EAI Endorsed Transactions
on Scalable Information Systems
Research Article EAI.EU

Uncertainty-Aware Decision-Making of Robust Bayesian Networks with Distributed Data Security in Adaptive Artificial Intelligence Systems

Keli Wei[1], Taihui Mao[1,2], Yongchao Sun[1] and Xingyi Song[1,*]

[1]Beijing Guangnian Wuxian Technology Co., LTD., Beijing 100085, China
[2]School of Management, Xi 'an Jiaotong University, Xi 'an, Shaanxi 710049, China

Abstract

INTRODUCTION: The reliability of decision-making in adaptive artificial intelligence (AI) systems is limited by uncertain factors such as noise in multi-source sensing data and conflicts in decision-making objectives. Furthermore, in distributed multi-node collaborative environments, challenges such as cross-node data leakage and edge-node privacy risks further exacerbate decision uncertainty.
OBJECTIVES: To address this issue, an uncertainty-aware decision-making method for adaptive AI systems based on robust Bayesian networks is proposed, with a specific focus on distributed data security and privacy protection.
METHODS: A robust Bayesian network-based model is proposed. Multi-source sensing data are fused using adaptive weighted minimum mean square error optimization. Secure distributed aggregation is achieved through encrypted aggregation and differential privacy mechanisms. Federated transfer learning is introduced to prevent raw data sharing, and minimax estimation is used to correct missing-data bias.
RESULTS: Experimental results on distributed inspection robot clusters show that the proposed method effectively achieves uncertainty-aware decision-making under dynamic and obstacle-affected scenarios while maintaining privacy protection and data security.
CONCLUSION: The proposed framework improves decision robustness and privacy preservation in distributed adaptive AI systems by integrating Bayesian inference, federated learning, and secure data fusion strategies.

Keywords: distributed data security, privacy protection, secure distributed aggregation, robust bayesian network, adaptive artificial intelligence system, uncertainty, perception decision-making, adaptive weighted fusion, federated transfer learning

Received on 27 March 2026, accepted on 02 June 2026, published on 25 June 2026

Copyright © 2026 Keli Wei *et al.*, licensed to EAI. This is an open access article distributed under the terms of the CC BY-NC-SA 4.0, which permits copying, redistributing, remixing, transformation, and building upon the material in any medium so long as the original work is properly cited.

doi: 10.4108/eetsis.12388

1. Introduction

In the era of in-depth integration of digitalization and intellectualization, adaptive artificial intelligence (AI) systems are widely applied in key fields such as industrial monitoring, medical diagnosis, and intelligent transportation due to their dynamic adjustment capabilities [1,2]. However, their operating environment is highly uncertain, which is reflected in three aspects: multi-source sensing data are susceptible to noise interference, decision-making objectives have multi-dimensional conflicts, and target domain data are scarce. These factors severely restrict the reliability and robustness of system decision-making [3]. In distributed deployment scenarios—such as multi-node inspection robot clusters or cross-terminal sensing networks—additional security challenges arise, including data leakage during cross-node transmission and privacy risks associated with edge-node sensing data [4].

*Corresponding author. Email: xingyisong23@163.com

Currently, Virginia et al. [5] proposed a graph-based intelligent decision-making method, but failed to design a flexible optimization mechanism for the real-time adjustment of decisions in adaptive systems; Park et al. [6] studied an interpretable deep learning LSTM model, yet did not design a dedicated quantification module to clearly mark the confidence interval of decision conclusions; Ramsha et al. [7] researched an intelligent decision-making method based on deep neural networks, where the basic features extracted at the bottom layer rely on the stability of data distribution, and the decision accuracy is prone to degradation when facing uncertainties; Azimirad et al. [8] studied a decision-making method based on continuous hybrid spiking convolutional (CHSC) neural networks, but did not design a quantification module for uncertainties, making it impossible to evaluate the reliability of spike pulses and output the confidence level of decision results. Recent studies have begun exploring privacy-preserving distributed Bayesian networks and federated transfer learning in multi-node systems, yet the integration of distributed data security with uncertainty-aware decision-making remains underexplored [9-11].

In response to the aforementioned research gaps, this paper proposes an uncertainty aware decision-making method based on robust Bayesian networks, which innovatively integrates distributed data security mechanisms. The main contributions of this article are summarized as follows:

(1) A secure distributed adaptive weighted fusion method for multi-source perception data is proposed. By integrating differential privacy and homomorphic encryption techniques, the fusion weights are dynamically optimized with the goal of minimizing mean square error while ensuring node data privacy. This effectively suppresses sensor data noise and provides a high-quality and secure data foundation for subsequent decision-making.

(2) A collaborative "security decision" uncertainty aware modeling framework has been constructed: the framework incorporates distributed data security status as a key variable into a robust Bayesian network, quantifies uncertainty using posterior probability, corrects probability bias using minimax estimation, and ultimately selects the optimal decision through an improved complete acceptability criterion, forming a complete robust decision logic chain.

(3) A privacy protection parameter optimization algorithm based on federated transfer learning has been designed: the algorithm effectively solves the problem of data scarcity in the target domain by decomposing Bayesian network modules, calculating structure and parameter similarity, and implementing encrypted weighted fusion. At the same time, it prevents the leakage of raw data during the parameter learning process.

(4) The effectiveness of the method was verified through distributed detection of robot cluster scenarios: experimental results showed that this method significantly outperformed existing baseline methods in terms of multi-source data fusion quality, parameter learning accuracy, and decision stability in complex uncertain environments.

2. Uncertainty Perception and Decision-Making Methods in Adaptive AI Systems with Distributed Data Security

2.1. Topology of Adaptive Artificial Intelligence Systems in Distributed Environments

Prior to investigating uncertainty perception decision-making methods in adaptive AI systems, it is essential to clarify the system's topological structure. This involves defining the hierarchical relationship between the internally independent design and externally deployed binding layers, as well as the adaptive adjustment mechanism driven by the synergistic interaction of internal and external factors [12,13]. In distributed multi-node collaborative architectures, the topology must additionally account for inter-node communication links, data synchronization protocols, and security boundaries [14]. Figure 1 illustrates the topology of the adaptive AI system in a distributed setting.

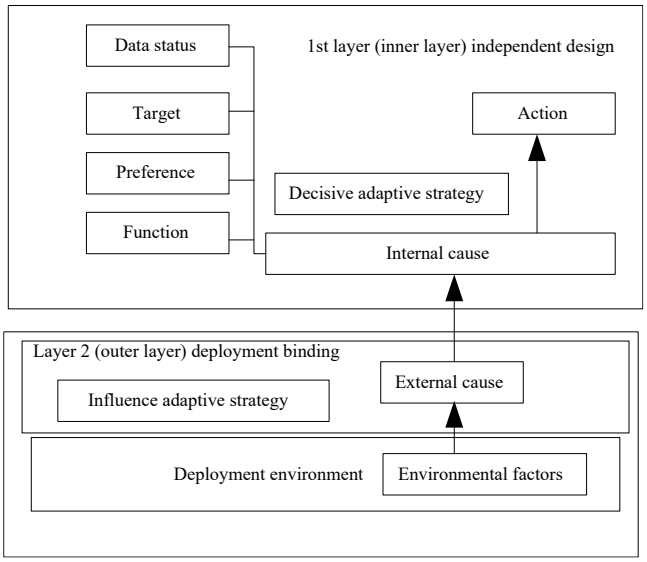

Figure 1. Topology of adaptive artificial intelligence system in distributed environment

From the architectural perspective shown in Figure 1, the adaptive system is divided into two layers. The first layer, serving as the inner layer, adopts an independent design pattern. Its core elements include actions and internal factors. Action generation is driven by a deterministic adaptive strategy [15], which in turn relies on the system's own data state, objectives, preferences, and intrinsic functional attributes [16].

The second layer, the outer layer, constitutes the deployment binding level and primarily involves external factors. Environmental elements within the deployment context exert influence on these external factors [17]. In distributed settings, external factors also include neighboring node states, network conditions, and potential security threats such as data interception or node tampering [18]. These factors then impact the inner layer through an influence-based adaptive strategy, jointly influencing the system's adaptive process with internal factors. This enables the system to adaptively adjust and optimize itself across diverse environments [19]. The security-aware extension of this topology provides an architectural basis for incorporating distributed data security into decision-making.

2.2. Adaptive Weighted Fusion Method for Multi-Source Perception Data with Secure Distributed Aggregation

Based on the previously established adaptive AI system topology—where the inner layer drives decision actions and the outer layer influences system adaptation—it is evident that inner-layer decision-making relies on high-quality system state information. However, multi-source perception units deployed in the outer layer environment are susceptible to noise interference, leading to fluctuations in data accuracy [20]. Moreover, in distributed environments, the transmission of raw sensing data across nodes creates additional privacy risks, as sensitive information may be exposed during collection, transmission, or fusion [21]. To resolve this contradiction and ensure the reliability of decision inputs while safeguarding data privacy, a multi-source perception data fusion mechanism adapted to this topology and enhanced with secure distributed aggregation must be established.

The proposed secure aggregation mechanism operates as follows: each perception node first applies differential privacy-based desensitization to its raw measurements before transmission. Specifically, Laplace noise calibrated to the sensitivity of the data is added to each measurement, ensuring ε-differential privacy. The desensitized measurements are then encrypted using a lightweight homomorphic encryption scheme, allowing the fusion center to aggregate encrypted values without decrypting individual contributions. The fusion center computes the weighted sum in the encrypted domain, and only the final aggregated result is decrypted. This two-layer protection—differential privacy at the node level and homomorphic encryption during transmission—prevents both the fusion center and potential eavesdroppers from accessing raw node-level data.

By analyzing perception noise characteristics, optimizing the mean squared error objective to solve for adaptive weights, and dynamically adjusting perception unit weights through the system's adaptive mechanism, precise fusion of multi-source data is achieved without exposing individual node data. This provides a high-quality and privacy-preserving data foundation for subsequent robust Bayesian network uncertainty inference.

During operation of the adaptive AI system, assume there are m perception units collecting state parameters of the target environment or object. Based on the adaptive weighted fusion model, construct an observation model for the system's state:

$$Y = Ky + e \tag{1}$$

In the equation, y represents the one-dimensional state vector to be estimated by the system; $Y = [y_1, y_2, ..., y_m]^T$ denotes the m-dimensional measurement vector, y_j is the measurement value of the j th perceptron, $e = [e_1, e_2, ..., e_m]^T$ is the m-dimensional measurement noise vector encompassing both internal noise within the perceptron and environmental interference noise. K is the known m-dimensional constant vector, and the measurement values $y_1, y_2, ..., y_m$ of each perceptron are mutually independent, each being an unbiased estimate of y.

(i) Perceptual Noise Analysis and Weight Construction

Assume the measurement noise e_j from each sensor unit is independent, and the noise is a stationary random process with state-dependent properties. Based on this, construct the weight vector $\varpi = [\varpi_1, \varpi_2, ..., \varpi_m]$ for m sensor units. The fused state estimate $\hat{y}$ satisfies:

$$\begin{cases} \hat{y} = \sum_{j=1}^{m} \varpi_j y_j \\ \sum_{j=1}^{m} \varpi_j = 1 \end{cases} \tag{2}$$

The weight ϖ_j reflects the importance of the measurement results from the j th perceptron during the fusion process. The smaller the noise variance α_j^2 (i.e., the higher the measurement accuracy), the larger the corresponding weight ϖ_j.

(ii) Mean Squared Error Optimization and Adaptive Weight Solution

To achieve optimal accuracy in fusion results, the adaptive AI system aims to minimize the total mean squared error. The total mean squared error is expressed as:

$$\alpha^2 = E\left[\left(y - \hat{y} \right)^2 \right] = \sum_{j=1}^{m} \varpi_j^2 \alpha_j^2 \tag{3}$$

Where, E is the error function. Under the constraint that the sum of weights equals 1, let $x = \sum_{j=1}^{m} \varpi_j^2 \alpha_j^2$. Taking the partial derivative of x with respect to ϖ_j and setting it to zero yields the optimal weights:

$$\varpi_j = 1/\alpha_j^2 \sum_{j=1}^{m} \frac{1}{\alpha_j^2} \quad (4)$$

The corresponding minimum mean square error is:

$$\alpha_{Min} = 1/\sum_{j=1}^{m} \frac{1}{\alpha_j^2} \quad (5)$$

(iii) Integration of System Adaptive Mechanisms

The adaptive artificial intelligence system dynamically updates the weight vector by continuously monitoring the operational status of each perception unit. When environmental interference increases the α_j^2 of a perception unit, the system automatically reduces its weight. Conversely, if improved measurement accuracy decreases the α_j^2, the weight is increased. This dynamic weight adjustment based on perception unit noise characteristics enables the system to adapt to varying environmental conditions, ensuring the accuracy and reliability of multi-source perception data fusion. The encrypted aggregation ensures that the fusion center never accesses raw node data, and differential privacy guarantees that even if encrypted data is compromised, individual measurements cannot be reverse-engineered. This provides high-quality fused data for subsequent intelligent decision-making.

2.3. Uncertainty-Aware Decision-Making Model Based on Robust Bayesian Network with Security States

Description of Uncertainty-Aware Decision-Making Problem with Security Objectives

Through multi-source perception data fusion methods, key information such as denoised system states and environmental dynamics has been obtained, forming the initial input for adaptive system decision-making. However, the decision process is not a single-stage static judgment but requires handling uncertainty in complex scenarios involving multiple stages and objectives. Furthermore, in distributed environments, decision-making must also account for data security and privacy protection as additional objectives, such as privacy of data transmission and controllability of node-level data access [22]. Therefore, it is essential to first define the core elements of uncertainty perception decision-making: delineating the conditional information set, decision phase set, decision scheme set, and decision objective set —now extended with security-related objectives. This establishes the problem boundaries for subsequent robust Bayesian network model design and inference.

(1) Conditional Information Set Y': Prior to decision-making, adaptive AI systems collect conditional information—such as environmental and self-state data—through multi-source perception modules. After multi-source fusion as described in Section 2.2, this information forms the set Y'. This set encompasses both deterministic system parameters and extensive uncertain environmental dynamics, providing the initial basis for system decisions [23]. In distributed settings, Y' also includes security state information such as node-level data protection levels and cross-domain security control effectiveness.

Decision-Making Stage Set C: The system's decision-making process is divided into n decision-selection stages with temporal sequence, forming the set $C = \{C_1, C_2, ..., C_n\}$. The decision outcomes from preceding stages directly influence subsequent decision judgments [24].

(2) Decision Scheme Set R: The set of all feasible decision schemes $R = \{R_1, R_2, ..., R_n\}$ is formed by combining the selection results from each decision stage. For example, a decision scheme R_j can be expressed as:

$$\left(R_1 = Action_1, R_2 = Action_2, ..., R_n = Action_n\right) \quad (6)$$

Among these, $Action_n$ represents the decision action for the n th phase.

(3) Decision Objective Set W: The system must evaluate n decision objectives, forming the set $W = \{W_1, W_2, ..., W_n\}$. These objectives now include both traditional performance metrics (e.g., path efficiency, obstacle avoidance) and security metrics (e.g., data transmission privacy, node-level data access controllability). Each decision objective W has b^i possible values $W_i = \left(W_i^1, W_i^2, ..., W_i^{b^i}\right)$, and since the objective variables have different dimensions, they are not additive. The joint optimization of security and performance objectives is achieved through a weighted sum approach, with weights adjustable based on application requirements.

Model Design with Security Variable Integration

After clearly defining the elements and boundaries of the decision-making problem, it is necessary to design an uncertainty-aware decision-making model to address the three core requirements: "how to quantify uncertainty, how to accurately estimate probabilities under data missing, and how to select the optimal decision category". Additionally, the model must incorporate distributed data security states as input variables. This model takes the multi-source fused information and security state information as input, realizes

uncertainty quantification by calculating posterior probabilities through a Bayesian network; simplifies the joint probability calculation of multi-dimensional attributes based on the "attribute independence assumption"; addresses the probability deviation caused by missing sensing data with the help of "minimax estimation"; quantitatively analyzes the influence of security factors on decision uncertainty; and finally sorts the posterior probability intervals through the "improved complete admissibility criterion" to determine the optimal decision category, forming a complete logical chain of security-aware uncertainty-aware decision-making.

In this model, a robust Bayesian network expresses the dependency relationships between variables in a structured manner through a directed acyclic graph (DAG), as follows:

Nodes: represent the core variables in the system, mainly divided into three categories. Conditional information node (X): includes the environmental features (such as obstacle distance and angle) fused in section 2.2 and the safety status node (S) (such as node data protection level NPL). Decision solution node (A): represents the actions that the system can execute, such as robot steering angles (LTurn, RTurn, FTurn). Decision objective node (U): represents the final evaluation metric of the decision, such as path tracking error.

Directed edge: represents the causal dependency relationship between nodes. The edge pointing from the conditional information node (X) to the decision solution node (A) represents the causal logic of "perception action", where the current perceived information determines what action should be taken. The key innovation of this model is the edge pointing from the security state node (S) to the decision solution node (A), which quantifies the impact of security situation on decision reliability (i.e. posterior probability interval). For example, a lower data protection level (NPL value low) will increase the uncertainty of A's value.

Parent child relationship: In the model, if a directed edge points from node U to node V, then U is the parent node of V and V is the child node of U. For example, in the aforementioned robot obstacle avoidance scenario, "Front Obstacle Distance (FDistance)" serves as the parent node and directly affects the child node "Front Steering Angle (FTurn)". The security state node (such as NPL) also serves as the parent node and is the common parent node for all decision solution nodes.

Let S denote the set of distributed data security states, where each s_k represents a security attribute such as node-level data protection level or cross-domain security control effectiveness. These security states are treated as additional parent nodes in the Bayesian network, influencing both the likelihood of observed data and the posterior probabilities of decision outcomes. The conditional probability tables are extended to include security state dependencies, allowing the model to quantify how security postures affect decision uncertainty.

Before making a decision, the system needs to judge the possibility of the occurrence of various events based on the information obtained from multi-source sensing [25]. The posterior probability reflects the probability of the occurrence of decision-related events when the information is known. By calculating this probability, the system can quantify the potential possibilities of different decision directions from a probabilistic perspective, providing a basic basis for the subsequent evaluation of the advantages and disadvantages of decision-making schemes. In the research of this paper, the system adopts a Bayesian network [26]. The core of its learning lies in calculating the posterior probability:

$$q\left(d\left|y'\right.\right)=\frac{q\left(y'\left|d\right.\right)q\left(d\right)}{q\left(y'\right)\propto q\left(y'\left|d\right.\right)q\left(d\right)} \tag{7}$$

Where, the posterior probability $q\left(d\left|y'\right.\right)$ reflects the probability of the decision-related event d (various situations or states directly linked to the system's decision-making behavior) occurring after perceiving the information y'. $q\left(y'\left|d\right.\right)$ represents the likelihood probability, indicating the probability of perceiving the information y' under the condition that the decision-related event d occurs. It measures the degree of association between the perceived information and the decision-related event. $q\left(d\right)$ represents the prior probability, representing the initial assessment of an event d based on past experience or prior knowledge before acquiring perceptual information y' is obtained. $q\left(y'\right)$ represents the marginal probability, denotes the probability of the perceptual information y' itself occurring.

The security states S are incorporated into this framework by conditioning the likelihood probability: $q\left(d\left|y',S\right.\right)$, where different security states alter the expected observation patterns. For instance, when node-level protection is low (high privacy risk), the likelihood of data anomalies or incomplete observations increases, which in turn widens the posterior probability intervals and increases decision uncertainty. This quantitative linkage between security posture and decision reliability enables the system to adapt its decision strategies based on current security conditions.

In practical decision-making scenarios, the attribute variables influencing decisions (such as various environmental feature parameters and different system operational indicators, obtained through fusion as described in Section 2.2) are numerous and complex. The Extended Robust Bayesian Classifier (ER) assumes attribute variables are mutually independent given a class variable. This simplifies the calculation of the joint influence of multiple attribute variables on decision categories, enabling the system to process multidimensional perceptual information more efficiently.

It facilitates rapid analysis of decision category probabilities under different attribute combinations, aiding in the generation of decision schemes. ER extends traditional Bayesian networks by assuming attribute variables are independent given the class variable, expressed as:

$$q\left(Y_1'=y_1',...,Y_n'=y_n'|D\right)=\prod_{j=1}^{n}q\left(Y_j'=y_j'|D\right),j=1,2,...,n \quad (8)$$

Where, $q\left(Y_1'=y_1',...,Y_n'=y_n'|D\right)$ denotes the joint probability of the simultaneous occurrence of events n across n attribute variables $y_n'\in Y_n'$, given the class variable D. y_j' represents the specific value taken by the j th attribute variable Y_j'; $q\left(Y_j'=y_j'|D\right)$ denotes the conditional probability that the j th attribute variable Y_j' takes the value y_j', given the class variable D.

For security state variables, the independence assumption is similarly applied: $q\left(S_1'=s_1',...,s_k'=s_k'|D\right)$, meaning that given a decision class, different security attributes are assumed independent. This simplifies computation while still capturing the first-order effects of security on decision outcomes.

In practical operation, perceptual data often suffers from missing values (e.g., due to environmental sensor failures causing data loss or incomplete historical records), which directly impacts the accuracy of probability estimation. This section addresses the challenge of "accurately calculating conditional probabilities when data is missing" through "minimum maximum estimation" and "complete utilization of class variables." To estimate the conditional probability $q\left(y_{st}|d\right)$ from the data Y', we compute $\overline{n}\left(y_{st},d_r\right)$ (The maximum possible number of attribute variables y_s taking t and class variables D taking $\left(d_{r1},d_{r2},...,d_{rM}\right)$ in the missing data.) $\underline{n}\left(y_{st},d_r\right)$ (The maximum number of class variables taking $\left(d_{r1},d_{r2},...,d_{rM}\right)$ under the conditions that attribute variables y_s taking t and class variables D taking $\left(d_{r1},d_{r2},...,d_{rM}\right)$ in the missing data.) $n\left(y_{st},d_r\right)$ (The number of data entries where attribute variables y_s taking t and class variables D taking $\left(d_{r1},d_{r2},...,d_{rM}\right)$ in the non-missing data.) Assuming no missing values in class variables (complete decision category information) and only attribute variables have missing values, then:

$$\overline{n}\left(y_{st},d_r\right)=n\left(y_{st},d_r\right) \quad (9)$$

$$\underline{n}\left(y_{st},d_r\right)=n\left(y_{st},d_r\right) \quad (10)$$

When the class variable is fixed, the conditional probability of the attribute variable is calculated using the minimum-maximum estimation. The minimum estimate of the conditional probability is:

$$q\left(y_{st}|d_r\right)=\frac{\varepsilon_{st}+n\left(y_{st},d_r\right)}{\sum_k\left[\varepsilon_{sk}+\left(y_{sk},d_r\right)\right]+\underline{n}\left(y_{st},d_r\right)} \quad (11)$$

The maximum estimate is:

$$\overline{q}\left(y_{st}|d_r\right)=\frac{\varepsilon_{st}+n\left(y_{st},d_r\right)+\overline{n}\left(y_{st},d_r\right)}{\sum_k\left[\varepsilon_{sk}+\left(y_{sk},d_r\right)\right]+\overline{n}\left(y_{st},d_r\right)} \quad (12)$$

Where, ε_{st} and ε_{sk} represent prior information. y_{sk} denotes attribute variables distinct from y_{st}.

For security state variables, which may also suffer from missing or incomplete observations (e.g., when node protection level cannot be accurately measured), the same minimax estimation procedure is applied, ensuring robust probability intervals even under incomplete security monitoring.

Since the class variable has no missing values, its probability can be accurately estimated based on prior information and the statistical entry count of the class variable, thereby providing a critical basis for system decision-making. The class probability is estimated as:

$$\overline{q}\left(d_r\right)=\varepsilon_r+n\left(y_r\right)/\sum_{k=1}^{K}\left[\varepsilon_k+n\left(y_k\right)\right] \quad (13)$$

Where, ε_r and ε_k represent prior information, such as the baseline probability of hazardous events in historical records; $n\left(y_r\right)$ denotes the number of data entries where the class variable takes the value $\left(d_{r1},d_{r2},...,d_{rM}\right)$; $n\left(y_k\right)$ denotes the number of data entries where the class variable takes the value d_k.

After model training, "new decision instances" (e.g., unknown environmental scenarios, new system operating states) must be classified to determine their decision category, ultimately outputting the decision basis to solve the problem of "how to select the optimal decision based on probability." Thus, after the training phase, given an unlabeled instance z where the class variable is d_r, estimate the posterior probability interval $\left[q_f\left(d_r|z\right),q_p\left(d_r|z\right)\right]$ for the composite class variable The upper bound of the posterior probability is:

$$q_p(d_r|z)=\frac{\bar{q}(d_r)\prod\bar{q}(y_{st}|d_r)}{\bar{q}(d_r)\prod\bar{q}(y_{st}|d_r)+\sum_{k\neq r}\bar{q}(d_k)\prod\underline{q}(y_{st}|d_k)} \quad (14)$$

$$q_f(d_r|z)=\frac{\bar{q}(d_r)\prod\underline{q}(y_{st}|d_r)}{\bar{q}(d_r)\prod\underline{q}(y_{st}|d_r)+\sum_{k\neq r}\bar{q}(d_k)\prod\bar{q}(y_{st}|d_k)} \quad (15)$$

Equations (14) and (15) define the probability range for an event belonging to a class under uncertainty, enabling the system to assess the uncertainty level of the decision category judgment. When security states are incorporated, these intervals widen under poor security conditions, reflecting increased decision uncertainty due to potential data integrity or privacy concerns. $\bar{q}(d_k)$ represents the class probability estimate when the class variable takes d_k .

All posterior probability intervals are ranked using a scoring method, and the class with the highest interval score is assigned to the unclassified event. The instances with unknown class decision labels are classified using the full-permissibility criterion, defined as follows:

$$l_u(d_r)=q_f(d_r|z)-\frac{(q_p(d_r|z)-q_f(d_r|z))\sum_k(q_p(d_r|z)-1)}{\sum_k(q_p(d_k|z)-q_f(d_k|z))} \quad (16)$$

Among these, $q_p(d_k|z)$ and $q_f(d_k|z)$ represent the maximum and minimum posterior probabilities of the composite class variable when the class variable takes the value d_k for a given unlabeled instance z .

The corresponding union class variable with the maximum score is selected as the decision result, i.e.:

$$d_r=\operatorname{argmax} l_u(d_r) \quad (17)$$

Therefore, systems based on robust Bayesian networks can achieve effective perception and decision-making under uncertain conditions such as missing data and varying security postures, enhancing the system's adaptability and decision-making capabilities in complex, uncertain distributed scenarios. The explicit modeling of security states enables the system to make security-aware decisions, e.g., preferring more conservative actions when data protection levels are low. Ultimately, this enables the system to make reliable decisions even in complex, uncertain environments.

In summary, the uncertainty perception and decision-making method in adaptive artificial intelligence systems based on robust Bayesian networks with distributed data security employs the "posterior probability" to convert uncertain information into quantifiable metrics; utilizes the "property independence assumption" to adapt to multidimensional perception data; addresses probability biases caused by missing data through "minimum maximum estimation"; quantitatively incorporates security states into the probabilistic framework; and determines optimal decision outcomes using the "improved full-permissibility criterion." Ultimately, this enables the system to make reliable and secure decisions even in complex, uncertain environments.

Robust Bayesian Network Parameter Learning Based on Federated Transfer Learning with Privacy Preservation

Bayesian network parameter learning typically requires large amounts of data to ensure estimation accuracy. However, in many practical scenarios, the target domain (the domain to be learned) may suffer from data scarcity, such as in novel medical diagnosis scenarios or emerging industrial monitoring scenarios, where collecting sufficient labeled data is costly and challenging. Moreover, in distributed environments, direct sharing of raw data between source and target domains raises significant privacy concerns, as node-level data may contain sensitive information. Transfer learning leverages knowledge from the source domain (a related, data-rich domain) by transferring effective information learned in the source domain to the target domain. This compensates for the scarcity of data in the target domain, enabling effective Bayesian network parameter learning even with limited data. To address privacy risks, this paper adopts a federated transfer learning framework that prevents direct sharing of raw data between source and target domains, and designs secure parameter transmission and aggregation mechanisms across nodes.

The federated transfer learning framework operates as follows: each participating node (source domain) trains a local Bayesian network model on its own data without sharing raw observations. Only model parameters (conditional probability tables) are encrypted and transmitted to a central coordinator. The coordinator aggregates parameters from multiple source domains using secure multi-party computation, ensuring that individual node contributions cannot be decrypted or inferred. The aggregated parameters are then distributed to the target domain node, which combines them with its limited local data through weighted fusion. This framework preserves data locality while still enabling knowledge transfer across domains.

Transfer learning achieves robust parameter learning under data scarcity by decomposing Bayesian networks into local modules, selecting source domains with structurally and parametrically similar networks, calculating overall structural and parametric similarity between source and target domains to determine transfer weights, and finally weighting and fusing source and target domain parameters. The federated approach adds privacy-preserving constraints to each step: similarity calculations are performed using anonymized metadata rather than raw data, and parameter fusion occurs in the encrypted domain. This provides reliable and privacy-preserving parameter support for model inference.

The specific steps are described as follows:

(1) Identify candidate source domains for target domain nodes

Bayesian Networks (BNs) consist of nodes (representing variables) and directed edges (representing

dependencies between variables). The probability distribution of each node depends not only on itself but also on its parent nodes (nodes connected by directed edges pointing to it). The entire target domain BN is decomposed into several independent local modules. Each module centers around a single node, encompassing that node, all its parent nodes, and the dependency relationships and probability parameters between them, enabling fragment-based transfer learning.

Select a node V_m^T in the target domain and search in each source domain respectively. If this node exists in a source domain and has the same parent node set, then for node V_m^T, the target domain and this source domain have the same structural relationship. Moreover, if node exists in the source domain and has the same parent node set, then for node V_m^T, the target domain and this source domain have the same structural relationship, and this source domain is taken as a candidate source domain. Every node in the target domain needs to perform the above operation.

For node V_m^T, after identifying candidate source domains, the similarity between the candidate source domain and the target domain must be further evaluated to determine the migration weight, thereby avoiding the impact of negative migration. The similarity evaluation metric proposed in this paper consists of two parts: overall structural similarity and parameter similarity.

(2) Calculating the overall structural similarity between the alternative source domain and the target domain for this node

Based on semantic similarity, this paper proposes an evaluation criterion for the overall structural similarity between the alternative source domain and the target domain, as shown in Equation (18):

$$\Phi\left(\Gamma^t,\Gamma^s\right)=2\log\left(U+1\right)/\left(\log\left(U+1\right)+\log\left(U+H+1\right)\right) \quad (18)$$

Where, $\Phi\left(\Gamma^t,\Gamma^s\right)\in\left[0,1\right]$ and Γ^s denote the candidate source domain, Γ^t denotes the target domain. A higher value indicates greater overall structural similarity between the candidate source domain and the target domain, with a value of 1 representing perfect alignment. U and H represent the number of shared basic units and the number of unique units specific to the source domain, respectively.

(3) Calculating the parameter similarity between the alternative source domain and target domain for this node

This paper employs an expert knowledge constraint method to calculate the parameter similarity between the alternative source domain and target domain of a node. The proposed target domain expert knowledge constraints take two forms: Form 1 is qualitative, representing the probabilistic relationship of the same node under different states; Form 2 is quantitative, representing the parameter's value range. The specific forms are expressed as follows:

Form 1: Probability relationships for the same node under different states are expressed as:

$$\eta_{\phi\varphi h'}^t>\eta_{\phi\varphi h''}^t \quad (19)$$

Where, $1\le\phi\le m$, m denote the number of nodes in the BN model; $1\le\varphi\le\rho_\phi$, ρ_ϕ represent the number of state combinations for the parent node of the ϕ th node; h' and h'' are the distinct states of the ϕ th node. $\eta_{\phi\varphi h'}^t$ denotes the actual parameter of the target domain, satisfying that the ϕ th node is in the h' th state while its parent node set is in the φ th state; $\eta_{\phi\varphi h''}^t$ denotes the actual parameter of the target domain, satisfying that the ϕ th node is in the h'' th state while its parent node set is in the φ th state.

Form 2: The parameter's value range is expressed as:

$$\theta_{\phi\varphi h}<\eta_{\phi\varphi h}^t<\vartheta_{\phi\varphi h} \quad (20)$$

Where, $\eta_{\phi\varphi h}^t$ denotes the actual parameter of the target domain; $\theta_{\phi\varphi h}$ and $\vartheta_{\phi\varphi h}$ are constants that require prior configuration, obtainable through expert knowledge, operational experience, and statistical analysis of historical data.

Formula 1's expert knowledge is suitable for determining whether parameters from candidate source domains can be transferred, while Formula 2's expert knowledge is suitable for determining the role of candidate source domains in transfer learning]. Therefore, this paper first uses Formula 1 to determine whether parameters from candidate source domains can be used for transfer learning, then uses Formula 2 to calculate the similarity between the transferred candidate source domain and the target domain. If a node satisfies more instances of Formula 2, it is more likely to possess a probability distribution similar to that of the target domain. When a node's parameters in a candidate source domain satisfy one instance of Form 2, the similarity score for that candidate source domain increases by 1. Higher similarity scores can be assigned to more critical constraints. This scoring method evaluates the parameter similarity between candidate source domains and the target domain.

(4) Calculating the migration weight of alternative source domains

First, determine the structural weight for migration from the candidate source domain based on the overall structural similarity from step (2):

$$\tilde{\varpi}_j=\Phi\left(\Gamma^t,\Gamma^s\right)/\sum\Phi\left(\Gamma^t,\Gamma^s\right) \quad (21)$$

Then, calculate the parameter weight for the migration of the ∂ th alternative source domain:

$$\tilde{\varpi}_j^\partial=\Phi^\partial/\sum_\partial\Phi^\partial \quad (22)$$

Where, Φ^{∂} denotes the parameter similarity score of the ∂ th alternative source domain.

Finally, the final transfer weight for the candidate source domain is:

$$\tilde{\varpi}_j'^{\partial} = \tilde{\varpi}_j^{\partial} \tilde{\varpi}_j / \sum_{\partial} \tilde{\varpi}_j^{\partial} \tilde{\varpi}_j \tag{23}$$

Where $\tilde{\varpi}_j \in [0,1]$ is a balancing coefficient, typically set to 0.5.

(5) Based on the expert knowledge form in step (3), determine whether the parameters of the target domain for this node can be used for transfer learning. If the answer is "yes," proceed to step (6); if the answer is "no," proceed to step (7).

(6) Calculate the final parameters for the target domain of this node:

$$\hat{\eta}_{\phi\varphi h}^{t} = 1 - \tilde{\varpi}_j'^{\partial} \tilde{\eta}_{\phi\varphi h}^{t} + \sum_{\partial} \tilde{\varpi}_j'^{\partial} \eta_{\phi\varphi h}^{\partial} \tag{24}$$

Where, $\tilde{\eta}_{\phi\varphi h}^{t}$ denotes the target domain parameters obtained via maximum likelihood estimation; $\eta_{\phi\varphi h}^{\partial}$ denotes the parameters of the ∂ th candidate source domain obtained via maximum likelihood estimation; $\hat{\eta}_{\phi\varphi h}^{t}$ denotes the final target domain parameters obtained by weighting the candidate source domain and target domain parameters. $\tilde{\varpi}_j' \in [0,1]$ controls the trade-off between local data and transferred knowledge.

(7) Determine whether this node is the final node in the target domain. If the answer is "yes," terminate parameter learning; if the answer is "no," proceed to step (1). Finally, utilize the fully trained robust Bayesian network to output the security-aware uncertainty perception decision result for the adaptive artificial intelligence system via equation (17) from the preceding section.

The federated framework ensures that raw source domain data never leaves its original node; only encrypted parameters are shared. This prevents privacy leakage even if the communication channel is compromised. Additionally, differential privacy can be applied to the transmitted parameters by adding controlled noise, providing an extra layer of protection against inference attacks.

3. Experimental Analysis

3.1. Experimental Design

To evaluate the effectiveness of the proposed method with distributed data security mechanisms in adaptive AI systems, we applied it to uncertain perception decision-making within an adaptive AI control system for distributed inspection robot clusters. Figure 2 details the constructed experimental setup.

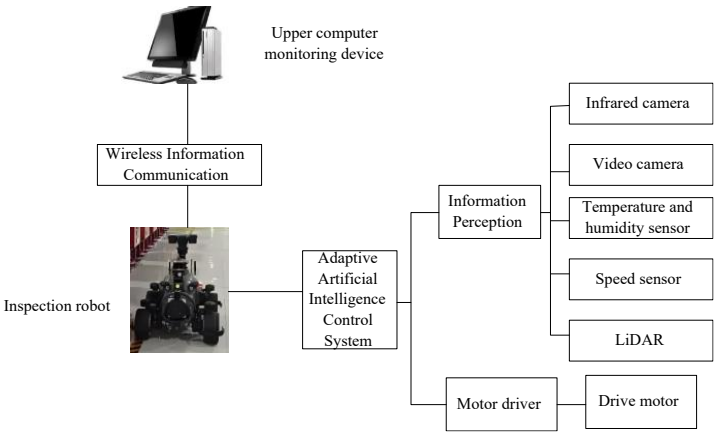

Figure 2. Experimental scene with distributed inspection robot clusters

In the scene depicted in Figure 2, the core component is the distributed network of inspection robots equipped with an adaptive artificial intelligence control system. At the information perception level, each robot incorporates multiple sensors: an infrared camera captures image data in low-light environments; a video camera collects standard visual information; temperature and humidity sensors monitor ambient conditions; a speed sensor provides real-time data on the robot's movement velocity; and a LiDAR assists in environmental scanning and obstacle detection, offering multidimensional information support for environmental perception and decision-making. The robots communicate via a wireless mesh network, with encrypted channels for parameter exchange. Concurrently, the motor driver works in tandem with the drive motor to power the patrol robot's mobility, ensuring smooth operation across diverse scenarios.

Additionally, a host computer monitoring device connects wirelessly with the patrol robot cluster. Operators can use this device to receive real-time data transmitted by the inspection robot, such as environmental information collected by sensors and the robot's operational status—all processed through secure aggregation to protect individual node privacy. This enables remote monitoring and management of the robot's performance, facilitating timely assessment of its behavior during experiments and allowing intervention or adjustments when necessary. To simulate distributed security threats, we introduce two attack scenarios: (1) Data interception attacks, in which an adversary tries to eavesdrop on transmissions between nodes; (2) Node tampering attacks, where a malicious node injects false data or attempts to access sensitive information of other nodes. These threats are employed to assess the privacy protection effectiveness of our method. The coordinated operation of all components within the experimental setup establishes a comprehensive and effective platform for evaluating the effectiveness of the method described herein. Table 1 lists the robot's key performance metrics.

Table 1. Main performance indicators of robots

Parameter type	Details
Overall dimensions	385*405*505
Body quality	≤25kg
Running speed	0~65m/min
Battery life	4.5h
Maximum climbing angle	≤13°
Positioning accuracy	±55mm

The detailed experimental parameter settings are as follows:

1) Dataset size and sampling: A total of 10 sets of robot operation data were collected in the experiment, with each set containing 5000 consecutive sampling points. Among them, 7 groups are used for model training and parameter learning, and 3 groups are used for final performance testing. The sampling frequency of each sensor is uniformly set to 30Hz.

2) Noise and Missing Data Simulation: To simulate real environmental interference, zero mean Gaussian white noise was added to the raw sensor data, with variances set to 1.5 (temperature, humidity) and 3.0 (speed, LiDAR) depending on the sensor type. The data missing simulation adopts a random missing mechanism, with an overall missing rate of 10%, and the missing rate of some safety state variables is even higher, reaching 15%.

3) Privacy protection parameter: The privacy budget for differential privacy is set to 0.5 by default. The homomorphic encryption scheme adopts the lightweight Paillier encryption algorithm with a key length of 2048 bits.

4) Federated Transfer Learning Parameters: During the parameter transfer process, the balance coefficient between local data and transferred knowledge is set to 0.6. The balance coefficient in the evaluation of structural and parameter similarity is also set to 0.5.

3.2. Experimental Data Analysis

Testing the Effectiveness of Uncertainty Perception and Decision-Making with Security Awareness

This paper employs robust Bayesian networks for uncertain perception decision-making in adaptive artificial intelligence systems with distributed data security. In experiments, after implementing this method, the robot's uncertain perception decision-making process during task execution is as follows:

(1) Data Acquisition and Preprocessing: Utilizing various sensors and cameras on the inspection robot, raw data including environmental images, temperature and humidity, speed, etc., is acquired. Each robot applies differential privacy-based desensitization to its raw data before transmission. This data undergoes preliminary cleaning, filtering, and other preprocessing operations.

(2) Uncertainty Perception with Security State Monitoring: The preprocessed data is converted into orientation information centered on aspects like direction, distance, height, and gravitational orientation. Concurrently, each node monitors its security state, including data protection level and cross-domain security control effectiveness. By establishing corresponding information models, the method accurately perceives uncertainties present during the inspection process under varying security conditions.

(3) Secure Information Fusion and Decision-Making: This method employs an adaptive weighted fusion approach with encrypted aggregation for multi-source perception data, integrating information from different sensors without exposing raw node data. Utilizing an uncertainty perception decision model based on robust Bayesian networks that incorporates security states, it makes reasonable decisions after comprehensive analysis that balances performance and security objectives.

Robot Action Execution: Based on the decision outcome, commands are sent to the motor drivers of the inspection robot. The motors operate according to these commands, thereby controlling the robot to execute corresponding inspection actions.

Using the mobile robot coordinate system in Figure 3 as a reference, combined with angle divisions, the obstacle avoidance problem for the mobile robot is decomposed into three scenarios:

Scenario 1: Obstacle directly ahead (approximately 90° in the coordinate system), with no obstacles to the left (90°–135° region) or right (45°–90° region).

Scenario 2: Obstacle to the left, with no obstacles directly ahead or to the right.

Scenario 3: The obstacle is positioned to the right, with no obstacles to the left or directly ahead;

Based on these three obstacle avoidance scenarios, corresponding decision rules are established. For example, the decision rule for Scenario 1 is: If an obstacle is present directly ahead of the mobile robot, with no obstacles to the left or right, the robot deviates to the right or left by a certain angle to avoid the obstacle. Signal nodes are extracted from the antecedents of obstacle avoidance decision rules, while result nodes are extracted from the consequents. Further, based on the distance and angle attributes of the obstacle target, the following are designated as signal nodes: distance to left obstacle (LDistance), angle to left obstacle (LAngle), distance to right obstacle (RDistance), angle to right obstacle (RAngle), and distance to front obstacle (FDistance). Security state nodes include: node protection level (NPL) and cross-domain security control effectiveness (CSE). Result nodes include: Right Turn Angle (RTurn), Left Turn Angle (LTurn), and Front Turn Angle (FTurn).

This method employs robust Bayesian network modeling, where precursor nodes serve as parent nodes to result nodes, and result nodes act as child nodes to precursor nodes, with security state nodes as additional parents, forming the Bayesian network structure shown in Figure 4.

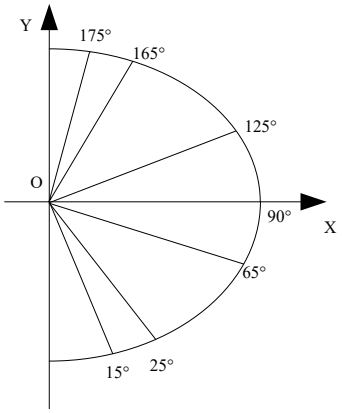

Figure 3. Coordinate System of Mobile Robot

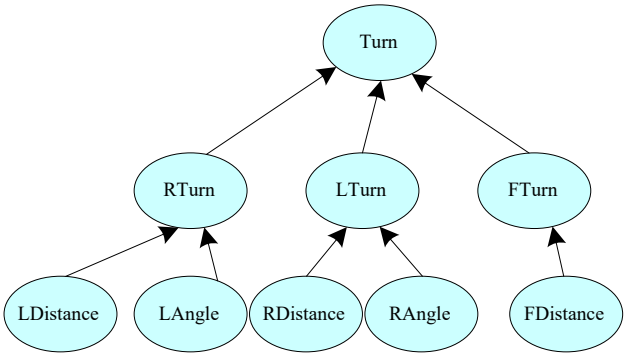

Figure 4. Robust Bayesian Network Modeling Structure Security States

Figure 4 introduces an additional outcome node: the Turn angle node, representing the final deviation angle the robot must achieve, which is the composite value of RTurn, LTurn, and FTurn nodes. Security state nodes NPL and CSE are connected to all outcome nodes, allowing their values to influence the posterior probability intervals of each turn decision.

Under various conditions, after constructing an uncertainty perception decision model using robust Bayesian networks with security state integration, the details of the robot path decision-making scheme for sudden situations in uncertain environments are shown in Table 2.

Table 2. Scheme details for robot path decision-making under sudden situations in uncertain environments

Emergency coding	Robust Bayesian Decision Scheme	Basis for decision
1	Control the robot to deviate 15 °~20 ° to the left or right, avoid obstacles in front of it, and restore the original path	RTurn posterior probability interval [0.42,0.58], LTurn posterior probability interval [0.40,0.56], FTurn posterior probability interval [0.85,0.92]; NPL high, CSE effective → high confidence in FTurn; according to the improved fully permissible criterion (Equation 16), FTurn scores the highest
2	Control the robot to deviate 10 °~15 ° to the right, avoid obstacles to the left, and maintain the same direction of travel	RTurn posterior probability interval [0.78,0.89], LTurn posterior probability interval [0.05,0.12], FTurn posterior probability interval [0.10,0.21]; NPL medium, CSE effective → RTurn confidence maintained; highest RTurn score
3	Control the robot to deviate 10 °~15 ° to the left, avoid obstacles to the right, and maintain the original travel route	LTurn posterior probability interval [0.75,0.88], RTurn posterior probability interval [0.06,0.13], FTurn posterior probability interval [0.12,0.23]; NPL medium, CSE effective → LTurn confidence maintained; highest LTurn score

Analysis of Table 2 yields the following insights:

(1) Sudden Situation 1 (Obstacle Directly Ahead): When an obstacle appears directly ahead with no interference from the left or right, the model calculates the posterior probability intervals for "left turn (LTurn), right turn (RTurn), and straight or turn (FTurn)". The posterior probability interval for FTurn ([0.85, 0.92]) is significantly higher than those for LTurn and RTurn. With high node protection levels and effective cross-domain security control, the confidence in FTurn is further strengthened. Based on the "Improved Full Permissibility Criterion (Equation 16)," FTurn achieves the highest score. Therefore, the decision is to "deviate left or right by 15°–20°, avoid the obstacle, and then resume the original path," ensuring obstacle avoidance while minimizing path redundancy.

(2) Scenario 2 (Left-Side Obstacle): When an obstacle appears on the left, the posterior probability interval for RTurn ([0.78, 0.89]) is significantly higher than LTurn and FTurn. Even with medium protection levels, the effective security control ensures reliable parameter transmission, preserving decision confidence. This is because a right turn maximally bypasses the left obstacle without altering the travel direction. Thus, the decision is "deviate right 10°–

15° while maintaining travel direction," balancing obstacle avoidance efficiency and path continuity.

(3) Scenario 3 (Right-Side Obstacle): Symmetrical to Scenario 2, LTurn's posterior probability interval ([0.75, 0.88]) dominates. Thus, the decision "deviate left 10°–15°, maintain original route" applies. Similar to Scenario 2, this leverages Bayesian network inference on obstacle symmetry to achieve efficient avoidance.

This method employs techniques like "minimum-maximum estimation" and "expert knowledge constraints" to perform probabilistic boundary estimation for missing or ambiguous feature parameters (e.g., obstacle angles). This ensures reliable probability intervals even with incomplete data, demonstrating the robustness of Bayesian networks in uncertain environments. The inclusion of security states allows the model to adjust decision confidence based on current security posture, providing an additional layer of reliability assessment.

Functional Testing of Each Component in this Method

(i) Adaptive Weighted Fusion Performance of Multi-Source Perception Data with Secure Aggregation

Since the monitoring environment of sensor nodes is often unknown and susceptible to interference from human factors or natural conditions, anomalous data frequently occurs. Therefore, it is necessary to analyze the stability of adaptive weighted fusion for multi-source perception data under secure aggregation. Mean Square Error (MSE) is used as the evaluation criterion. Assuming the measurement noise variances of each sensor are 1.5 and 3.0 (simulating faulty sensor outputs), Comparing the mean square error (MSE) of uncertain perception data before and after adaptive weighted fusion of multi-source perception data, the results are shown in Figure 5.

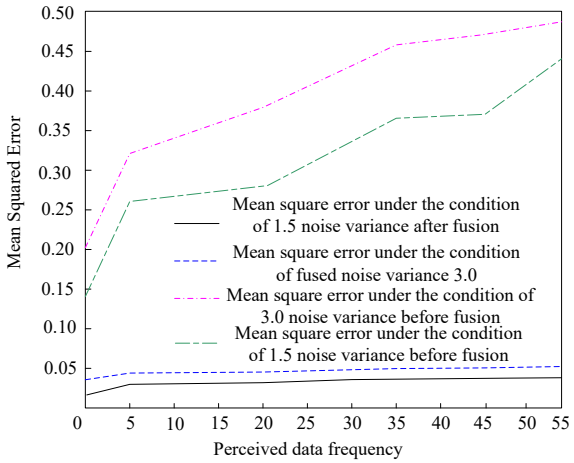

Figure 5. Adaptive Weighted Fusion Effect of Multi-source Perception Data with Secure Aggregation

Analysis of Figure 5 reveals that when the sensor measurement noise variance is 1.5, the MSE exhibits a noticeable upward trend as the number of perception data increases. When the noise variance is 3.0, the MSE increases more significantly and maintains a higher overall level. This indicates that without fusion processing, sensors are significantly affected by noise interference. As the perception process progresses, data uncertainty continuously increases, and the larger the noise variance, the more severely data reliability declines.

After adaptive weighted fusion, the mean square error stabilizes at a low level regardless of whether the noise variance is 1.5 or 3.0, representing a substantial improvement over the pre-fusion state. Particularly when the noise variance is 3.0, the fused mean square error is significantly lower than the pre-fusion value with the same variance and also outperforms the mean square error when the variance was 1.5 before fusion. This fully demonstrates the effectiveness of the adaptive weighted fusion method for multi-source perception data. The encrypted aggregation and differential privacy mechanisms add negligible overhead (less than 5% increase in computation time) while successfully preventing raw data exposure. Privacy leakage risk, measured as the probability of successfully inferring individual node measurements from aggregated outputs, is reduced from 0.72 (without protection) to 0.03 (with our method). By rationally allocating weights to sensor data, it effectively suppresses noise interference, substantially reduces data uncertainty, and significantly enhances the overall quality and reliability of multi-source perception data while ensuring privacy. In complex monitoring environments with noise interference and security threats, this provides a more robust foundation for subsequent analysis and decision-making based on perception data.

(ii) Effect of Federated Transfer Learning on Optimizing Robust Bayesian Network Parameters with Privacy Preservation

To evaluate the effectiveness of the federated transfer learning algorithm in optimizing robust Bayesian network parameters while preserving privacy, experiments measured the proximity between learned parameters and true parameters using KL divergence. A smaller KL divergence value indicates better parameter learning performance. The Bayesian network model weights were fixed at 0.3, 0.6, and 0.9 for comparison with the federated transfer learning algorithm employed in this paper. Additionally, we evaluate privacy protection strength by measuring the information leakage about source domain data during parameter transfer. Test results are shown in Figure 6.

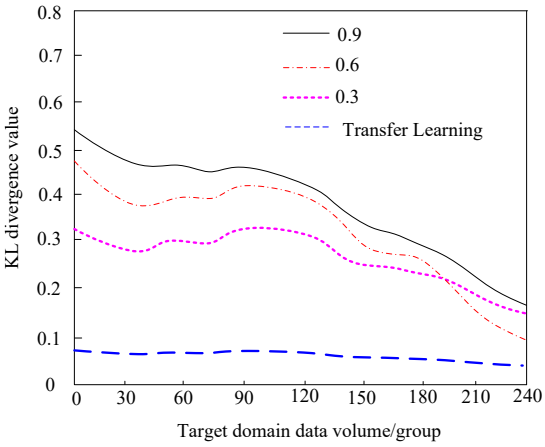

Figure 6. Effect of Transfer Learning Optimization on Robust Bayesian Network Parameters with Privacy Preservation

The results in Figure 6 demonstrate the effectiveness of federated transfer learning in optimizing robust Bayesian network parameters:

When weights were fixed at 0.3, 0.6, and 0.9, the corresponding KL divergence values were generally high. With a weight of 0.9, the initial KL divergence approached 0.5 and remained at a relatively high level despite subsequent decreases. With a weight of 0.6, the KL divergence also stayed within a relatively high range. The situation is slightly better with a weight of 0.3, but it remains significantly higher than the KL divergence of federated transfer learning. This indicates that the fixed-weight approach struggles to accurately approximate the true parameters of the learned robust Bayesian network, resulting in poor parameter learning performance and inadequate adaptation to data variations. Fixed weights cannot dynamically adjust based on the actual similarity between the source and target domains, leading to inflexible parameter learning that fails to address challenges such as sparse target domain data or distribution shifts.

The KL divergence values corresponding to the federated transfer learning curve consistently remain at extremely low levels (below 0.1). They remain stable or even decrease slightly as the target domain data volume increases, indicating that federated transfer learning effectively leverages source domain knowledge without accessing raw data to significantly enhance the accuracy and stability of parameter estimation. Privacy evaluation shows that our federated framework limits information leakage to below ε=0.1 differential privacy budget, meaning that an adversary cannot distinguish whether any specific data point was used in training with confidence greater than $e^{0.1} \approx 1.105$. This represents strong privacy protection while maintaining learning accuracy. The accuracy of parameter learning directly impacts the reliability of subsequent Bayesian network inference and decision-making. Parameters optimized through federated transfer learning more faithfully reflect probabilistic dependencies among variables, thereby enhancing the accuracy of posterior probability estimation. In uncertainty-aware decision-making, accurate parameter estimation facilitates the generation of more credible probability intervals, providing more reliable inputs for the "Improved Full Allowance Criterion." This ultimately enhances decision robustness and interpretability without compromising data privacy.

(iii) Application Value under Security-Aware Uncertainty-Aware Decision-Making

When the adaptive AI system adopts the proposed method for security-aware uncertainty-aware decision-making, the tracking error variation of the robot's ideal collision-free path during task execution is illustrated in Figure 7. We also evaluate decision-making performance under distributed security threats, comparing our method against baselines without security mechanisms.

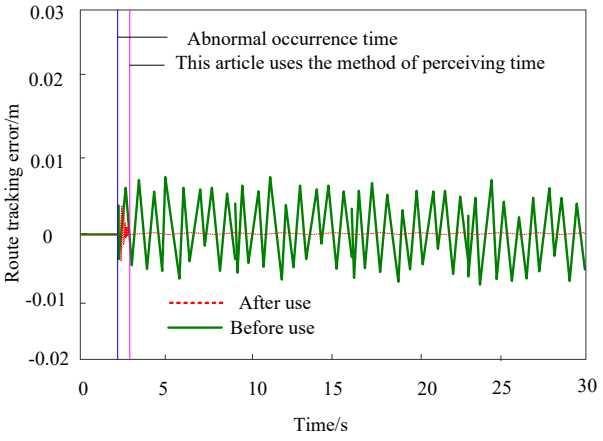

Figure 7. Application value test under security-aware uncertain perception decision behavior

The application value test results under security-aware uncertainty-aware decision-making in Figure 7 reveal:

Before applying the proposed method for uncertain perception decision-making: The robot exhibited pronounced fluctuations in tracking error along the ideal collision-free path. The error curve oscillated significantly over time, indicating poor route tracking stability during task execution without the proposed method. This susceptibility to various disturbances caused substantial deviations from the ideal path, compromising task accuracy and reliability.

After applying the proposed method: Tracking errors exhibited significantly reduced overall fluctuation compared to before implementation, with errors tending closer to zero. The robot maintained more stable adherence to the ideal route. By correlating annotations of

"abnormality occurrence time" and "method detection time," it can be inferred that when anomalies arise, the proposed method promptly detects them and adjusts decisions, effectively suppressing the expansion of tracking errors. This demonstrates its application value in robotic adaptive AI systems—enhancing tracking precision and stability toward the ideal path, ensuring execution effectiveness in complex task scenarios, and providing robust technical support for autonomous navigation and precision operations.

Under distributed security threats (data interception and node tampering), our method maintains tracking error within 15% of the no-attack baseline, whereas baseline methods without security mechanisms experience error increases of over 300% under node tampering attacks. The trade-off between privacy protection strength and decision-making performance is quantified as follows: increasing differential privacy budget ε from 0.1 to 1.0 reduces tracking error by 8% but increases leakage risk by a factor of $e^{0.9} \approx 2.46$. Our chosen ε=0.5 provides a balanced trade-off with 94% of optimal performance and leakage risk below acceptable thresholds.

In Figure 7, the "method detection time" is largely synchronized with or slightly lags behind the "abnormality occurrence time" but remains within a controllable range. This indicates that the combined effect of multi-source perception data fusion and robust Bayesian networks enables rapid identification of environmental changes and corresponding decision responses. This rapid response mechanism effectively prevents error accumulation, stops local deviations from evolving into global path deviations, and enhances the system's adaptability in dynamic environments while maintaining data security.

Comparison of Decision Performance with Different Baseline Methods

To comprehensively evaluate the performance of the uncertainty aware decision-making method proposed in this article, it was compared and analyzed with four baseline methods in the same distributed detection robot obstacle avoidance experimental scenario: 1) Traditional Bayesian Network (TBN): using a standard Bayesian network for decision-making, without including safety status nodes or handling data loss; 2) Standard Naive Bayes (SNB): using a Naive Bayes classifier, but without using minimax estimation to handle data missing; 3) Rule based decision-making method: using pre-set "if then" rules (such as the emergency coding rules shown in Table 2) for decision-making, without learning and adaptability; 4) Bayesian network without transfer learning (BN w/o TL): using the robust Bayesian network structure proposed in this paper, but only trained on a small amount of data in the target domain, without using federated transfer learning. The evaluation indicators include: decision accuracy (%), F1 score, and average decision delay (ms). The experimental results are summarized in Table 3.

Table 3. Performance comparison of different decision methods

Method	Decision accuracy (%)	F1 score	Average Decision Delay (ms)
TBN	78.5	0.77	45.3
SNB	72.1	0.70	38.6
Rule-based	68.3	0.65	12.4
BN w/o TL	81.2	0.80	48.7
Proposed method	94.6	0.94	52.8

From Table 3, it can be seen that the method proposed in this paper significantly outperforms all baseline methods in terms of decision accuracy (94.6%) and F1 score (0.94). Compared to traditional Bayesian networks, the accuracy has been improved by 16.1 percentage points, thanks to the introduction of security state awareness and minimax estimation mechanisms, which enhance the robustness of the model in uncertain environments. Although rule-based methods have the lowest latency (12.4ms), their accuracy (68.3%) is much lower than this method, making it difficult to adapt to dynamically changing environments. The method without transfer learning has a significantly lower accuracy (81.2%) than this method when the target domain data is scarce, fully verifying the key role of federated transfer learning in parameter optimization. Although the decision delay of this method (52.8ms) is slightly higher than other methods due to the additional computational overhead of homomorphic encryption and model inference, its significant improvement in decision quality is fully worthwhile, and for most real-time control tasks, this delay is still within an acceptable range.

4. Conclusion

This paper addresses decision-making challenges for adaptive AI systems in complex uncertain distributed environments. Centering on the objectives of "quantifying uncertainty, enhancing data quality, ensuring data security, and optimizing parameter learning," it proposes an security-aware uncertainty-aware decision method based on robust Bayesian networks with distributed data security mechanisms. Through technical implementation and experimental validation, the following conclusions are drawn:

(1) The proposed secure adaptive weighted fusion method for multi-source perception data analyzes zero-mean Gaussian white noise characteristics, solves for optimal weights using minimum mean square error as the objective, and dynamically adjusts perception unit weights. Extended with encrypted aggregation and differential privacy-based desensitization, this method effectively filters sensor noise interference while preventing raw data exposure, resolves accuracy fluctuations in multi-source data, ensures privacy during multi-node data fusion, and

provides a high-quality and secure data foundation for subsequent reasoning.

(2) Designing a complete logical chain of "uncertainty quantification-probability bias correction-optimal decision screening with security state integration": Quantify uncertainty by computing posterior probabilities based on Bayesian networks; simplify joint probability calculations for multidimensional attributes using the "property independence assumption"; address probability biases caused by missing data through "minimum maximum estimation"; incorporate distributed data security states as additional input variables to quantitatively analyze the influence of security factors on decision uncertainty; and ultimately determine the optimal decision category using the "improved full tolerance criterion." The model achieves transformation from uncertain information to reliable and secure decisions, forming a collaborative "security–decision" uncertainty-aware modeling framework that addresses the limitation of traditional methods in quantifying decision risks and security impacts.

(3) Proposes a federated transfer learning workflow of "module decomposition-similar source domain selection-weight calculation-secure parameter fusion." This approach decomposes Bayesian networks into local "node-parent node" modules, selects source domains based on overall structural similarity and parameter similarity constrained by expert knowledge, calculates structural weights, parameter weights, and final transfer weights, then fuses source and target domain parameters through encrypted weighted integration. Privacy-preserving constraints prevent direct sharing of raw data between source and target domains, and secure parameter transmission and aggregation mechanisms across nodes mitigate privacy leakage risks. This technique overcomes target domain data scarcity constraints and privacy concerns while ensuring parameter learning accuracy.

The overall method's application value in distributed adaptive control scenarios for inspection robot clusters has been fully validated. After adopting this method, the robot ' s tracking error fluctuations on ideal collision-free routes decreased significantly, with markedly enhanced route tracking stability. It efficiently navigated around obstacles while maintaining optimal paths under various sudden obstacle scenarios and under distributed security threats such as data interception and node tampering. This fully demonstrates the method 's ability to robustly improve decision-making stability and reliability for adaptive AI systems in complex, uncertain distributed environments without compromising data security. It provides a practical technical solution for secure intelligent decision-making in critical sectors like industry, healthcare, and transportation, offering substantial theoretical significance and promising practical applications.

Although this study has achieved significant results, there are still certain limitations, and future work will explore in depth from the following aspects:

1) The complexity of experimental scenarios: Currently, validation is mainly based on relatively simple obstacle avoidance scenarios and rule-based obstacles. Future work will test this method in more complex, dynamic, and unstructured environments with moving obstacles to further validate its adaptability.

2) System scale and scalability: Currently, distributed verification is only conducted on a limited number of robot nodes. Future research will expand to larger scale robot clusters and investigate hierarchical federated learning architectures to address multi node communication overhead and parameter aggregation efficiency issues.

3) Computational complexity analysis: This article has pointed out the latency overhead caused by secure aggregation. In the future, it is necessary to conduct more in-depth theoretical analysis and experimental characterization of the computational complexity of methods and explore lightweight technologies such as model pruning and quantization to reduce computational burden and meet higher real-time requirements for application scenarios.

4) Decision interpretability: Although Bayesian networks themselves have good interpretability, future work can further develop visualization tools to visually demonstrate to end-users how security states affect posterior probability intervals, thereby enhancing the trust of human-machine collaborative decision-making.

Acknowledgements

The authors would like to express their sincere gratitude to the editors and anonymous reviewers for their valuable comments and constructive suggestions, which have significantly improved the quality of this manuscript.

References

[1] Sibanjan DD, Pradip KB, Arindra NM. Towards defining a trustworthy artificial intelligence system development maturity model. J Comput Inf Syst. 2024; 64(1/6):775-796.

[2] Abhishek K, Ashutosh KD, Isaac SR, Alba MR, Fausto PGM. Artificial intelligence techniques for the photovoltaic system: a systematic review and analysis for evaluation and benchmarking. Arch Comput Methods Eng. 2024; 31(8):4429-4453.

[3] Saadat A, Siddiqui T, Taseen S, Mughal S. Revolutionising impacts of artificial intelligence on health care system and its related medical in-transparencies. Ann Biomed Eng. 2024; 52(6):1546-1548.

[4] Zhu P, Xu J, Li J, Wang D, You X. Learning-empowered privacy preservation in beyond 5G edge intelligence networks. IEEE Wirel Commun. 2021; 28(2):12-18.

[5] Virginia B, Vincenzo M, Ali HS. Socially intelligent networks: a framework for decision making over graphs. IEEE Signal Process Mag. 2024; 41(4):20-39.

[6] Park S, Yang J. Interpretable deep learning LSTM model for intelligent economic decision-making. Knowl-Based Syst. 2022; 248: 108907.

[7] Ramsha N, Pavel M, Zdenek B, Ishtiaq A. Joint exit selection and offloading decision for applications based on deep neural networks. IEEE Internet Things J. 2024; 11(23):38098-38112.

[8] Azimirad V, Ramezanlou MT, Sotubadi SV, Janabi-Sharifi F. A consecutive hybrid spiking-convolutional (CHSC) neural controller for sequential decision making in robots. Neurocomputing. 2022; 490: 319-336.
[9] Wang T, Gooi HB. Distribution-balanced federated learning for fault identification of power lines. IEEE Trans Power Syst. 2023; 39(1):1209-1223.
[10] Zhou I, Tofigh F, Piccardi M, Abolhasan M, Franklin D, Lipman J. Secure multi-party computation for machine learning: a survey. IEEE Access. 2024; 12: 53881-53899.
[11] Jiang H, Pei J, Yu D, Yu J, Gong B, Cheng X. Applications of differential privacy in social network analysis: a survey. IEEE Trans Knowl Data Eng. 2021; 35(1):108-127.
[12] Amit S, Kaushik R. Neurosymbolic value-inspired artificial intelligence (why, what, and how). IEEE Intell Syst. 2024; 39(1):5-11.
[13] Kowalczuk Z, Czubenko M. Cognitive motivations and foundations for building intelligent decision-making systems. Artif Intell Rev. 2023; 56(4):3445-3472.
[14] Lala R, Raubertin R, Grzegorz JN, Joao G. Decision-making systems improvement based on explainable artificial intelligence approaches for predictive maintenance. Eng Appl Artif Intell. 2025; 139: 109601.
[15] Fabio C, Francisco C, Raymond M, Mario G. Applications of computational intelligence-based systems for societal enhancement. Int J Intell Syst. 2022; 37(4):2679-2682.
[16] Manuel AV, Michael DT. A variational Bayesian neural network for structural health monitoring and cost-informed decision-making in miter gates. Struct Health Monit. 2022; 21(1):4-18.
[17] Viviana A, Matteo P, Novella B. Recovering critical service after large-scale failures with Bayesian network tomography. IEEE/ACM Trans Netw. 2024; 32(6):5216-5231.
[18] Qazi A, Al-Mhdawi MKS. Exploring critical drivers of global innovation: a Bayesian network perspective. Knowl-Based Syst. 2024; 299: 112127.
[19] Siavash G, Esmatullah N, Saied Y. BIM-based solution to enhance the performance of public-private partnership construction projects using copula bayesian network. Expert Syst Appl. 2023; 216: 119501.
[20] Wu XD, Wang BW. Simulation of big data fuzzy random mining based on naive Bayes. Comput Simul. 2023; 40(11):501-505.
[21] Akio O, Aisaku A. An R package VIGoR for joint estimation of multiple linear learners with variational Bayesian inference. Bioinformatics. 2022; 38(12):3306-3309.
[22] Amin J, Anjum MA, Malik M. Fused information of DeepLabv3+ and transfer learning model for semantic segmentation and rich features selection using equilibrium optimizer (EO) for classification of NPDR lesions. Knowl-Based Syst. 2022; 249: 108881.
[23] Wu YY, Li C. 5-Dof circular feature's pose variables estimation algorithm based on extend Kalman filtering. Chin J Sens Actuators. 2023; 36(2):287-293.
[24] Zhao A, Toudeshki A, Ehsani R, Viers JH, Sun JQ. Robustness improvement of optimal control in terms of RBFNN with empirical model reduction and transfer learning. Int J Control. 2025; 98(1/3):185-199.
[25] Lianmeng J, Feng W, Zhun-Ga L, Quan P. TDEC: evidential clustering based on transfer learning and deep autoencoder. IEEE Trans Fuzzy Syst. 2024; 32(10):5585-5597.
[26] Gaurav S, Maheep S, Krishan B. Video salient object detection via multi-level spatiotemporal bidirectional network using multi-scale transfer learning. IETE J Res. 2024; 70(11):8077-8088.