $\delta y_n, \delta x_n, \delta y_{n-1}$, and δy_{n-2} are all zero, and the values of $\overline{y}_{n-2} - \overline{y}_{n-1}, \overline{y}_n - \overline{y}_{n-1}$, and $\overline{y}_{n-2} - \overline{y}_n$ are also zero. Then, we have:

$$\begin{bmatrix} \delta \dot{x}_{n-2} \\ \delta \dot{y}_{n-2} \end{bmatrix} = \begin{bmatrix} -k_{n-2}(\delta x_{n-2} - \delta x_{n-1}) \\ 0 \end{bmatrix} \qquad (34)$$

$$\begin{bmatrix} \delta \dot{x}_{n-1} \\ \delta \dot{y}_{n-1} \end{bmatrix} = \begin{bmatrix} -k_{n-1}(\delta x_{n-1} - \delta x_n) \\ 0 \end{bmatrix} \qquad (35)$$

$$\begin{bmatrix} \delta \dot{x}_n \\ \delta \dot{y}_n \end{bmatrix} = 0 \qquad (36)$$

Likewise, for y-axis translation, we have:

$$\begin{bmatrix} \delta \dot{x}_{n-2} \\ \delta \dot{y}_{n-2} \end{bmatrix} = \begin{bmatrix} 0 \\ -k_{n-2}(\delta y_{n-2} - \delta y_{n-1}) \end{bmatrix} \qquad (37)$$

$$\begin{bmatrix} \delta \dot{x}_{n-1} \\ \delta \dot{y}_{n-1} \end{bmatrix} = \begin{bmatrix} 0 \\ -k_{n-1}(\delta y_{n-1} - \delta y_{n-2}) \end{bmatrix} \qquad (38)$$

$$\begin{bmatrix} \delta \dot{x}_n \\ \delta \dot{y}_n \end{bmatrix} = 0 \qquad (39)$$

Obviously, x-axis translation is equivalent to y-axis translation. Thus, only x-axis translation is discussed here.

For the linear system, the new control laws can be expressed as:

$$\begin{bmatrix} \delta \dot{x}_1 \\ \delta \dot{y}_1 \\ \delta \dot{x}_2 \\ \delta \dot{y}_2 \\ \vdots \\ \delta \dot{x}_{n-2} \\ \delta \dot{x}_{n-1} \end{bmatrix} = AB \begin{bmatrix} \delta x_1 \\ \delta y_1 \\ \delta x_2 \\ \delta y_2 \\ \vdots \\ \delta x_{n-2} \\ \delta x_{n-1} \end{bmatrix} \qquad (40)$$

$$A = \begin{bmatrix} \alpha & \beta & \gamma \end{bmatrix}$$

$$B = \begin{bmatrix} K_\alpha R_{e\_\alpha}^{-1} R_\alpha & K_\beta R_{e\_\beta}^{-1} R_\beta & \gamma K_\gamma R_{e\_\gamma}^{-1} R_\gamma \end{bmatrix}^T \qquad (41)$$

where, $\widehat{K_\alpha} = \widehat{K_\beta} = \widehat{K_\gamma} = (\overset{i=1}{\underset{n-3}{\oplus}} K_i) \oplus k_{n-2} \oplus k_{n-1}$,

$$A = \begin{cases} [1 \quad 0 \quad 0] \; When \; a \; form \; fault \; torellance \\ [0 \quad 1 \quad 0] \; When \; b \; form \; fault \; torellance \\ [0 \quad 0 \quad 1] \; When \; c \; form \; fault \; torellance \end{cases}$$

Here, both $\widehat{R_e}$ and $\hat{R}$ are $(2n-4) \times (2n-4)$, $\widehat{R_e}$ is obtained by removing (2n-4), (2n-2), (2n-1) and 2n rows and columns from R_e; $\hat{R}$ is obtained by removing (2n-4), (2n-2), (2n-1), 2n rows and (2n-3) column from R.

Let V' be the set of followers, and V_f' be a subset of V'. Then, we have:

$$R_\alpha = \begin{bmatrix} R(V') & & X & \\ 0 & \overline{x}_{n-2} - \overline{x}_{n-1} & \overline{x}_{n-1} - \overline{x}_{n-2} & 0 \\ 0 & 0 & \overline{x}_{n-1} - \overline{x}_n & \overline{x}_n - \overline{x}_{n-1} \end{bmatrix} \qquad (42)$$

$$R_{e\_\alpha} = (\overset{n-3}{\underset{i=1}{\oplus}} E_i) \oplus (\overline{x}_{n-1} - \overline{x}_{n-2}) \oplus (\overline{x}_n - \overline{x}_{n-1}) \qquad (43)$$

$$R_\beta = \begin{bmatrix} R(V') & & X & \\ 0 & \overline{x}_{n-2} - \overline{x}_{n-1} & \overline{x}_{n-1} - \overline{x}_{n-2} & 0 \\ 0 & \overline{x}_{n-2} - \overline{x}_n & 0 & \overline{x}_n - \overline{x}_{n-2} \end{bmatrix} \qquad (44)$$

$$R_{e\_\beta} = (\overset{n-3}{\underset{i=1}{\oplus}} E_i) \oplus (\overline{x}_{n-2} - \overline{x}_{n-1}) \oplus (\overline{x}_n - \overline{x}_{n-2}) \qquad (45)$$

$$R_\gamma = \begin{bmatrix} R(V') & X \\ 0 & 0 & \overline{x}_{n-1} - \overline{x}_n & \overline{x}_n - \overline{x}_{n-1} \\ 0 & \overline{x}_{n-2} - \overline{x}_n & 0 & \overline{x}_n - \overline{x}_{n-2} \end{bmatrix} \qquad (46)$$

$$R_{e\_\gamma} = (\overset{n-3}{\underset{i=1}{\oplus}} E_i) \oplus (\overline{x}_n - \overline{x}_{n-1}) \oplus (\overline{x}_n - \overline{x}_{n-2}) \qquad (47)$$

Suppose that the two outgoing edges from node i are $i{\rightarrow}j$ and $i{\rightarrow}k$. Then, we have:

$$E_i = -\begin{bmatrix} x_i - x_j & y_i - y_j \\ x_i - x_k & y_i - y_k \end{bmatrix} \qquad (48)$$

5. CONCLUSIONS

This paper explores deep into the emergency management of government data security based on the AI. Various AI techniques were created to ensure government data security, including intelligent government data security monitoring technology, ANN and fuzzy identification technology, and firewall security technology based fuzzy clustering. In the meantime, an AI-based security model was established on the novel concept of the GDRC. On this basis, the authors put forward an emergency management system for government data security, and verified its ability to safely control multiple datacenters through simulation of multi-agent formation. The future research will introduce more security factors to the proposed security model.

REFERENCES

[1] Khan, Z., Pervez, Z., Ghafoor, A. (2014). Towards cloud based smart cities data security and privacy management. 2014 IEEE/ACM 7th International Conference on Utility and Cloud Computing, London, pp. 806-811. https://10.1109/UCC.2014.131

[2] Colin, T. (2012). Big data security. Network Security, 7: 5-8. https://10.1016/S1353-4858(12)70063-6

[3] Zhang, D. (2018). Big data security and privacy protection. Advances in Computer Science Research, 77: 275-278. https://doi.org/10.2991/icmcs-18.2018.56

[4] Ramachandran, M., Chang, V. (2016). Towards performance evaluation of cloud service providers for cloud data security. International Journal of Information Management, 36(4): 618-625. https://doi.org/10.1016/j.ijinfomgt.2016.03.005

[5] Gunasekaran, M., Chandu, T., Daphne, L., Revathi S. (2017). Big data security intelligence for healthcare industry 4.0. Cybersecurity for Industry 4.0, 7: 103-126. https://doi.org/10.1007/978-3-319-50660-9_5

[6] Syed, I.H.S., Vasilis, P., Ioannis, M. (2020). Government (Big) data ecosystem: definition, classification of actors, and their roles. International Journal of Computer and Information Engineering, 14(4): 102-114. https://doi.org/10.5281/zenodo

[7] Edoardo, C., Simone, C., Marco, G., Mauro, I., Davide, M., Pietro, P. (2019). Modeling and evaluating a complex edge computing based systems: An emergency management support system case study. Internet of Things, 6: 156-168. https://doi.org/10.1016/j.iot.2019.100054

[8] Puthal, D., Nepal, S., Ranjan R., Chen, J. (2016). A secure big data stream analytics framework for disaster management on the cloud. 2016 IEEE 18th International Conference on High Performance Computing and Communications; IEEE 14th International Conference on Smart City; IEEE 2nd International Conference on Data Science and Systems (HPCC/SmartCity/DSS), Sydney, NSW, pp. 1218-1225 https://doi.org/10.1109/HPCC-SmartCity-DSS.2016.0170

[9] Carminati, B., Ferrari, E., Guglielmi, M. (2016). Detection of unspecified emergencies for controlled information sharing. IEEE Transactions on Dependable and Secure Computing, 13(6): https://doi.org/630-643.10.1109/TDSC.2015.2427846

[10] Chen, C., Li, C., Liu, S., Wu T., Pan, J. (2017). A provable secure private data delegation scheme for mountaineering events in emergency system. IEEE Access, 5: 3410-3422. https://doi.org/10.1109/ACCESS.2017.2675163

[11] Jackson, D., Hayes, P. (2016). Ensuring security of data and information flow in emergency response decision support. 2016 11th International Conference on Availability, Reliability and Security (ARES), Salzburg, pp. 792-797. https://doi.org/10.1109/ARES.2016.49

[12] Cao, H., Bai, Y., Chen, J., Fang, H. (2013). Control of 2D minimally persistent formations with three co-leaders in a cycle. International Journal of Advanced Robotic Systems, 210(21): 1-7. https://doi.org/10.5772/54494

[13] Cao, H., Bai, Y., Liu, H. (2012). Distributed rigid formation control algorithm for multi-agent systems. Kybernetes, 41(10): 1650-1661. https://doi.org/10.1108/03684921211276819

[14] Olfati-Saber, R., Murray, R. (2002). Distributed cooperative control of multiple vehicle formations using structural potential functions. Proc. of 15th IFAC World Congress, 35(1): 495-500. https://doi.org/10.3182/20020721-6-es-1901.00244

[15] Jiang, L., Zhang, R. (2012). Stable formation control of multi-robot system with communication delay. International Journal of Advanced Robotic Systems, 9(1): 1-10. https://doi.org/10.5772/45603

[16] Tanner, H., Jadbabaie, A., Pappas, G.J. (2003). Stable flocking of mobile agents, part I: fixed topology. 42nd IEEE International Conference on Decision and Control (IEEE Cat. No.03CH37475), Maui, HI, pp. 2010-2015. https://doi.org/10.1109/CDC.2003.1272910

[17] Saulnier, K., Saldaña, D., Prorok, A., Pappas, G.J., Kumar, V. (2017). Resilient flocking for mobile robot teams. IEEE Robotics and Automation Letters, 2(2): 1039-1046. https://doi.org/10.1109/LRA.2017.2655142

[18] Ramazani, S., Selmic, R., de Queiroz, M. (2017). Rigidity-based multiagent layered formation control. IEEE Transactions on Cybernetics, 47(8): 1902-1913. https://doi.org/10.1109/TCYB.2016.2568164

[19] Krick, L., Broucke, M.E., Francis, B.A. (2009). Stabilisation of infinitesimally rigid formations of multi-robot networks. International Journal of Control, 82(3): 423-439. https://doi.org/10.1109/CDC.2008.4738760

[20] Asimow, L., Roth, B. (1978). Rigidity of graphs, II.

Journal of Mathematical Analysis and Applications, 68(1): 171-190. https://doi.org/10.1016/0022-247x(79)90108-2

[21] Yu, C., Anderson, B.D.O., Dasgupta, S. (2009). Control of minimally persistent formations in the plane. Siam Journal on Control and Optimization, 48(1): 206-233. https://doi.org/10.1137/060678592