frontiers | Frontiers in Mechanical Engineering

TYPE Original Research
PUBLISHED 15 January 2026
DOI 10.3389/fmech.2026.1759452

OPEN ACCESS

EDITED BY
Chengxi Zhang,
Jiangnan University, China

REVIEWED BY
Lanhao Zhao,
Beijing University of Technology, China
Haoming Zhu,
Xi'an University of Science and Technology, China

*CORRESPONDENCE
Nan Zhang,
zhangnan2025206@126.com

RECEIVED 03 December 2025
REVISED 04 January 2026
ACCEPTED 05 January 2026
PUBLISHED 15 January 2026

CITATION
Zhang N (2026) Online fault compensation control method for ROV based on decoupling algorithm.
Front. Mech. Eng. 12:1759452.
doi: 10.3389/fmech.2026.1759452

Online fault compensation control method for ROV based on decoupling algorithm

Nan Zhang*

Asset Management Department, Tianjin Bohai Vocational Technical College, Tianjin, China

Introduction: With the increasing complexity of underwater operations, remotely operated vehicles systems face the dual challenges of multi-source interference and component failures in unknown environments.
Methods: To achieve high-precision control of remotely operated vehicles arms under fault conditions, this paper proposes an online fault compensation control method based on a decoupling algorithm. This method separates the end-effector position and attitude control of the master and slave arms through a pose decoupling algorithm, constructs an observer-based fault diagnosis mechanism, and combines H∞ robust control and online adaptive strategies to achieve dynamic compensation for combined sensor and thruster faults.
Results: The results show that in dual-arm cooperative operation, the spatial trajectory tracking deviation of the robotic arm can be controlled within 4.3 mm, with a maximum deviation of 2.643 mm in the X-axis direction and a planning deviation of 3.075 mm in the Y-axis direction. Compared with backstepping fault-tolerant control and power sliding mode control, the method used in this study has a maximum deviation of only 0.01° in yaw angle control, a position control error reduced to 1.2 mm, and a maximum trajectory tracking error of 2.1 mm, which is significantly better than the comparative methods. Furthermore, the system can rapidly approach the desired posture within 50 seconds and maintains stable operation under various fault scenarios.
Discussion: This demonstrates that the proposed method can effectively improve the operational accuracy and fault "tolerance of remotely operated vehicles in complex environments, providing a new technology for solving the control problems of robot systems under fault conditions."

KEYWORDS

control, decoupling algorithm, fault compensation control, remote control operation, remotely operated vehicles

1 Introduction

With the increasing demand for marine resource exploration and underwater operations, the operational capabilities and reliability of Remotely Operated Vehicles (ROVs) have received widespread attention as key equipment for humans in extreme underwater environments (Zhou et al., 2025; Weber-Lewerenz and Traverso, 2023). Among them, the integrated hydraulic manipulator of the operational ROV is widely used in precision tasks such as pipeline inspection, equipment maintenance, and sample retrieval. However, the underwater environment is characterized by strong uncertainty, time-varying ocean current interference, and high-pressure confinement, which can easily lead to performance degradation or sudden failure of the ROV's sensors and thrusters (Hu et al., 2024). Furthermore, the strong coupling and nonlinear dynamic characteristics of

the manipulator during movement pose a severe challenge to its precise posture control (Deng et al., 2024).

Therefore, how to design an intelligent control method that can achieve high-precision pose tracking and online self-healing capability under the complex working conditions of ROVs with external disturbances, internal model uncertainties and multiple sensor/actuator failures has become a research difficulty and frontier focus in the current field. Yuan et al. proposed a finite-time adaptive fault-tolerant control method. This method handled system uncertainties by integrating the backstepping method and neural network technology, and introduced an error transformation strategy to transform transient performance constraints into equivalent stable problems. Simultaneously, a novel adaptive upper bound estimation strategy was utilized to effectively compensate for neural network training errors and external disturbances. The research findings denote that the developed control scheme can ensure that the position tracking error converges to near zero in a finite time, and it has been verified in the simulation of a two-link manipulator and the Franka-Emika Panda robot experiment (Yuan and Sun, 2025). Li et al. proposed a fixed-time fault-tolerant control strategy based on a fast disturbance observer. This method realized the synchronous estimation of actuator failure, matching and mismatch disturbances by constructing a novel disturbance observer, and designed a fixed-time non-singular terminal sliding surface with faster convergence characteristics based on the estimation information. The findings revealed that the method can still maintain superior control performance under extreme conditions of mismatch disturbance and actuator failure at the same time, and its convergence speed and robustness are significantly better than traditional control schemes (Li et al., 2025). Zhang et al. proposed an innovative control strategy that combines hybrid gain adaptive and robust specified performance control. The method achieved comprehensive suppression of actuator failure, model uncertainty, and external disturbance by constructing a hybrid gain adaptive law and specified performance control framework. The outcomes demonstrated that the method could ensure that trajectory tracking has a predefined stable time and accuracy under extreme compound faults (Zhang JX. et al., 2024). Zhang et al. proposed a non-fragile specified performance control scheme for actuator failure robot manipulators to solve the fragility problem of existing specified performance control methods. The method effectively avoided control singularity problems by adaptively relaxing the constraint range of the adjustment term when the error is close to the boundary. Then, the error transformation technique was used to transform the tracking control problem with performance constraints into an equivalent system stabilization problem, and the fuzzy neural network was employed to approximate the system uncertainty (Zhang J. et al., 2024).

In summary, to solve the issues of strong coupling, insufficient fault tolerance for complex faults, and poor operational intuitiveness in existing ROV arm control systems operating in complex underwater environments, this study innovatively proposes an intelligent control method that integrates pose decoupling and online fault compensation. By employing a heterogeneous master arm configuration to achieve separate control of the end effector's position and attitude, operational intuitiveness is improved. An observer-based real-time fault diagnosis mechanism is designed, and combined with H ∞ robust control and online adaptive compensation strategies, external disturbances are effectively suppressed, and dynamic fault-tolerant control is achieved. This research aims to improve the accuracy of fault compensation control for ROVs and enhance their operational efficiency.

2 Materials and methods

2.1 ROV remote control operating system based on decoupling algorithm

Due to the complexity of underwater environments and the harsher working conditions, the ROV remote control operating system needs to meet requirements such as anti-interference and waterproofing. Therefore, a hydraulic robotic arm is designed to achieve underwater operation control of the robot. A dynamic model of the ROV robotic arm is constructed. Equation 1 shows the joint space dynamic equation.

$$\tau = M(q)\ddot{q} + C(q,\ddot{q})\dot{q} + G(q) + F(\dot{q}) \tag{1}$$

In Equation 1, τ represents the joint force vector. q, $\dot{q}$ and $\ddot{q}$ represent the joint position, velocity, and acceleration vectors, respectively. $M(q)$ represents the inertia matrix, $C(q,\ddot{q})$ represents the Coriolis force and centripetal force matrices, $G(q)$ represents the gravity vector, and $F(\dot{q})$ represents the friction vector. Considering the additional fluid effects during ROV operation, Equation 2 is obtained.

$$\tau = M(q)\ddot{q} + C(q,\ddot{q})\dot{q} + G(q) + F(\dot{q}) + \tau_{hydro} \tag{2}$$

In Equation 2, τ_{hydro} represents the additional mass force. When the robot accelerates in the water, it will drive the surrounding fluid instruments to accelerate, which is equivalent to increasing its own weight. Therefore, its calculation formula is shown in Equation 3 (Ren et al., 2023).

$$\tau_{hydro} = M_a(q)\ddot{q} \tag{3}$$

In Equation 3, $M_a(q)$ represents the additional mass matrix. Meanwhile, the robot will experience resistance from the water flow during operation, as shown in Equation 4 (Lin et al., 2023; Liao et al., 2023).

$$\begin{cases} \tau_{drag} = D(q,\dot{q})\dot{q} \\ F_{drag} = \dfrac{1}{2}\rho C_d A v^2 \end{cases} \tag{4}$$

In Equation 4, τ_{drag} represents fluid resistance, $D(q,\dot{q})$ represents the damping matrix, and F_{drag} represents the simplified fluid resistance. ρ represents fluid density, C_d represents drag coefficient, A represents the frontal area, and v represents relative velocity. Therefore, the complete underwater dynamic equations of the robotic arm are shown in Equation 5 (Adam et al., 2024; Wang et al., 2024).

$$\tau = [M(q) + M_a(q)]\ddot{q} + C(q,\dot{q})\dot{q} + G_{gb}(q) + F(\dot{q}) + D(q,\dot{q})\dot{q} \tag{5}$$

In Equation 5, G_{gb} represents the gravity and buoyancy term. To ensure precise control and operational comfort of the six-degree of

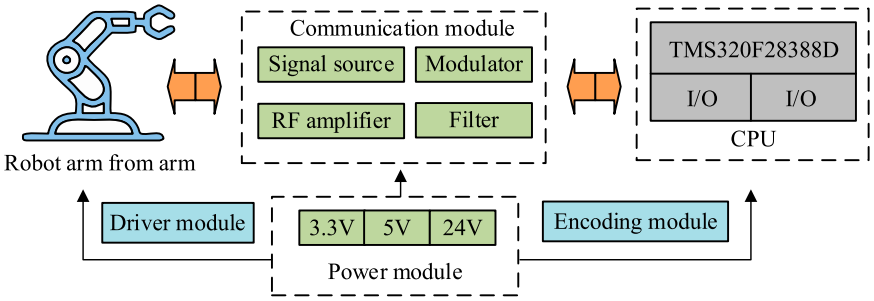

FIGURE 1
Master Manipulator hardware system of robot arm.

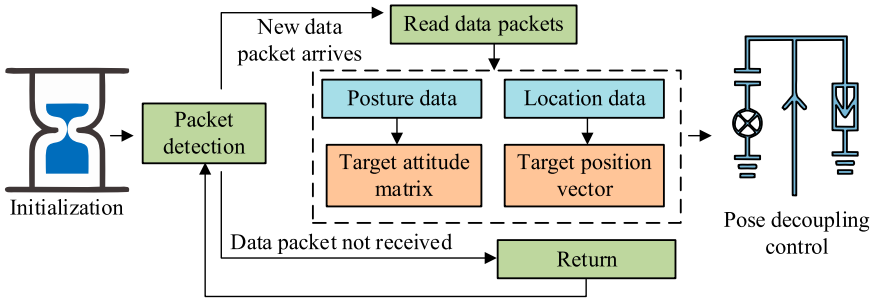

FIGURE 2
Operation process of pose decoupling algorithm.

freedom hydraulic manipulator, the master arm is designed with a heterogeneous configuration. End-effector position control is achieved through three degrees of freedom in the hand, and posture control through three degrees of freedom in the wrist, ensuring position-posture decoupling while also considering ergonomics to improve operational flexibility and comfort. Therefore, the hardware system of the master arm is shown in Figure 1.

As shown in Figure 1, the hardware adopts a heterogeneous configuration design, and its system mainly consists of a core processing unit, a pose acquisition module, a driver and communication module, and a power supply unit. Among them, the pose acquisition module obtains real-time joint data, and the core processing unit executes decoupling algorithms and generates control instructions based on this. Finally, precise and decoupled control of the slave end robotic arm is achieved through the drive and communication module. The Master Manipulator hardware system can be modeled as a multi-degree-of-freedom kinematic chain, and its end pose can be calculated through forward kinematics as shown in Equation 6.

$$T_{end}^{bose} = \prod_{i=1}^{6} T_i^{i-1}(\theta_i) \tag{6}$$

In Equation 6, T_{end}^{bose} represents the pose matrix from the base to the end, and T_i^{i-1} represents the homogeneous transformation matrix of the i-th joint. θ_i represents the joint angle. Simultaneously, the master arm system uses a decoupling algorithm to control the robotic arm's posture. Figure 2 shows the operation flow of the robotic arm posture decoupling algorithm.

From Figure 2, the decoupling algorithm first initializes. Then the algorithm continuously detects whether new data packets arrive. If no data packets are received, the detection continues in a loop. Once the data packet is confirmed to be received, the data in the data packet is read immediately. Then the algorithm enters the data parsing stage and separates the attitude data and position data from the data packet. Then the system converts the parsed attitude data into the target attitude matrix and the position data into the target position vector. Finally, the algorithm outputs the calculated target attitude matrix and target position vector, thus completing one pose decoupling control process. After that, the system returns to the data packet detection stage and starts a new loop. Among them, the first to third degree joints will first decouple from the pose to the end of the master hand, then from the end of the master hand to the desired matrix of the robot arm and finally complete the joint angle servoing. The fourth to sixth joints realize the pose transformation through the free rotation of the three-dimensional angle and finally reach the desired matrix pose and realize the joint servoing. The decoupling algorithm means that the ratio of the end position of the master hand to the end position of the slave arm is a constant. The remote control operation of the robot arm can be realized by controlling the constant. The core of the pose decoupling algorithm is to decompose the end pose matrix into position vectors and pose

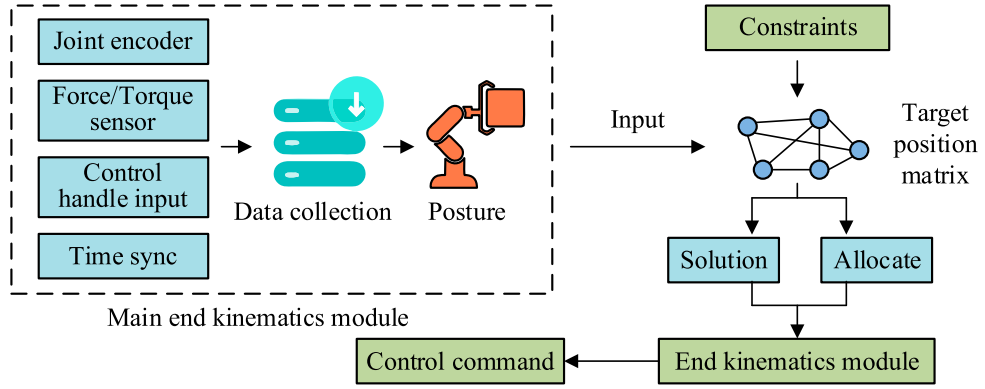

FIGURE 3
Process of pose collaborative decoupling operation.

matrices. The formula for the end pose matrix is shown in Equation 7.

$$T = \begin{bmatrix} R & p \\ 0 & 1 \end{bmatrix} \tag{7}$$

In Equation 7, T represents the end pose matrix, R represents a 3×3 orthogonal matrix, and R represents a 3×1 column vector. In the algorithm process, after parsing the data packet, the target pose description and target position coordinates are obtained, which need to be converted into the matrix and vector forms required for control. The goal of position servo is to drive the end effector of the robotic arm to the desired position, while the goal of attitude servo is to rotate it to the desired posture. To achieve this goal, it is necessary to map the desired end effector posture to the desired angles of each joint through inverse kinematics. The motion space formula of the master hand of the robot arm is Equation 8 (Chen et al., 2025).

$$\begin{cases} a_m = |w_{mx\max} - w_{mx\min}| \\ b_m = |w_{my\max} - w_{my\min}| \\ c_m = |w_{mz\max} - w_{mz\min}| \end{cases} \tag{8}$$

In Equation 8, $w_{mx\max}$, $w_{mx\min}$, $w_{my\max}$, $w_{my\min}$, $w_{mz\max}$, and $w_{mz\min}$ represent the maximum and minimum values of the master hand space in different directions xyz, respectively. The pose coordination decoupling operation process through the decoupling algorithm is shown in Figure 3.

As shown in Figure 3, the pose coordination decoupling process first involves the master end-effector kinematics module calculating the pose of the master arm's end effector and inputting it into the system as the desired target position matrix. Then, the system, combined with preset dual-arm collaborative pose constraints, calculates and allocates the target position matrix to generate the desired pose matrices for the left and right robotic arms' end effectors that meet the coordination requirements. Finally, the slave end-effector kinematics module converts the calculated desired pose matrices into specific control commands for each robotic arm joint, thereby achieving precise control of the dual-arm collaborative operation. In dual arm collaborative work, the master hand commands (or task planning) generate a "virtual master end" pose, which is then decoupled and assigned to the left and right robotic arms based on collaborative constraints. The expected position of the left and right arms is represented by the following model as Equation 9.

$$\begin{cases} P_{L,des} = P_{master} + \Delta P_L \\ P_{R,des} = P_{master} + \Delta P_R \end{cases} \tag{9}$$

In Equation 9, $P_{L,des}$ and $P_{R,des}$ represent the calculated expected positions of the left and right robotic arm ends, respectively. P_{master} represents the position of the "reference target point" directly generated by the Master Manipulator operation or task planning, and ΔP_L and ΔP_R respectively represent the fixed or adjustable offset vectors of the left and right arms relative to the reference target point.

2.2 Design of online fault compensation control system for ROV

After the decoupling operation of the ROV's robotic arm is completed, the robot's robotic arm may experience sensor and thruster failures due to water pressure and different complex environments. Therefore, to ensure the normal operation of the robot, a robust online fault compensation control system for the robot's robotic arm based on thrust is designed. When the robot experiences a sensor failure, the output torque changes as shown in Equation 10 (Zhang et al., 2023; Lang et al., 2024).

$$u(t) = u_c(t) + d_a(t) \tag{10}$$

In Equation 10, $u(t)$ represents the input torque during the sensor failure process, $u_c(t)$ represents the input control signal when there is no failure, and $d_a(t)$ represents the thruster failure. The dynamic equation of the robot arm considering both thruster and sensor failures is shown in Equation 11.

$$\begin{cases} \dot{x}(t) = Ax(t) + B_1(u_c(t) + d_a(t) + B_2 d_w(t)) \\ \dot{y}(t) = Cx(t) + d_s(t) \end{cases} \tag{11}$$

In Equation 11, $\dot{x}(t)$ represents the derivative of the state vector $x(t)$ with respect to time, A represents the system matrix, B_1

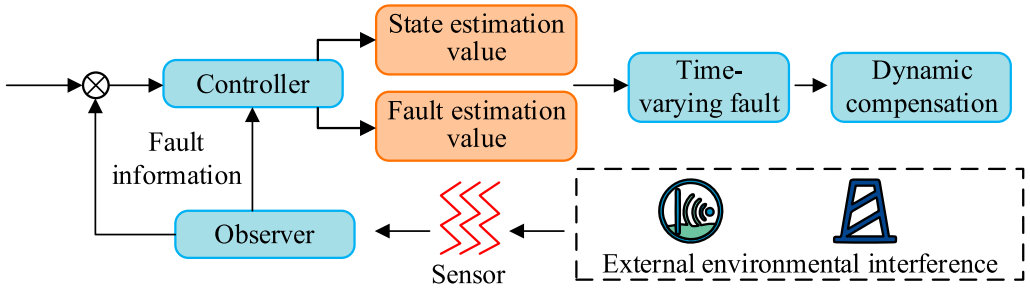

FIGURE 4
Control system structure.

represents the control input matrix, B_2 represents the external disturbance distribution matrix, $d_w(t)$ represents the external disturbance, C represents the output matrix, and $d_s(t)$ represents the sensor fault. The entire control system structure is shown in Figure 4.

In Figure 4, the control architecture first uses the designed observer to perform real-time joint estimation of the system state and compound faults, and then feeds back the estimated fault information to the controller. The controller integrates the state estimate and the fault estimate, combines the adaptive law to dynamically compensate for time-varying faults, and constructs a robust control law based on H ∞ control theory, thereby effectively suppressing external disturbances while ensuring system stability. The fault-tolerant control of robot posture online faults is achieved by designing an observer. First, the posture system is converted into an augmented form through the system state, as shown in Equation 12 (Qiu et al., 2024; Sengolrajan et al., 2023).

$$\begin{cases} G_1\dot{E}(t) = A_1E(t) + B_1u_c(t) + B_2d_w(t) \\ \dot{y}(t) = G_2\dot{E}(t) \end{cases} \tag{12}$$

In Equation 12, $\dot{E}(t)$ is the inverse matrix of $x(t)$, $d_s(t)$ and $d_w(t)$, $A_1 = [A, 0, B_1]$, G_1 is matrix $[I_{6*6}, 0_{6*6}, 0_{6*3}]$, G_2 is matrix $[C, I_{6*6}, 0_{6*3}]$, and $G_3 = [0_{3*6}, 0_{3*6}, I_{3*3}]$. Then the entire observer expression is as shown in Equation 13.

$$\dot{E}_s(t) = Q_1A_1E_s(t) + Q_1B_1u_c(t) + Q_2G_2\dot{E}(t) + L\left(\dot{y}(t) - G_2\dot{E}_s(t)\right) \tag{13}$$

In Equation 13, Q_1 is the inverse matrix of G, and L represents the observation gain matrix. $Q_1G_1 + Q_2G_2 + Q_3G_3 = I$ and Q_2 are the sensor measurement outputs $\dot{y}(t)$ used to correct and estimate the entire augmented state. Q_3 is the actuator fault directly extracted from the augmented state. Since the underwater robot has both external and internal faults, the attitude control of the entire robot is difficult to estimate accurately. Therefore, a virtual observer is proposed for accurate pose localization. At the same time, auxiliary variables are introduced into the observer, and the differential terms of the variables are used to offset the corresponding terms in the observation to reduce the influence of the differential terms and improve the accuracy estimation effect of the robot's robotic arm. The formula for the differential term is shown in Equation 14 (Ye et al., 2024; Shen et al., 2023).

$$\dot{K}(t) = (Q_1A_1 - LG_2)K(t) + Q_1B_1u_c(t) + (Q_1A_1 - LG_2)Q_2G_2E(t) + L\dot{y}(t) \tag{14}$$

In Equation 14, $\dot{K}(t)$ represents the differential term. An online fault compensation control system is designed using an online fault compensation control observer. The calculation formula for convergence error is shown in Equation 15.

$$\dot{e}(t) = (A_e - LC_e)e(t) + D_ed(t) \tag{15}$$

In Equation 15, A_e and C_e represent the augmented matrices, L represents the gain matrix of the observer, D_e represents the interference distribution matrix, $d(t)$ represents external interference, $\dot{e}(t)$ represents the convergence error of the observer, and the observation error $\dot{e}(t)$ is uniformly ultimately bounded, and the error exponent $d(t)$ converges to zero at that time. By designing the gain L reasonably, the convergence of the observer can be ensured. The fault compensation control flow is shown in Figure 5.

In Figure 5, the surface control system provides a human-machine interface through a display screen. Operation commands are processed by an industrial control computer, communicating with the display terminal via Ethernet and connecting to peripheral devices such as knobs, emergency stop switches, joysticks, and indicator lights via an Arduino platform. Control signals are transmitted to the underwater communication receiver via a surface communication receiver in a cable. The core Raspberry Pi microcontroller of the underwater control system receives commands, processes sensor data in conjunction with the Maxlink Pixhawk navigation control module, and precisely drives eight thrusters through an ESC module to achieve motion control. Simultaneously, it controls payloads such as cameras and LED lights, forming a complete underwater operation closed loop. The surface control system controls the ROV by issuing commands. To enable communication, a navigation control system is used to transmit communication signals. The navigation control communication transmission device is shown in Figure 6.

In Figure 6, the processor acquires attitude data by connecting to the three-axis gyroscope and accelerometer via the SPI bus, and reads the orientation information from the three-axis magnetometer and the data from the barometric depth gauge via the bus. Simultaneously, it monitors voltage through the ADC channel and receives safety switch and remote control commands

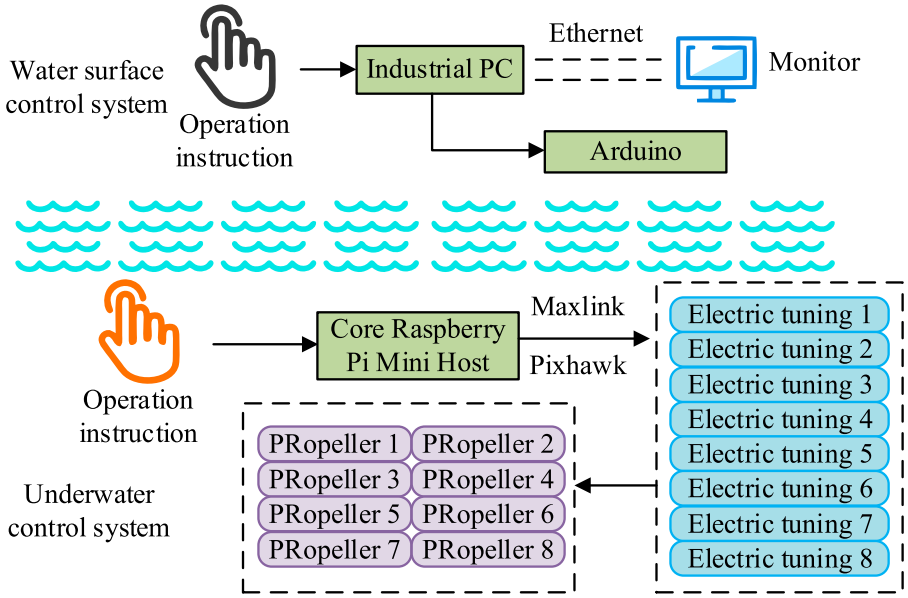

FIGURE 5
Fault compensation control process.

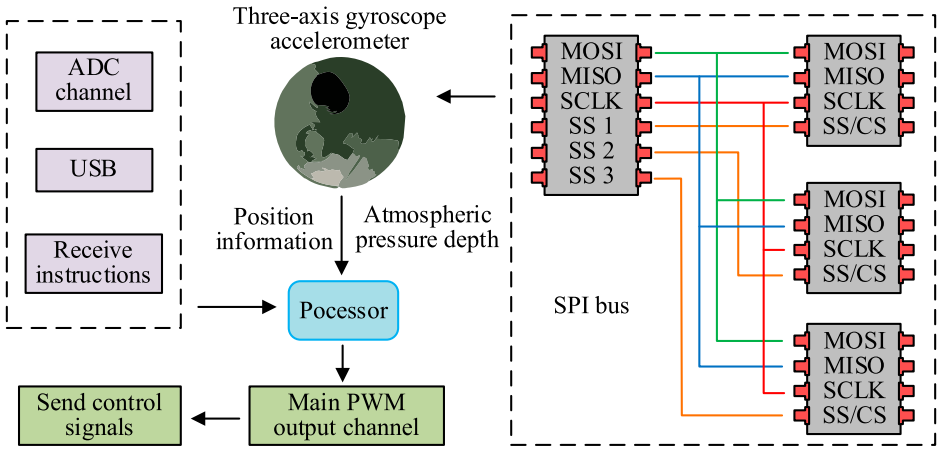

FIGURE 6
Aviation control communication transmission device.

through multiple interfaces such as USB. Finally, it sends control signals through the main PWM output channel and achieves multi-device collaborative control with the help of the CAN bus, UART, and auxiliary PWM.

3 Results and analysis

3.1 Analysis of robot pose control results based on decoupling algorithm

To verify the operational effectiveness of the decoupled operating system, this study uses a large-scale operational ROV system as the experimental object. Simulation analysis was used to verify the system's operational performance. A high-precision model of the ROV and robotic arm based on hydrodynamic effects was constructed using the Gazebo high-fidelity physics simulation environment. A collaborative task involving underwater valve grasping and turning was designed. To verify the performance of the pose-decoupled control, a position tracking experiment was conducted on the remote hydraulic arm. During the simulation, the surface control unit (Ubuntu system) integrated with the ROS framework and Gazebo simulation was used. The underwater computing unit used lightweight Linux to process sensor data, and the real-time control unit used FreeRTOS to implement the underlying drivers. The master controller, while fixing the attitude of its end effector sphere, performed displacement,

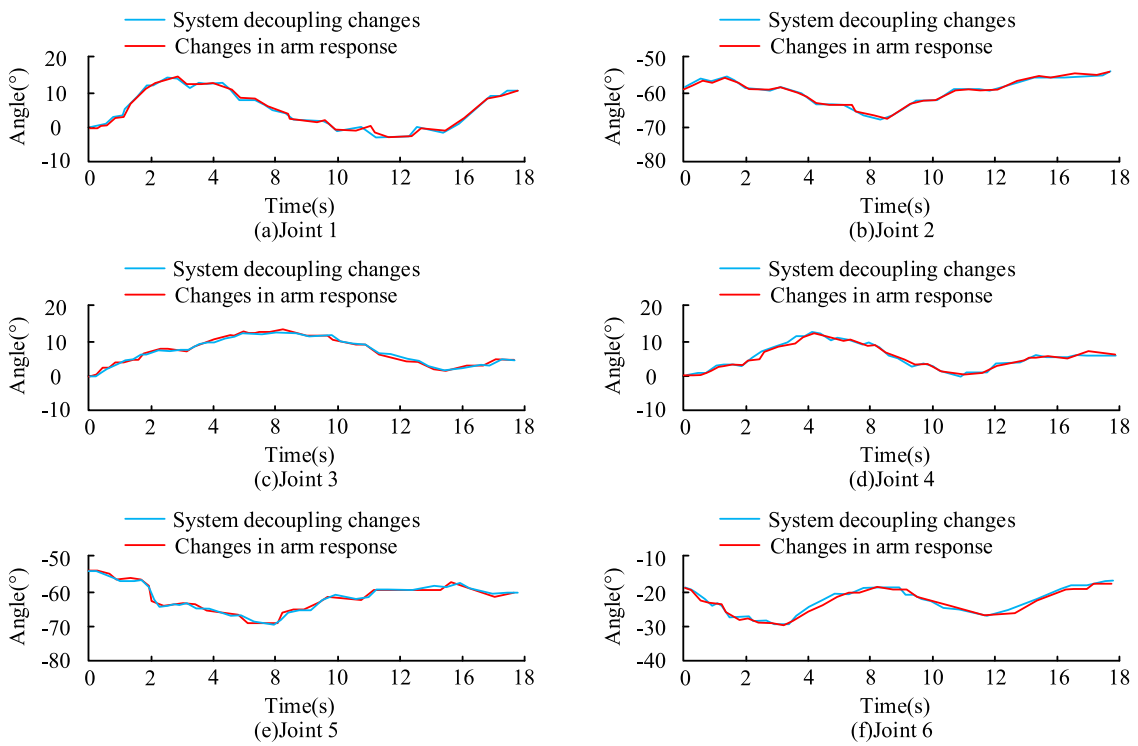

FIGURE 7
Changes in the operation of the robot arm system. **(a)** Joint 1. **(b)** Joint 2. **(c)** Joint 3. **(d)** Joint 4. **(e)** Joint 5. **(f)** Joint 6.

using this command to control the position of the end effector of the remote hydraulic arm. The operational changes of the robot arm system are shown in Figure 7.

From Figures 7a–f, in the analysis of decoupling operations of different robot arm joints, each joint at each position could follow the master arm system for decoupling control, and the motion parameters and motion status of the slave arm were consistent with the parameters of the master arm system. This indicates that the entire system conforms to the actual operational changes during operation. Furthermore, the angle change deviation of each different joint did not exceed 2°, indicating that the master arm system can control the changes in the slave arm during the entire system operation. Meanwhile, different robot joints could achieve angle control changes in a very short time, which also indicates that the operation of the entire system is more rapid. The simulation analysis of the end effector motion trajectory of the ROV in different spatial directions yields the results shown in Figure 8.

As shown in Figure 8a, there was a significant deviation in the operating position between the master arm system and the slave arm in the X-direction simulation results. The largest deviation occurred around 5 s, reaching a maximum of 3 mm. This is due to the coupling connection between different components during the robotic arm assembly process, resulting in a certain spatial deviation in the entire robotic arm motion system, thus causing deviations in motion angle and height during movement. As shown in Figure 8b, the maximum height deviation of the master-slave system in the longitudinal motion change of the robotic arm could reach 0.3 mm, with some time deviations persisting. As shown in Figure 8c, the Z-axis motion change analysis showed that the Z-axis deviation was the largest among the three directions, reaching a maximum of 5 mm. Therefore, the robotic arm exhibited significant motion deviations during system operation simulation. This is due to the large mapping of the robotic arm's end-effector posture during operation, leading to dynamic errors in the kinematic model. However, the overall motion deviation is relatively small, indicating that the system can achieve decoupled control of the robotic arm. The simulation results for different poses were compared and analyzed, as shown in Table 1.

As shown in Table 1, there were significant differences in the robot's trajectory tracking accuracy across different coordinate axes during various pose changes. The right arm's actual X-axis deviation reached 2.643 mm, while the planned Y-axis deviation was 3.075 mm. The actual spatial deviation of the right arm reached 4.251 mm, exceeding the planned value, indicating error accumulation during dynamic following. The left arm's actual spatial deviation was 4.035 mm, approximately 0.41 mm higher than the planned value, suggesting similar error accumulation characteristics in both arms. However, the table shows that the spatial deviations of both arms were controlled within 4.3 mm, indicating that the system maintains good trajectory tracking performance and can meet basic operational requirements.

3.2 Analysis of the effect of online fault compensation control

To verify the effectiveness of the proposed fault compensation control, a robot system with dimensions of 608 × 294 × 196 mm and

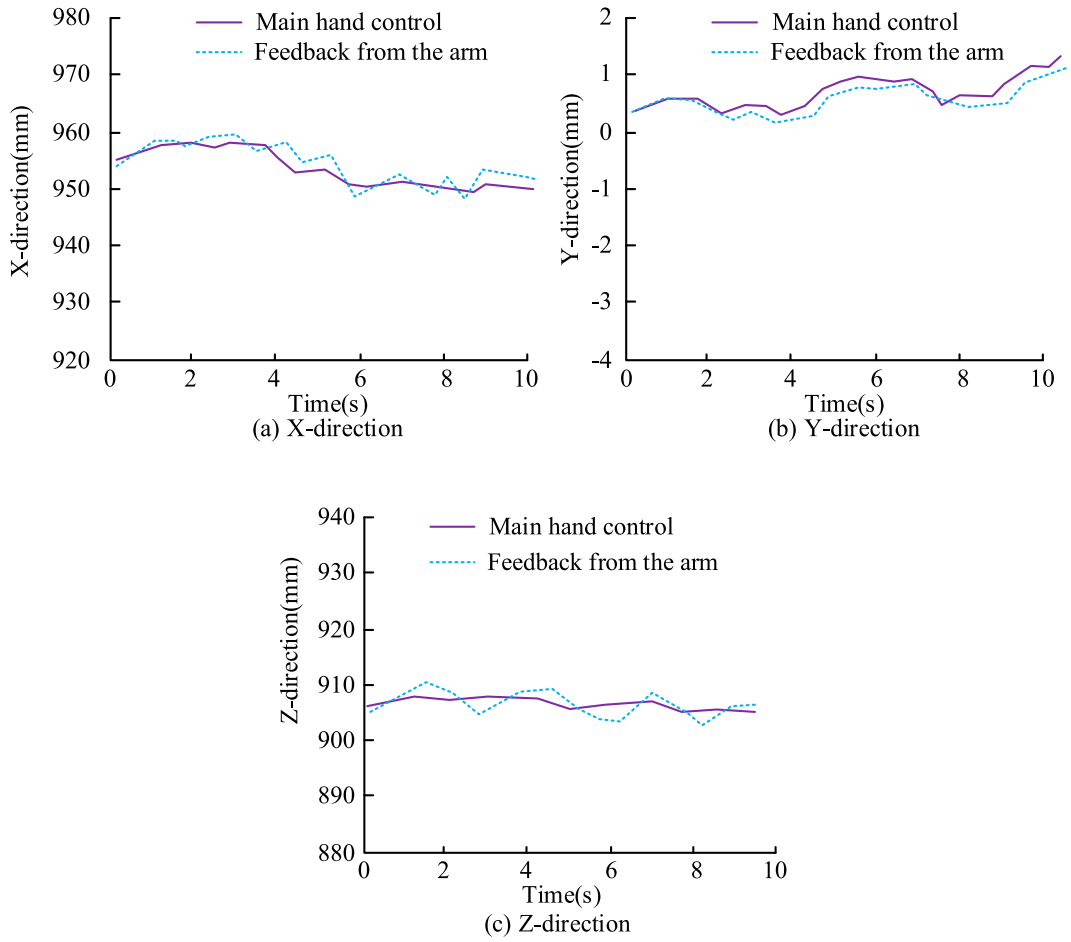

FIGURE 8
Simulation control analysis of robot arm with different dimensions. **(a)** X-direction. **(b)** Y-direction. **(c)** Z-direction.

TABLE 1 Comparison results of anti vibration deviation for different poses.

Object	X-axis deviation (mm)	Y-axis deviation (mm)	Z-axis deviation (mm)	Spatial deviation (mm)
Left arm planned	1.752	2.413	1.928	3.625
Left arm actual	2.185	2.867	1.794	4.035
Right arm planned	2.341	3.075	1.526	4.128
Right arm actual	2.643	2.892	1.773	4.251
Target planned	2.121	2.764	1.647	3.894
Target actual	2.235	2.781	1.635	4.017

a total weight of 8 kg was simulated and analyzed. It was equipped with a 300 Wh battery pack. Its operating environment temperature ranged from −10 °C to 45 °C, with a maximum diving depth of 200m, a maximum speed of 3 m/s, and a horizontal operating radius of 400 m. The propulsion system provided thrust output of 5.7 kg in the forward direction, 4 kg in the upward direction, and 3.6 kg in the lateral direction. The simulation analysis results from different angles were compared and analyzed, as shown in Figure 9. The backstepping fault-tolerant control method was compared and analyzed with the proposed method. Backstepping fault-tolerant control is an advanced control strategy that combines backstepping and fault-tolerant control concepts. Through a recursive and systematic design process, and by virtually designing a control law and stability guarantee for each subsystem, a practical control law capable of tolerating faults for the entire system was finally derived. The comparison of different methods for supplementary control from different angles is shown in Figure 9.

As shown in Figure 9a, in the comparison of yaw angle control effects, the angle changes of different fault-compensation control methods approached the desired angle change with time. The

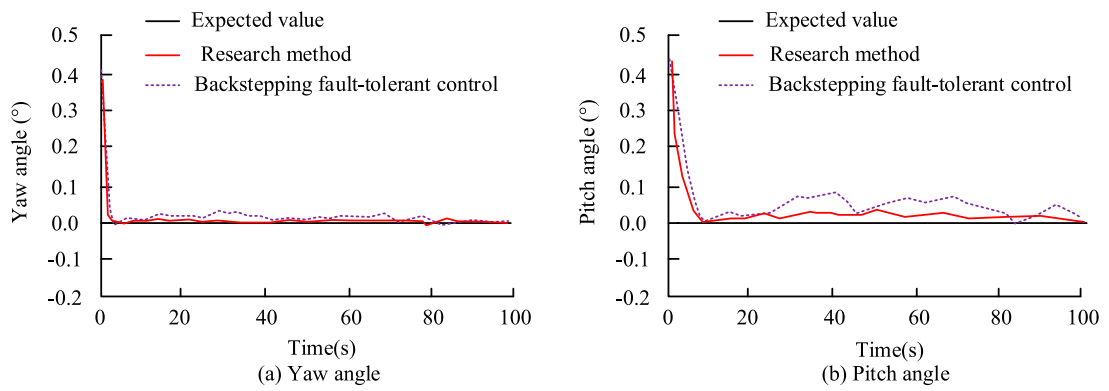

FIGURE 9
Comparison of supplementary control from different perspectives using different methods. **(a)** Yaw angle. **(b)** Pitch angle.

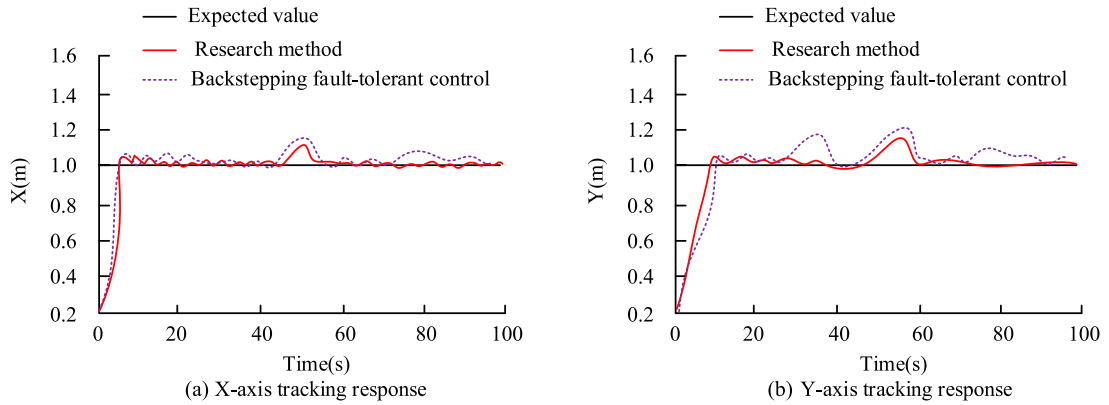

FIGURE 10
Comparison of compensation control with different control methods. **(a)** X-axis tracking response. **(b)** Y-axis tracking response

backstepping fault-tolerant control showed a relatively large deviation from the desired angle, with a maximum deviation of 0.03°, while the research method had a maximum deviation of only 0.01°. This indicates that the compensation control used in the study has a better control effect compared to other methods. Furthermore, in the angle change, the research method approached the desired angle in a very short time, indicating that the control is faster and has a better control effect. As shown in Figure 9b, in the comparison of pitch angle simulation results, the maximum deviation angle of the backstepping fault-tolerant control reached 0.08°, while the maximum deviation angle of the research method was only 0.03°. It is evident that the research method also has a better control effect than the backstepping fault-tolerant control in the pitch angle comparison analysis. The control response effect under the condition that both the thruster and sensor malfunctions is shown in Figure 10.

As shown in Figure 10a, the proposed control compensation method exhibited better control performance in the X-axis direction. After reaching the desired value, its control deviation showed a fluctuating trend, reaching a maximum deviation of 0.07 m at 50s. The backstepping fault-tolerant control also reached a maximum deviation of 1.0 m at 50s, an increase of 0.03 m compared to the proposed method. This indicates that the backstepping fault-tolerant control performs worse in motion tracking for ROVs compared to the method used in the study. Figure 10b shows that in Y-axis tracking control, the proposed method exhibited smaller deviation changes, with a maximum deviation of only 0.09 m. The backstepping fault-tolerant control method showed a larger deviation, reaching a maximum of 0.11m, and exhibited three significant deviations. Therefore, the proposed control method demonstrates better tracking control performance among different control methods. To analyze the actual effectiveness of the proposed method, a comparative analysis of more different control methods was conducted, as shown in Table 2. The study compared the model predictive fault-tolerant control method and the power sliding mode control method.

As shown in Table 2, in the X-axis direction, the proposed control method exhibited the best performance, with a yaw angle deviation of only 0.8°, a position control error of 1.2 mm, and a maximum trajectory tracking error of 2.1 mm. In contrast, the model predictive fault-tolerant control showed a yaw angle deviation of 1.3°, a position control error of 2.2 mm, and a maximum tracking error of 3.8 mm in the same direction, representing a significant increase in all indicators compared to the proposed method. The power sliding

TABLE 2 Comparison of deviation between different control methods.

Control method	Axis	Yaw error (°)	Pitch error (°)	Roll error (°)	Position control error (mm)	Trajectory tracking max error (mm)	Trajectory tracking average error (mm)
Research method	X-axis	0.8	0.6	0.5	1.2	2.1	1.2
Research method	Y-axis	0.9	0.7	0.6	1.5	2.3	1.4
Model predictive fault-tolerant control	X-axis	1.3	1.0	0.9	2.2	3.8	2.1
Model predictive fault-tolerant control	Y-axis	1.5	1.3	1.1	2.6	4.3	2.5
Power sliding mode control	X-axis	2.0	1.8	1.5	3.5	6.2	3.8
Power sliding mode control	Y-axis	2.3	2.1	1.8	4.1	6.8	4.3

mode control method performed the worst, with a yaw angle deviation of 2.0°, a position control error of 3.5 mm, and a maximum tracking error of 6.2 mm. In the Y-axis direction, the proposed method demonstrated the best tracking control performance, with a yaw angle deviation of 0.9°, a position control error of 1.5 mm, and a maximum trajectory tracking error of 2.3 mm. The model predictive fault-tolerant control exhibited a yaw angle deviation of 1.5°, a position control error of 2.6 mm, and a maximum tracking error of 4.3 mm. In contrast, the power-law sliding mode control achieved a yaw angle deviation of 2.3°, a position control error of 4.1 mm, and a maximum tracking error of 6.8 mm. This demonstrates that the proposed control method maintains minimal control deviation and the most stable tracking performance in different motion directions, showcasing its superior control capabilities in complex underwater environments.

In comparison with recent representative works, the proposed online fault compensation control method based on decoupling algorithm shows significant performance advantages in the experimental results of the study. Compared with Yuan et al.'s finite time adaptive fault-tolerant control, the research method not only achieved higher trajectory tracking accuracy, but also maintained fast response under composite faults. The system could approach and stabilize at the desired attitude within 50 s. Compared with the sliding mode control based on fixed time disturbance observer proposed by Li et al., the research method achieved smaller steady-state error in yaw angle control, and in simulations with model uncertainty and external water flow interference, the position control error was reduced to 1.2 mm, which is much lower than the comparative method. Compared with the hybrid gain adaptive specified performance control proposed by Zhang et al., this method significantly improved the intuitiveness of operation and the flexibility of dual arm coordination through attitude and position decoupling design. At the same time, by combining H ∞ robust control and online adaptive strategy, the maximum trajectory tracking error in both X and Y directions was significantly lower than that of model predictive fault-tolerant control and power sliding mode control. These results fully show that the intelligent control strategy proposed by the research, which combines decoupling architecture and real-time fault compensation, has obvious progressiveness and practicability in improving the operating accuracy, robustness and operating efficiency of ROV in complex underwater environment.

4 Summary and future work

To achieve high-precision operation and fault compensation control of ROVs in complex environments, this study proposed an online fault compensation control method based on a decoupling algorithm. The new method achieved collaborative control of the master and slave manipulators through a pose decoupling algorithm and constructed an observer-based fault diagnosis and compensation mechanism, employing H ∞ robust control theory to suppress external disturbances. Results showed that in a dual-arm collaborative operation experiment, the actual X-axis deviation of the right arm was 2.643 mm, the planned Y-axis deviation reached 3.075 mm, and the overall spatial deviation was 4.251 mm, while the actual spatial deviation of the left arm was 4.035 mm, all controlled within an error range of 4.3 mm. In comparison with different control methods, the proposed method exhibited a yaw angle deviation of only 0.8° in the X-axis direction, a position control error of 1.2 mm, and a maximum trajectory tracking error of 2.1 mm, significantly outperforming model prediction fault-tolerant control and power sliding mode control. Under fault conditions, the maximum deviation of the yaw angle control proposed in this paper was only 0.01°, and the pitch angle deviation did not exceed 0.03°, and it approached the desired attitude within 50 s. This demonstrates that the proposed decoupling algorithm and online fault compensation mechanism can effectively improve the operational accuracy and robustness of the ROV. While the study verifies the control effect under dual sensor and thruster faults, the impact of environmental factors such as different current intensities and visibility has not yet been analyzed. Therefore, future research will expand to multi-condition verification. Furthermore, the current system's adaptability to sudden multiple faults still has limitations. Therefore, future development requires a deep learning-based intelligent fault diagnosis and adaptive parameter adjustment module. Although the proposed method has demonstrated good control accuracy and fault tolerance in simulations and specific tasks, it still has certain limitations: firstly, the research verification environment is relatively ideal, and the impact of complex dynamic disturbances such as time-varying ocean

currents and low visibility in real oceans has not been fully considered. Secondly, the current system still lacks adaptability to extreme working conditions where multiple sudden failures occur simultaneously in sensors and thrusters. Finally, the experimental scenario is relatively single and has not yet covered practical engineering constraints such as communication delay and packet loss. The fault diagnosis mechanism's ability to identify unknown or intermittent fault modes also needs to be improved.

Data availability statement

The original contributions presented in the study are included in the article/supplementary material, further inquiries can be directed to the corresponding author.

Author contributions

NZ: Writing – review and editing, Writing – original draft.

Funding

The author(s) declared that financial support was not received for this work and/or its publication.

Conflict of interest

The author(s) declared that this work was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Generative AI statement

The author(s) declared that generative AI was not used in the creation of this manuscript.

Any alternative text (alt text) provided alongside figures in this article has been generated by Frontiers with the support of artificial intelligence and reasonable efforts have been made to ensure accuracy, including review by the authors wherever possible. If you identify any issues, please contact us.

Publisher's note

References

Adam, A. H. A., Chen, J., Xu, M., Kamel, S., Mustafa, G. I., Zaki, Z. A., et al. (2024). Power decoupling enhancement of a triple active bridge converter with feedforward compensation based on model predictive control and fuzzy logic controller in DC microgrid systems. *IEEE Access* 12 (1), 140310–140328. doi:10.1109/ACCESS.2024.3469815

Chen, L., Liu, J., Ding, S., Zhao, J., Gao, J., and Chen, H. (2025). Decoupling control of fuel cell air supply system based on data-driven feedforward and adaptive generalized supertwisting algorithm. *IEEE Trans. Circuits Syst. I Regul. Pap.* 72 (7), 3606–3619. doi:10.1109/TCSI.2025.3526144

Deng, L., Luo, S., He, C., and Wu, H. (2024). Underwater small and occlusion object detection with feature fusion and global context decoupling head-based YOLO. *Multimed. Syst.* 30 (4), 208–209. doi:10.1007/s00530-024-01410-z

Hu, J., Zhang, J., Zheng, Y., Wang, B., Wu, J., Wang, X., et al. (2024). Hybrid-driven continuum robot with decoupled motion patterns for dexterous manipulation. *IEEE Robotics Automation Lett.* 9 (9), 7437–7444. doi:10.1109/LRA.2024.3427557

Lang, J., Tong, C., Zheng, Y., Bai, J., and Zheng, P. (2024). Decoupled dead-time compensation method using revised-resonant control-based disturbance observer in PMSM drives. *IEEE Trans. Power Electron.* 40 (1), 340–350. doi:10.1109/TPEL.2024.3458406

Li, W., Zhang, Z., Wang, P., and Wan, H. (2025). A faster fixed-time fault tolerant sliding mode control for robot manipulators with mismatched disturbances. *IEEE Access* 24 (6), 56576–56586. doi:10.1109/ACCESS.2025.3554177

Liao, Y., Ning, D., Liu, P., Du, H., and Li, W. (2023). Decoupled semiactive vibration control of electrically interconnected suspension based on disturbance compensation. *IEEE Trans. Industrial Electron.* 71 (7), 7684–7694. doi:10.1109/TIE.2023.3303647

Lin, S., Cao, Y., Wang, Z., Yan, Y., Shi, T., and Xia, C. (2023). Speed controller design for electric drives based on decoupling two-degree-of-freedom control structure. *IEEE Trans. Power Electron.* 38 (12), 15996–16009. doi:10.1109/TPEL.2023.3314468

Qiu, Y., Zhang, C., Hametner, C., Zeng, T., Ferrara, A., Wang, Y., et al. (2024). Coupling mechanism analysis and decoupling control of the air supply system for fuel cell engine in fuel cell vehicle. *IEEE Trans. Transp. Electrification* 10 (4), 10059–10072. doi:10.1109/TTE.2024.3368627

Ren, L., Ren, S., Wang, H., Zhao, Y., Xu, J., and Zhong, Z. (2023). Cascade hybrid decoupling control scheme for synchronous reluctance motor current dynamic with adaptive backstepping disturbance compensation. *IEEE J. Emerg. Sel. Top. Power Electron.* 11 (5), 4595–4605. doi:10.1109/JESTPE.2023.3290548

Sengolrajan, T., Kalaivani, C., Ashok, J., and Manikandan, A. (2023). A novel design of 9 level Cascade multi-level inverter for decoupled double synchronous reference frame in state delay controller. *J. Eng. Res.* 11 (3), 143–157. doi:10.1016/j.jer.2023.100106

Shen, Y., Zakzewski, D., Hasnain, A., Resalayyan, R., and Khaligh, A. (2023). Sensorless control for DC-parallel active power decoupling in PV microinverters. *IEEE Trans. Power Electron.* 38 (11), 14628–14637. doi:10.1109/TPEL.2023.3306281

Wang, Y., Dong, L., Chen, Z., Sun, M., and Long, X. (2024). Integrated skyhook vibration reduction control with active disturbance rejection decoupling for automotive semi-active suspension systems. *Nonlinear Dyn.* 112 (8), 6215–6230. doi:10.1007/s11071-024-09337-w

Weber-Lewerenz, B. C., and Traverso, M. (2023). Navigating applied artificial intelligence (AI) in the digital era: how smart buildings and smart cities become the key to sustainability. *Artif. Intell. Appl.* 1 (4), 214–227. doi:10.47852/bonviewAIA32021063

Ye, Z., Huang, J., Mu, X., and Mao, Q. (2024). Multidimensional q-compensated reverse time migration using a high-efficient decoupled viscoacoustic wave equation. *Geophys. Prospect.* 72 (6), 2109–2122. doi:10.1111/1365-2478.13501

Yuan, Y., and Sun, W. (2025). Finite-time adaptive fault-tolerant control for robot manipulators with guaranteed transient performance. *IEEE Trans. Industrial Inf.* 8 (1), 3336–3345. doi:10.1109/TII.2024.3523592

Zhang, Z., Zhang, G., Wang, S., and Shi, C. (2023). Hysteresis modeling and compensation for tendon-sheath mechanisms in robot-assisted endoscopic surgery based on the modified Bouc-Wen model with decoupled model parameters. *IEEE Trans. Med. Robotics Bionics* 5 (2), 218–229. doi:10.1109/TMRB.2023.3249234

Zhang, J. X., Ding, J., and Chai, T. (2024a). Fault-tolerant prescribed performance control of wheeled mobile robots: a mixed-gain adaption approach. *IEEE Trans. Automatic Control* 69 (8), 5500–5507. doi:10.1109/TAC.2024.3365726

Zhang, J., Han, P., Wu, Z., Liu, Q., and Yang, J. (2024b). Nonfragile prescribed performance control of robot manipulators with actuator faults. *Int. J. Control, Automation Syst.* 22 (11), 3472–3481. doi:10.1007/s12555-024-0174-z

Zhou, J., Liu, C., Long, B., Zhang, D., Jiang, Q., and Muhammad, G. (2025). Degradation-decoupling vision enhancement for intelligent underwater robot vision perception system. *IEEE Internet Things J.* 12 (11), 17880–17895. doi:10.1109/JIOT.2025.3540033