Int j simul model 23 (2024) 4, 716-727
Original scientific paper

OPTIMIZING HUMAN-MACHINE SYSTEMS IN AUTOMATED ENVIRONMENTS

Xing, H. R.
Art, Design & Architecture, University of New South Wales, Sydney, Australia
E-Mail: xing3345183562@163.com

Abstract
With the rapid advancement of industrial automation technologies, human-machine collaboration systems have become critical for enhancing productivity and safety in highly automated environments. However, current human-machine collaboration systems still face numerous challenges in practical applications, especially in dynamic and complex work scenarios, ensuring safety and efficiency in the human-machine collaboration process lacks a systematic solution. To address this issue, this paper proposes a braking control method based on discrete-time model prediction and an adaptive human-machine safety distance prediction model using a multilayer perceptron (MLP) network. By modelling and predicting the system's dynamic data, this research aims to improve the efficiency and safety of human-machine collaboration, providing theoretical support and practical guidance for the design and management of automated systems.
(Received in August 2024, accepted in November 2024. This paper was with the author 1 month for 1 revision.)

Key Words: Highly Automated Environments, Human-Machine Collaboration, Discrete-Time Model, Multilayer Perceptron (MLP), Safety Distance Prediction, Braking Control

1. INTRODUCTION

With the rapid development of Industry 4.0, automation technology has gradually penetrated into various manufacturing and service industries, and human-machine collaboration systems are playing an increasingly important role in the production process [1-5]. Human-machine collaboration in highly automated environments can greatly improve production efficiency and quality [6-9]. However, this also brings new challenges, such as how to ensure the safety and efficiency of human-machine collaboration systems, especially in complex and dynamic work environments. Effective risk management and system optimization strategies are needed to cope with potential dangers and uncertainties [10, 11]. Therefore, the study of optimization and risk management in human-machine collaboration systems in highly automated environments is particularly critical.

The significance of this research lies in that, through the effective optimization of human-machine collaboration systems, it is possible to improve productivity while reducing the risk of human operational errors and minimizing safety hazards between machines and personnel [12-15]. More importantly, in the rapidly developing and fast-evolving field of automation, how to achieve intelligent collaboration between humans and machines, and how to ensure personnel safety in complex working conditions, are key issues in current research [16, 17]. By thoroughly exploring these issues, safer and more efficient human-machine collaboration models can be provided for future intelligent manufacturing systems.

However, existing research methods often focus on human-machine interaction under static conditions, ignoring the variability and complexity of dynamic work scenarios in highly automated environments. Particularly in practical operations, there is a lack of effective prediction and management of safety distances and braking control in human-machine collaboration [18-20]. In addition, common methods often fail to effectively combine multi-level and multi-dimensional data, making it difficult to comprehensively and accurately evaluate and optimize the safety and stability of human-machine collaboration [21-23].

This paper aims to fill the above gaps through research in two key directions. First, a braking control method based on discrete-time model prediction for human-machine collaboration systems in highly automated environments is proposed to improve the accuracy and safety of system response. Second, an adaptive human-machine safety distance prediction model based on a MLP network is developed, which can adjust the safety distance between humans and machines according to the dynamic changes of different scenarios, ensuring smooth and safe collaboration processes. This research not only provides theoretical support for the optimization of current automation systems but also holds important value for safety management in practical industrial applications.

2. BRAKING CONTROL METHOD FOR HUMAN-MACHINE COLLABORATION SYSTEMS IN HIGHLY AUTOMATED ENVIRONMENTS

In highly automated environments, the complexity of human-machine collaboration is mainly reflected in how to ensure that the collaborative work between robotic arms and humans can maintain efficiency while avoiding potential safety accidents. Based on this, braking control has become one of the core issues to ensure safe and efficient collaboration. In dynamic working conditions, the physical characteristics of the robotic arm, such as motion speed, angle, and angular velocity, directly affect its response speed and the accuracy of its motion trajectory. Therefore, this paper studies the robotic arm braking method based on discrete-time model prediction. The main reason is that the motion process of the robotic arm is continuous and complex; however, an efficient control system needs to discretize this continuous motion to enable precise calculation and control in computer-based systems. The discrete-time model can transform the continuous-time motion into a series of discrete moments, and through the discretized time steps, predict the motion state of the robotic arm, thus providing a precise basis for braking decisions. At the same time, the motion speed of the robotic arm is directly related to its response capability in emergency situations. If the speed is too fast, the braking distance will be extended, thereby increasing the risk of collision. Therefore, in braking control, it is necessary to monitor the speed of the robotic arm in real-time and dynamically adjust the braking strategy based on its current speed, ensuring that it can safely stop in the shortest time possible. In addition, angle and angular velocity are also important factors affecting the motion of the robotic arm. The motion of the robotic arm in space is often multi-degree-of-freedom, requiring simultaneous control of multiple joints' angles and angular velocities. Changes in angle can affect the motion direction of the robotic arm, while angular velocity determines the speed of joint rotation. Incorporating both into the braking control model can effectively prevent instability during high-speed motion of the robotic arm. The advantage of the discrete-time model also lies in its ability to meet the real-time monitoring and control requirements in highly automated environments. Through motion state prediction based on the discrete-time model, it is possible to detect the trend of the robotic arm entering a dangerous area earlier, allowing for preemptive braking measures. Especially in multi-task and dynamic interaction scenarios, early prediction of the robotic arm's motion trajectory can effectively avoid direct collisions with humans, improving the safety of human-machine collaboration. Fig. 1 shows the human-machine collaboration process in highly automated environments.

First, in the braking control process, the system state vector can be defined as $a_l(s) = [\phi, n, c]$, where ϕ represents the swing angle of the robotic arm, n is the speed of the robotic arm, and c is the angular velocity of the swing angle. These three parameters describe the motion state of the robotic arm and are the core variables of the braking control strategy. In a non-zero initial state, at least one of the robotic arm's speed, load swing angle, or angular velocity has a non-zero value, so it is necessary to comprehensively consider the changing trends of these three

variables to ensure that the robotic arm and load can brake safely and smoothly. There exists a moment s_f that satisfies the following relationship:

$$\varphi(s)=0, n(s)=(0), \varphi^{*}(s)=0, \forall s \geq s_f \tag{1}$$

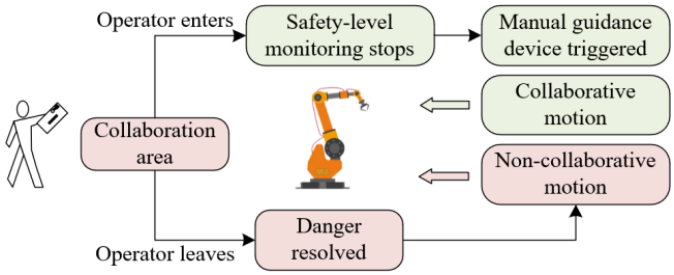


Figure 1: Human-machine collaboration process in highly automated environments.

To further describe the system's dynamics, it can be assumed that the total force acting on the system is $D=D_x+d$, where D_x is the external force, and d is the internal resistance generated during the motion of the robotic arm. Based on this assumption, the system's spatial state expression can be established, reflecting the changes in the robotic arm's position, speed, and angular velocity over time. Through this expression, we can describe the system's state at a certain moment and provide a dynamic basis for the braking control strategy. This yields:

$$\begin{cases} \dot{n}=\dfrac{lh}{L}\varphi+\dfrac{1}{L}D \\ \varphi^{**}=-\dfrac{(l+L)h}{mL}\varphi+\left(-\dfrac{1}{mL}\right)D \end{cases} \tag{2}$$

Assuming the input signal is represented by i, based on the above equation, the system's spatial state expression can be obtained:

$$\dot{a}_l=\begin{bmatrix} 0 & 0 & 1 \\ \dfrac{lh}{M} & 0 & 0 \\ -\dfrac{(l+L)h}{mL} & 0 & 0 \end{bmatrix} a_l+\begin{bmatrix} 0 \\ \dfrac{1}{L} \\ -\dfrac{1}{mL} \end{bmatrix} i, i(s)\in E \tag{3}$$

$$b=a_l \tag{4}$$

To achieve real-time control in highly automated environments, the continuous spatial state expression needs to be discretized. The discretization process converts the continuous time dimension s into a series of discrete time points s_j, where j represents the j^{th} time step. For the state vector $a_l(s)$, the discretized form can be expressed as $a_l(s_j)$, i.e., the state of the system at the j^{th} discrete moment. By further discretizing the above two equations, we can obtain the discretized system spatial state expression:

$$\begin{cases} a_l(j+1)=X_l a_l(j)+Y_l i(j) \\ b(j)=Z_l a_l(j), Z_l=U_{4\times 4} \end{cases} \tag{5}$$

Assuming the prediction time domain is represented by V_o, the control time domain is represented by V_z, and they satisfy $V_z \leq V_o$, with the sampling time represented by j_s. Further deriving the above equation gives the following result:

$$\begin{cases} a_l(j_u+V_o \mid j_u)=X_l^{V_o} a_l(j_u)+X_l^{V_o-1} Y_l i(j_u)+\cdots+X_l^{V_o-V_z} Y_l i(j_u+V_z-1) \\ b(j_u+V_o \mid j_u)=Z_l a_l(j_u+V_o \mid j_u), Z_l=U_{4\times 4} \end{cases} \tag{6}$$

When $a_l(j_s)$ is given, all system state variables between time steps $j_s + 1$ and $j_s + V_o$ can be predicted using the above equation. Assuming the system's input signal is represented by I, and the system state matrix is represented by B, the following spatial state expression can be obtained to realize the rolling time domain control of the model:

$$B = OA_l\left(j_s\right) + \Omega I$$
$$\Omega = \begin{bmatrix} Z_l Y_l & 0 & \cdots & 0 \\ Z_l X_l Y_l & Z_l Y_l & \cdots & 0 \\ \vdots & \vdots & & \\ Z_l X_l^{Y_o - 1} Y_l & Z_l X_l^{Y_o - 2} Y_l & \cdots & \cdots \end{bmatrix};\ O = \begin{bmatrix} Z_l X_l \\ Z_l X_l^2 \\ \vdots \\ Z_l X_l^{V_o} \end{bmatrix}, O \in E^{3V_o \times 3}, \Omega \in E^{3V_o \times V_z} \tag{7}$$

In a non-zero initial state, the robotic arm and load may already be in motion, possessing a certain speed, swing angle, and angular velocity. To achieve rapid braking, a reasonable reference trajectory must be designed so that the system can quickly transition from the initial state to the target state of complete stop, avoiding abrupt state changes or overshoot. By designing the reference trajectory, the robotic arm's speed, load swing angle, and angular velocity can be kept within safe ranges, ensuring smooth braking in the shortest possible time. The reference trajectory design also provides the foundation for constructing the cost function. During the braking process, the reference trajectory at different discrete time domains can provide a basis for the system's control input, thus constructing an optimal cost function to help optimize the system's braking control strategy.

The goal of the reference trajectory is to ensure that the system converges steadily to the target state at each discrete time point, while minimizing the amplitude of the load swing angle change. During this process, the robotic arm's speed and angular velocity will gradually decrease until a complete stop is achieved. Suppose the control period of the discrete points is represented by S_j, the time convergence factor by η, and the initial state of the robotic arm by $b(s_0) = [\phi(s_0)\ n(s_0)\ \dot{\phi}(s_0)]^T$, where the initial angle of the load, the initial speed of the robotic arm, and the initial angular velocity of the load are represented by $\phi(s_0)$, $n(s_0)$, and $\phi^*(s_0)$, respectively. The reference trajectory expression is as follows:

$$e\left(j\right) = b\left(s_0\right) \times \left(1 - \tanh\left(\eta j S_j\right)\right) \tag{8}$$

After generating the reference trajectory, it is necessary to update the system's spatial state expression based on the system's dynamic model. Suppose the set of all reference trajectories within the rolling time domain of the model is represented by $E(j_u)$. Using the above equation, the spatial state expression of the system at each discrete moment during the braking process can be obtained:

$$E\left(j_u\right) = \begin{bmatrix} e^S\left(j_u\right) \\ e^S\left(j_u\right) \\ \vdots \\ e^S\left(j_u + V_o - 1\right) \end{bmatrix} \tag{9}$$

Once the reference trajectory generation and state expression update are complete, the cost function can be constructed to optimize the system's braking control process. The cost function is determined by trajectory tracking error, braking force, and load swing angle constraint. The trajectory tracking error measures the difference between the actual system state and the reference trajectory, ensuring that the system state always closely follows the reference trajectory. The braking force represents the energy or braking force required by the control system during the braking process, ensuring that the braking is not only fast but also energy-efficient. The load swing angle constraint ensures that the load swing angle remains within a small range, ensuring system safety.

3. ADAPTIVE HUMAN-MACHINE SAFETY DISTANCE PREDICTION IN HIGHLY AUTOMATED ENVIRONMENTS

With the development of automation technology, human-machine collaboration systems are gradually working in highly complex environments. These environments have many uncertainties, such as the diversity of tasks, environmental changes, and the operational status of equipment, all of which affect the overall safety of the system. In these dynamic scenarios, traditional fixed safety distance algorithms may not adapt to changing working conditions, leading to potential safety risks. As a classical neural network model, the MLP has strong nonlinear mapping capabilities and adaptability, capable of handling complex input-output relationships. Therefore, this paper adopts MLP for adaptive safety distance prediction to better address these uncertainties. The discrete-time model discussed in the previous section can accurately predict the trends in the robotic arm's speed, angle, and angular velocity during motion, providing reliable input data for the MLP's safety distance prediction. Fig. 2 shows the principle of adaptive human-machine safety distance monitoring in highly automated environments.

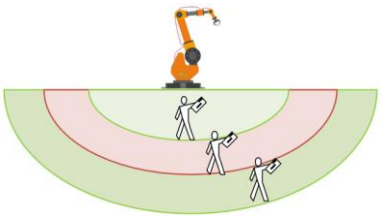


Figure 2: Adaptive human-machine safety distance monitoring in highly automated environments.

The MLP is a feedforward neural network consisting of an input layer, hidden layers, and an output layer. For adaptive human-machine safety distance prediction in highly automated environments, the specific architecture of the MLP needs to be designed according to the system's complexity and real-time requirements. Generally, a typical MLP architecture can include the following components:

(1) **Input Layer:** The input layer contains all feature vectors related to the state of the robotic arm, such as speed, angular velocity, angle, and load swing angle. These feature vectors provide a comprehensive description of the robotic arm's current state, serving as the foundational data for subsequent predictions. The design of the input layer requires standardizing these feature vectors to ensure that the input data is within the same scale, facilitating processing by the MLP network. By incorporating diverse features in the input layer, the MLP can more accurately capture the motion state of the robotic arm, thereby providing a reliable data basis for safety distance prediction.

(2) Hidden Layer: The hidden layer utilizes the nonlinear activation functions of multiple neurons to extract complex features from the input data. In highly automated environments, the relationship between the motion state of the robotic arm and the safety distance is often highly nonlinear. By employing multiple neurons and nonlinear activation functions, the hidden layer can effectively extract deep-level features from the input data. The number of hidden layers and the number of neurons in each layer need to be adjusted according to the complexity of the actual problem. Generally, 2-3 hidden layers can handle most complex problems, but in very complex environments, more layers may be necessary.

(3) Output Layer: The output layer is responsible for generating the final predicted safety distance value. In highly automated environments, the predicted safety distance needs to be dynamically adjusted based on the real-time state of the robotic arm. Therefore, the design of the output layer must consider the accuracy and real-time nature of the predicted values. Typically, a linear activation function can be employed in the output layer to ensure the

continuity and interpretability of the prediction results. The predicted results from the output layer should be compared with the actual safety distance, and adjustments to the MLP model's weights and biases can be made through a feedback mechanism. Utilizing a gradient descent optimization algorithm can continually enhance the model's predictive capabilities and improve the accuracy of safety distance predictions.

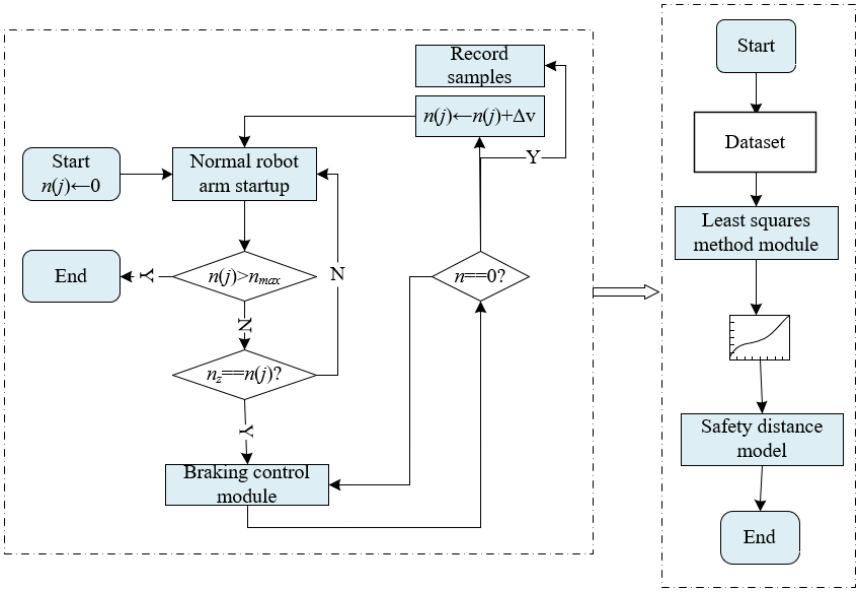


Figure 3: Offline human-machine safety distance prediction process.

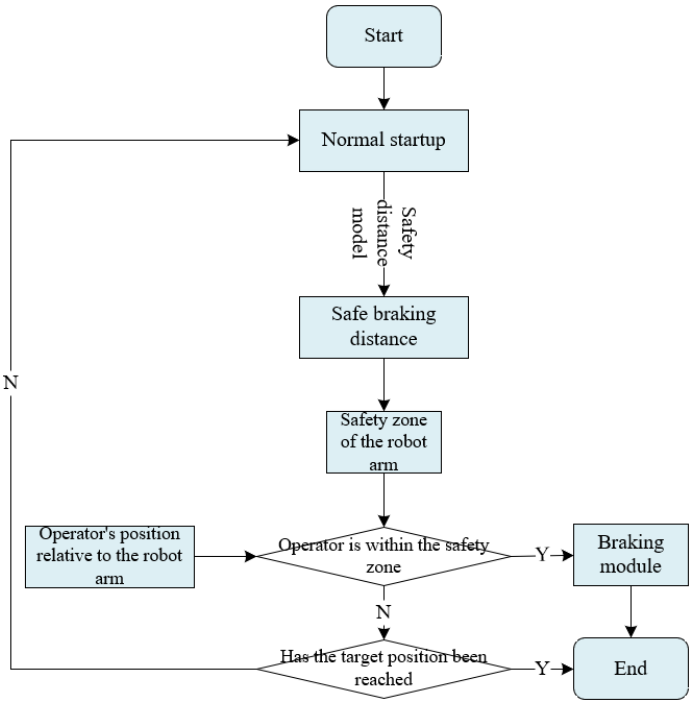


Figure 4: Online human-machine safety distance prediction process.

Figs. 3 and 4 illustrate the offline and online human-machine safety distance prediction processes utilized in practical applications. The training and application steps of the MLP network for adaptive human-machine safety distance prediction in highly automated environments can be customized and optimized according to the standard neural network training process, integrating the specific requirements of adaptive human-machine safety distance prediction. The detailed steps are as follows:

Step 1: Define input samples and network structure

Assuming the input sample is a_1, its input feature dimension is v. The feature vector corresponding to the input signal a_1 contains key information about the current state of the robotic arm or robot in the highly automated environment, such as speed, angle, position, and acceleration. In the network, the number of neurons in the input layer matches the feature dimension, meaning each input feature is connected to one neuron in the input layer. Thus, the input layer directly receives the input features and transmits them to the neurons in the hidden layer.

$$a_1 = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1v} \end{bmatrix} \tag{10}$$

Step 2: Calculate signal transmission from input layer to hidden layer

In the MLP, each input neuron is connected to every neuron in the hidden layer. Assuming there are g neurons in the hidden layer, the weight vector for the first input neuron is $q_{11} = [q_{11}, q_{12}, ..., q_{1g}]$ and its bias is y_{11}. For the first hidden layer neuron, the definitions and calculations for the weight vector and offset vector are:

$$\begin{cases} Q_1 = \begin{bmatrix} q_{11}^S & q_{12}^S & \cdots & q_{1v}^S \end{bmatrix}^S \\ y_1 = \begin{bmatrix} y_{11} & y_{12} & \cdots & y_{1g} \end{bmatrix} \end{cases} \tag{11}$$

Further, calculation formulas for the weight and offset of the hidden layers can be attained as:

$$\begin{cases} Q_2 = \begin{bmatrix} q_{21}^S & q_{22}^S & \cdots & q_{2g}^S \end{bmatrix}^S \\ y_2 = \begin{bmatrix} y_{21} & y_{22} & \cdots & y_{2w} \end{bmatrix} \end{cases} \tag{12}$$

Step 3: Nonlinear activation of hidden layer

To ensure the network's nonlinear mapping capability and prevent the MLP from degenerating into a linear model, a nonlinear activation function is applied to the hidden layer neurons. Common activation functions include Sigmoid, ReLU, and tanh functions. Assuming we choose the ReLU function, denoted as $\delta(a)$, the output results from the input layer and hidden layer can be obtained through:

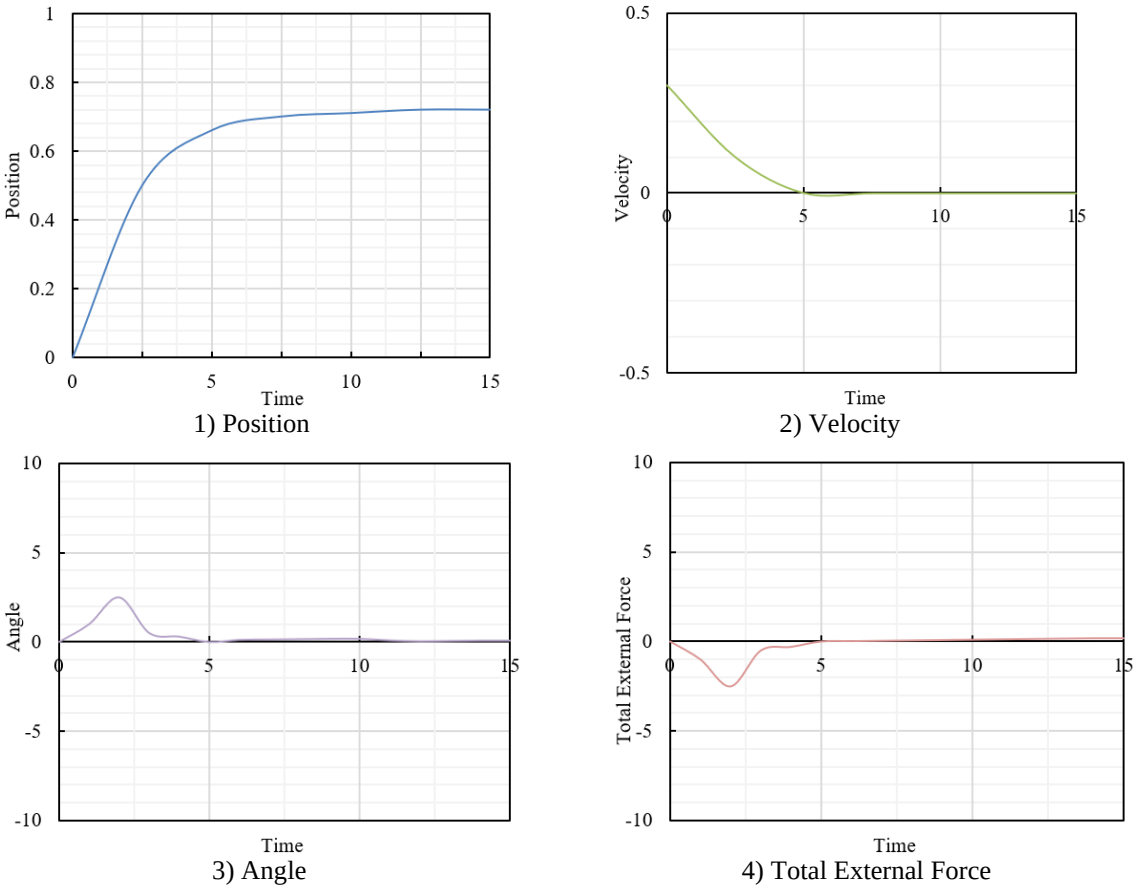
$$\begin{cases} G = \delta\left(aQ_1 + y_1\right) \\ P = GQ_2 + y_2 \end{cases} \tag{13}$$

Once training is complete, the MLP model can be used for real-time predictions of adaptive human-machine safety distance in highly automated environments. For each prediction, the status information of the robotic arm is inputted, and the predicted result for the current human-machine distance is generated through forward propagation in the MLP network. This result can be used to dynamically adjust the motion path of the robotic arm, ensuring operational safety.

4. EXPERIMENTAL RESULTS AND ANALYSIS

From the experimental results shown in Fig. 5, the changes in displacement, velocity, angle, and total external force at different time points demonstrate the effectiveness of the brake control. Between 0 and 15 seconds, the displacement gradually stabilizes, ultimately stopping at around 0.72, indicating the system's ability to quickly reduce the range of motion in the initial response phase. Meanwhile, the velocity drops rapidly from an initial value of 0.3 to close to 0 and remains nearly zero, suggesting that the system can respond promptly and decelerate effectively. Additionally, the angle shows rapid adjustments when influenced by external forces; particularly as the total external force gradually weakens, the angle stabilizes,

demonstrating that the system can maintain the stability of the robotic arm's posture while controlling the total external force. The total external force data also indicates that, when subjected to significant negative external forces at the initial phase, the system can quickly adjust until the external force returns to near zero, further ensuring system stability. Based on the above data analysis, it can be concluded that the proposed control method can achieve rapid and safe brake control under the influence of total external forces. In highly automated environments, sudden changes in external forces pose challenges to the system's stability and response speed; however, experimental data indicate that the system can quickly reduce speed and adjust angles through a reasonable adjustment mechanism when impacted by significant external forces, ensuring stability. Even when a substantial negative load is applied initially, the system can quickly adapt and control the position and angle of the robotic arm, ensuring safety and precision during the collaboration process, indicating that this control method possesses strong resistance to external force disturbances.

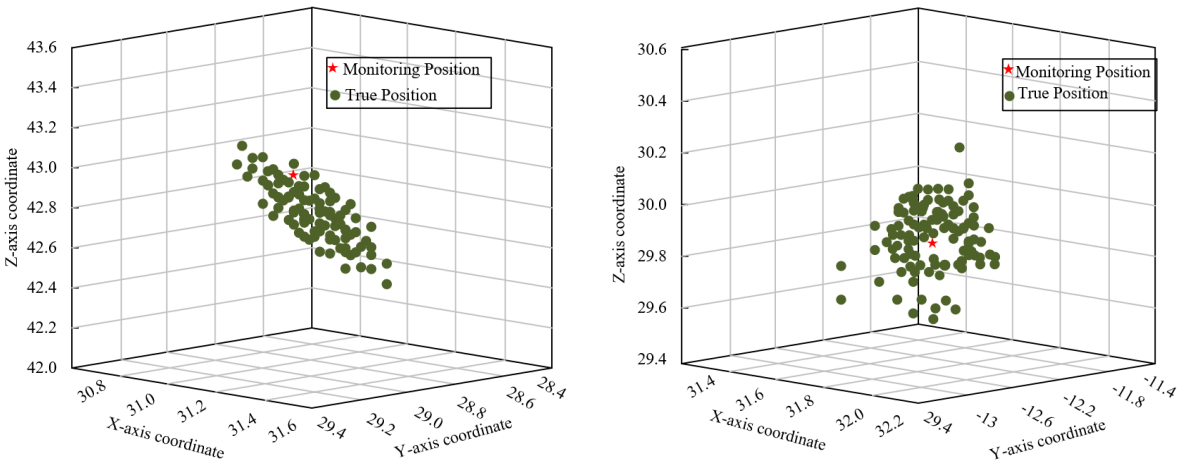


1) Position 2) Velocity

3) Angle 4) Total External Force

Figure 5: Brake control effect of human-machine collaboration system in highly automated environment.

Based on the experimental data shown in Fig. 6, the design of the reference trajectory aims to maintain a smooth transition of the robotic arm's joints and ends during motion, thereby achieving rapid braking through reasonable trajectory sampling points. Specifically, the 3D position coordinates of the robotic arm's joints and ends at different time points can be inferred from the speed, angle, and other data obtained in the experiment. For the joints, assuming the robotic arm is affected by external forces during motion, with an initial non-zero velocity, the displacement gradually stabilizes over the course of 0 to 15 seconds, the velocity decreases from 0.3 to nearly zero, and the angle experiences some fluctuations initially but gradually stabilizes (around 0.08). These data indicate that the system's joints and ends approach the target state progressively, with the 3D position changes aligning and maintaining a smooth transition, thereby avoiding severe state changes and position overshooting.



1) At joints 2) At ends

Figure 6: 3D coordinate situation of reference trajectory sampling points.

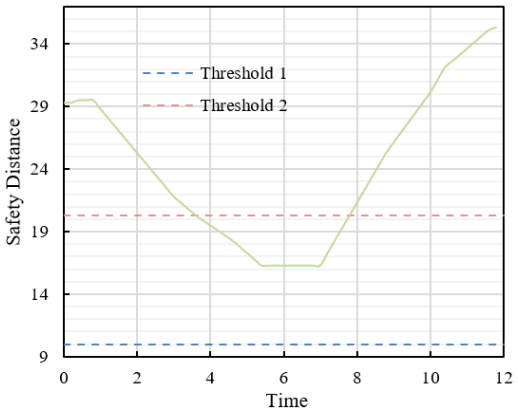


Figure 7: Human-machine safety distance greater than collision threshold experiment.

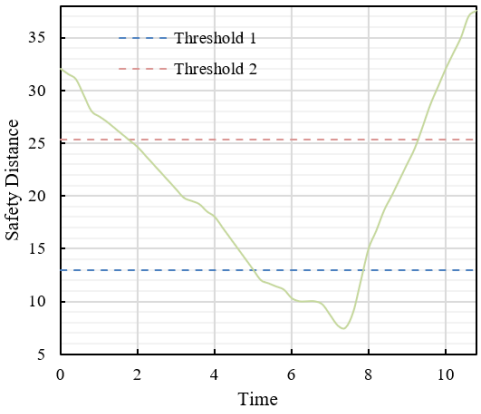


Figure 8: Human-machine safety distance less than collision threshold experiment.

In the experimental data presented in Figs. 7 and 8, the system demonstrates a relatively stable adjustment trend when the human-machine safety distance exceeds the collision threshold. Between 0 and 12 seconds, the safety distance decreases from an initial value of 29.3 to a minimum of 16.3, maintaining a stable state for an extended period until it starts increasing again after 18 seconds, ultimately reaching 35.3. This indicates that the system can effectively adjust the safety distance between humans and machines under varying external conditions, gradually reducing the safety distance in the initial phase, controlling it within a stable range, and then adjusting it to ensure smooth collaboration within the human-machine system. Furthermore, the data for reference threshold 1 (10) and threshold 2 (20) show that the actual safety distance remains consistently above the collision threshold, indicating that the system exhibits good collision avoidance control in this experimental scenario. In contrast, during the experiment where the human-machine safety distance falls below the collision threshold, the system displays more pronounced fluctuations. The safety distance gradually decreases from an initial value of 32, eventually reaching the critical value of 10 around 16 seconds, remaining at a low level for a period, and even dropping to 7.7 at one point. This indicates that, in this case, the system fails to effectively prevent a continuous decrease in the safety distance, significantly falling below thresholds 1 (13) and 2 (25). However, over time, the system gradually recovers and raises the safety distance to 37.5 after 26 seconds. This data shows that although the system did not promptly decelerate or maintain sufficient safety distance under certain conditions, it can still progressively restore a safe state for human-machine collaboration after adjustments. Thus, in the experiment with the safety distance below the collision threshold,

while the system's dynamic adjustment mechanism is effective, there are instances where it may not completely prevent a rapid decrease in safety distance, indicating that further optimization and improvement are necessary.

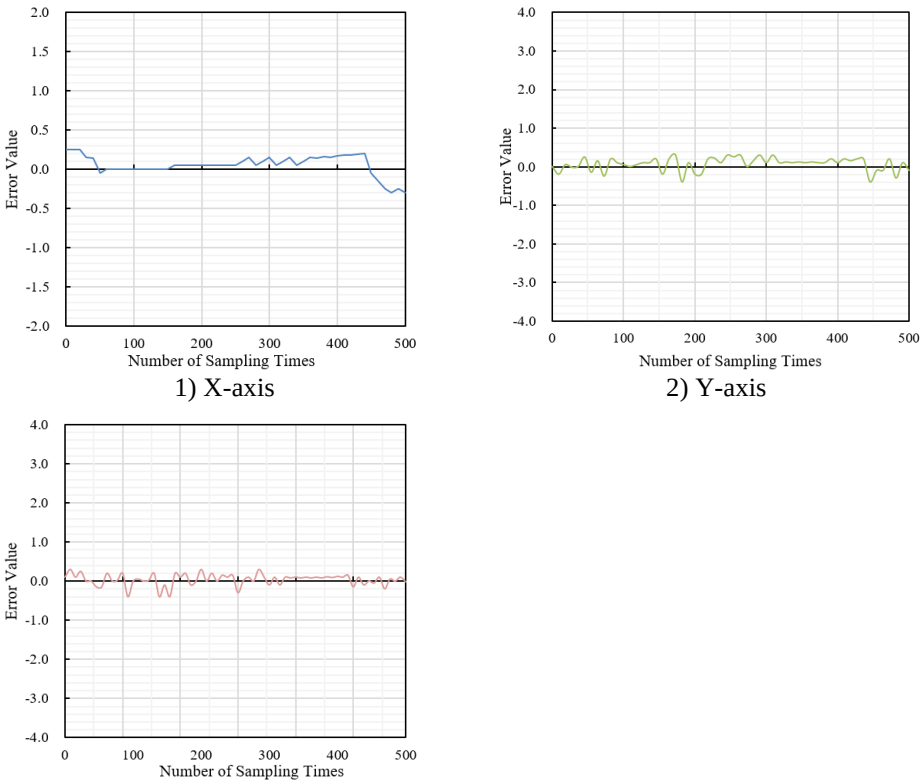


Figure 9: Human-machine safety distance measurement errors.

In the experimental results shown in Fig. 9, the measurement errors along the X, Y, and Z axes exhibit a certain fluctuation trend at different sampling points. Specifically, the error for the X-axis remains relatively stable at around 0.25 during the initial sampling points (0-100), then experiences a slight decrease to -0.05 between points 100 and 200, before gradually approaching zero. Within the 200–300 sampling points range, the error gradually increases to 0.15, followed by a significant negative fluctuation after the 500 sampling points (reaching -0.3). The Y-axis initially fluctuates considerably, with the error value dropping quickly from 0 to -0.2 within the 0–100 sampling points, then frequently alternating between positive and negative values in the 200–300 sampling points range (from -0.25 to 0.3), and stabilizing at a positive value of around 0.2 after the 400–500 sampling points. The Z-axis exhibits a more complex error change trend, starting at 0.1, experiencing considerable fluctuations (maximum error reaching -0.4) in the 100–200 sampling points range, gradually rising to 0.15 in the 300–400 sampling points, and approaching zero at the end of the 500 sampling points. These data indicate significant differences in measurement errors across the three axes, with overall fluctuations suggesting that the system's accuracy at different sampling points remains unstable.

Based on the above experimental results, the following conclusions can be drawn: First, the measurement errors of the system show noticeable fluctuations at different axes and sampling points, especially along the Y-axis and Z-axis, where frequent alternation between positive and negative errors indicates a lack of stability in the vertical direction of the sensors or measurement model. Second, despite some sampling points having error values controlled within ±0.1, certain intervals (such as X-axis 300–500 sampling points, Y-axis 100–300 sampling points, and Z-axis 0–100 sampling points) exhibit considerable fluctuations, with errors exceeding the threshold of 0.3. This could lead to inaccuracies in the system's judgment

of human-machine safety distance, thereby affecting the stability and safety of the collaboration system. To address these issues, future research should further optimize the measurement model, reduce the fluctuation amplitude of error values across different axes, and enhance overall system accuracy through dynamic compensation and multidimensional fusion prediction methods, thereby ensuring the safety and responsiveness of the human-machine collaboration system in complex scenarios.

5. CONCLUSION

This study explored the issues of brake control and safety distance prediction in human-machine collaboration systems within highly automated environments through two key directions. Firstly, the brake control method based on discrete time models has effectively enhanced the system's response accuracy and safety. Experimental results indicate that the human-machine collaboration system can promptly adjust the safety distance, maintaining it above the collision threshold in various scenarios, particularly demonstrating high stability in experiments where the safety distance exceeds the collision threshold. However, in experiments where the safety distance falls below the collision threshold, the system exhibited certain latency; although it can ultimately recover the safety distance, it failed to maintain adequate distance during certain periods, reflecting that the system's response mechanism still requires optimization under specific conditions.

Secondly, the adaptive safety distance prediction model based on a multilayer perceptron network can dynamically adjust the safety distance according to the scene, ensuring smooth collaboration. However, the error analysis of the three-dimensional coordinate reference trajectory sampling points indicates significant errors in certain ranges, particularly with frequent fluctuations in the Y-axis and Z-axis directions, impacting accuracy. This suggests that while the model achieves good predictive performance overall, there are limitations in dealing with large error fluctuations in complex multidimensional dynamic environments. Future research should further optimize the prediction model, reduce measurement errors, enhance the system's responsiveness in complex scenarios, and explore more sensing technologies and machine learning algorithms to improve the stability and safety of human-machine collaboration systems.

REFERENCES

[1] Javernik, A.; Buchmeister, B.; Ojstersek, R. (2023). The NASA-TLX approach to understand workers workload in human-robot collaboration, *International Journal of Simulation Modelling*, Vol. 22, No. 4, 574-585, doi:10.2507/IJSIMM22-4-658

[2] Sattinger, V.; Papa, M.; Stuja, K.; Kubinger, W. (2019). Methodik zur entwicklung sicherer kollaborativer produktionssysteme im rahmen von Industrie 4.0 (in German), *Elektrotechnik und Informationstechnik*, Vol. 136, No. 1, 318-325, doi:10.1007/s00502-019-00744-1

[3] Nia, N. G.; Bahrami, F.; Kaplanoglu, E.; Nasab, A. (2024). Unified neuromechanical control model for rhythmic and discrete hand movements, *Mathematical Modelling of Engineering Problems*, Vol. 11, No. 2, 279-289, doi:10.18280/mmep.110201

[4] Brending, S.; Lawo, M.; Pannek, J.; Sprodowski, T.; Zeising, P.; Zimmermann, D. (2017). Certifiable software architecture for human robot collaboration in industrial production environments, *IFAC-PapersOnLine*, Vol. 50, No. 1, 1983-1990, doi:10.1016/j.ifacol.2017.08.171

[5] Yadav, S.; Bhardwaj, V.; Thakur, D.; Sharma, V. (2023). Enhancing robotic process automation task selection: An integrated approach leveraging process mining and feature extraction, *Ingénierie des Systèmes d'Information*, Vol. 28, No. 5, 1247-1254, doi:10.18280/isi.280513

[6] Xiong, W.; Wang, C.; Ma, L. (2023). Partner or subordinate? Sequential risky decision-making behaviors under human-machine collaboration contexts, *Computers in Human Behavior*, Vol. 139, Paper 107556, 12 pages, doi:10.1016/j.chb.2022.107556

[7] Ema, A.; Akiya, N.; Osawa, H.; Hattori, H.; Oie, S.; Ichise, R.; Kanzaki, N.; Kukita, M.; Saijo, R.; Takushi, O.; Miyano, N.; Yashiro, Y. (2016). Future relations between humans and artificial intelligence: a stakeholder opinion survey in Japan, *IEEE Technology and Society Magazine*, Vol. 35, No. 4, 68-75, doi:10.1109/MTS.2016.2618719

[8] Ben-Assuli, O.; Heart, T.; Klempfner, R.; Padman, R. (2023). Human-machine collaboration for feature selection and integration to improve congestive heart failure risk prediction, *Decision Support Systems*, Vol. 172, Paper 113982, 12 pages, doi:10.1016/j.dss.2023.113982

[9] Rodrigues, I. R.; Dantas, M.; de Oliveira Filho, A. T.; Barbosa, G.; Bezerra, D.; Souza, R.; Marquezini, M. V.; Endo, P. T.; Kelner, J.; Sadok, D. (2023). A framework for robotic arm pose estimation and movement prediction based on deep and extreme learning models, *The Journal of Supercomputing*, Vol. 79, No. 7, 7176-7205, doi:10.1007/s11227-022-04936-z

[10] Xiao, Y.; Yang, S.; Xu, Z.; Liao, W.; Lu, Y. (2023). A human-machine collaboration frame in daylighting optimization of semi-outdoor space design by using phased synergistic method: a case study, *Journal of Building Engineering*, Vol. 79, Paper 107879, 23 pages, doi:10.1016/j.jobe.2023.107879

[11] Maurtua, I.; Ibarguren, A.; Kildal, J.; Susperregi, L.; Sierra, B. (2017). Human-robot collaboration in industrial applications: safety, interaction and trust, *International Journal of Advanced Robotic Systems*, Vol. 14, No. 4, Paper 1729881417716010, 10 pages, doi:10.1177/1729881417716010

[12] Ye, Y.; You, H.; Du, J. (2023). Improved trust in human-robot collaboration with ChatGPT, *IEEE Access*, Vol. 11, 55748-55754, doi:10.1109/ACCESS.2023.3282111

[13] Dobra, Z.; Dhir, K. S. (2020). Technology jump in the industry: human-robot cooperation in production, *Industrial Robot*, Vol. 47, No. 5, 757-775, doi:10.1108/IR-02-2020-0039

[14] Dalmasso, M.; Domínguez-Vidal, J. E.; Torres-Rodríguez, I. J.; Jiménez, P.; Garrell, A.; Sanfeliu, A. (2024). Shared task representation for human–robot collaborative navigation: the collaborative search case, *International Journal of Social Robotics*, Vol. 16, No. 1, 145-171, doi:10.1007/s12369-023-01067-0

[15] Liu, X.; Ge, S. S.; Zhao, F.; Mei, X. (2021). A dynamic behavior control framework for physical human-robot interaction, *Journal of Intelligent & Robotic Systems*, Vol. 101, No. 1, Paper 14, 18 pages, doi:10.1007/s10846-020-01286-x

[16] Hiatt, L. M.; Narber, C.; Bekele, E.; Khemlani, S. S.; Trafton, J. G. (2017). Human modeling for human-robot collaboration, *The International Journal of Robotics Research*, Vol. 36, No. 5-7, 580-596, doi:10.1177/0278364917690592

[17] Beschi, M.; Faroni, M.; Copot, C.; Pedrocchi, N. (2020). How motion planning affects human factors in human-robot collaboration, *IFAC-PapersOnLine*, Vol. 53, No. 5, 744-749, doi:10.1016/j.ifacol.2021.04.167

[18] Latella, C.; Lorenzini, M.; Lazzaroni, M.; Romano, F.; Traversaro, S.; Akhras, M. A.; Pucci, D.; Nori, F. (2019). Towards real-time whole-body human dynamics estimation through probabilistic sensor fusion algorithms: a physical human-robot interaction case study, *Autonomous Robots*, Vol. 43, No. 6, 1591-1603, doi:10.1007/s10514-018-9808-4

[19] Wang, C.; Zhao, J. (2024). Role dynamic assignment of human-robot collaboration based on target prediction and fuzzy inference, *IEEE Transactions on Industrial Informatics*, Vol. 20, No. 1, 471-481, doi:10.1109/TII.2023.3266378

[20] Cai, Z.; Feng, Z.; Zhou, L.; Ai, C.; Shao, H.; Yang, X. (2022). A framework and algorithm for human-robot collaboration based on multimodal reinforcement learning, *Computational Intelligence and Neuroscience*, Vol. 2022, Paper 2341898, 13 pages, doi:10.1155/2022/2341898

[21] Inkulu, A. K.; Bahubalendruni, M. V. A. R. (2024). Human-robot collaborative task planning for assembly system productivity enhancement, *Robotic Intelligence and Automation*, Vol. 44, No. 1, 120-130, doi:10.1108/RIA-05-2023-0067

[22] Samarathunga, S. M. B. P. B.; Valori, M.; Faglia, R.; Fassi, I.; Legnani, G. (2023). Considerations on the dynamics of biofidelic sensors in the assessment of human-robot impacts, *Machines*, Vol. 12, No. 1, Paper 26, 21 pages, doi:10.3390/machines12010026

[23] Inkulu, A. K.; Raju Bahubalendruni, M. V. A.; Dara, A.; SankaranarayanaSamy, K. (2021). Challenges and opportunities in human robot collaboration context of Industry 4.0 – a state of the art review, *Industrial Robot*, Vol. 49, No. 2, 226-239, doi:10.1108/IR-04-2021-0077