

## Article

# Design and Reliability Analysis of a Tunnel-Detection AUV Based on a Heterogeneous Dual CPU Hot Redundancy System

Xiangbin Wang , Yushan Sun <sup>\*</sup>, Lei Wan, Hongyu Bian and Xiangrui Ran 

Science and Technology on Underwater Vehicle, Harbin Engineering University, Harbin 150001, China; uuvbin@163.com (X.W.); wanlei@hrbeu.edu.cn (L.W.); bianhongyu1116@hrbeu.edu.cn (H.B.); ranxiangrui@hrbeu.edu.cn (X.R.)

\* Correspondence: sunyushan@hrbeu.edu.cn

**Abstract:** A water conveyance tunnel is narrow and enclosed with a complex distribution of flow field. The performance of sensors such as Doppler log, magnetic compass, sonar, and depth gauge used by conventional underwater vehicles in the tunnel is greatly affected and can even fail. Aiming at the special operating environment and operational requirements of water conveyance tunnels, this paper designed an architecture suitable for pressurized water conveyance tunnel-detection autonomous underwater vehicles (AUVs). The tunnel-detection AUV (called AUV-T in this paper) with the architecture proposed in this paper could easily and smoothly complete inspection tasks in water conveyance tunnels, and field tests have verified the effectiveness of the architecture. Since an AUV in a water conveyance tunnel cannot go to the surface to rescue itself, in order to ensure its safety we designed the heterogeneous dual-CPU (Central Processing Unit) hot redundancy system based on dual communication lines. The reliability analysis showed that the system can significantly reduce the probability of AUV failure and ensure that the AUV can still be recovered even if it fails in the tunnel.

**Keywords:** autonomous underwater vehicles; dual CPU hot redundancy; water conveyance tunnel; Markov



**Citation:** Wang, X.; Sun, Y.; Wan, L.; Bian, H.; Ran, X. Design and Reliability Analysis of a Tunnel-Detection AUV Based on a Heterogeneous Dual CPU Hot Redundancy System. *Electronics* **2021**, *10*, 22. <https://doi.org/10.3390/electronics10010022>

Received: 12 November 2020

Accepted: 22 December 2020

Published: 25 December 2020

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

Water conveyance tunnels are commonly used in water conservancy projects and play a vital role in urban domestic water transportation. However, there are many uncertain factors in the construction process of water conveyance tunnels, especially some long-distance and deeply buried water conveyance tunnels. After running for a certain time, tunnels are likely to suffer from diseases of different degrees [1]. Therefore, it is necessary to conduct regular inspections of water conveyance tunnels. The traditional type of drainage inspection needs to stop water delivery and empty the tunnel, which leads to a huge workload. An autonomous underwater vehicle (AUV) can be used to detect a water conveyance tunnel at any time without changing the working state of the water conveyance tunnel, which not only saves a lot of manpower, material resources, and financial resources but also is convenient and flexible and does not affect urban domestic water, which is the development direction of tunnel detection in the future.

The environment in pressurized water conveyance tunnels is narrow, enclosed, and unknown, with the characteristics of complex flow field distribution, harsh environment, low light levels, and insignificant environmental characteristics. In the water conveyance tunnels, the performance of conventional sensors, such as Doppler log, magnetic compass, sonar, and depth gauge, will be greatly affected or even fail. If the conventional design idea is adopted, it will be impossible to complete the detection task in the water conveyance tunnel. On the other hand, the enclosed environment prevents the AUV from adopting the conventional self-rescue measures of floating in case of emergency. As we all know, AUVs are high-cost, and if they break down in a tunnel and cannot be recovered they will cause a

great loss. For this special operating environment, it is necessary to design an AUV with a unique working mode and ensure its operation safety to the maximum extent.

Glushko et al. [2] discussed software architecture for controlling a vehicle actuator system; the paper introduced an additional abstraction level to reduce the complexity of controlling unconventional actuators. Tipsuwan and Hoonsuwan [3] described a design and implementation of an AUV for petroleum pipeline inspection. The AUV is controlled by a 6-degrees-of-freedom PID (Proportion Integral Differential) controller. The software structure of the AUV is based on a Robot Operation System. The AUV was under testing when the paper was published. There is a big difference between the working environment of a pipeline inspection AUV and that of a tunnel inspection AUV. Conventional sensors fail in water conveyance tunnels, and the above architecture is not applicable to the AUV-T. Pidić et al. [4] presented an AUV for the inspection of water-filled tunnels. A two-dimensional height vs. time plot of the tunnel was produced. A proof-of-concept for traversing water-filled tunnels using water flow and variable buoyancy was presented in the paper. The data were collected in a scaled-down version of the tunnel. The AUV could only obtain a simple dimensional height vs. time plot, which will be compared with tunnel schematics if it cannot meet the actual detection needs. Jia et al. [5] designed a model predictive path tracking controller for a tunnel-tracking AUV. The experiment was conducted in a canal to prove the effectiveness of the proposed controller. However, the controller was only verified in the canal experiment and was not suitable for the narrow water conveyance tunnel environment. Freire et al. [6] achieved a reasonable level of safety and reliability through the enforcement of coding standards in C language. Besides this, the level of reliability was provided by the ruggedness and simplicity of the components in hardware aspects. Das et al. [7] described the control architecture required for the autonomous navigation and guidance control of AUV-150. The control architecture was presented as an ensemble of both hardware and software modules. The leak detection unit was used in the AUV. If any leakage occurs, the computer will shut down the entire system. The reliability of the above-mentioned system was only improved theoretically, but the reliability of the control computer was not substantially improved from the hardware. Wang et al. [8] designed triple-redundancy hardware and software control architectures for dynamically positioned ships and rigs. The hardware-redundant configuration included three computers which connect to each other through dual networks. The simulation was carried out on a drilling rig. Wang et al. [9] proposed the triple-redundant hardware architecture and hierarchical software for the DP control system. The model test together and the HIL (hard-in-the-loop) test were conducted. The computer and dual networks in the system are homogeneous, without considering the actual situation.

In this paper, considering the environmental characteristics of water conveyance tunnel and the difficulties in inspection work, an AUV architecture for water conveyance tunnel detection and a heterogeneous dual-CPU hot redundancy system based on dual communication lines was proposed. The design of two control computers could reduce the probability of AUV failure and could guarantee the recovery of the AUV in the event of failure in the tunnel to the maximum extent.

The rest of this paper is organized as follows. Firstly, the second chapter introduces the hardware structure and architecture design of the AUV-T. This chapter highlights the key design challenges and how they are proposed to be tackled. The third chapter introduced the structure of the dual-CPU hot redundancy system and analyzes its reliability. Chapter four introduces and analyzes the results of the inspection test in the water conveyance tunnel. Chapter five includes a summary of our findings.

## 2. Materials and Methods

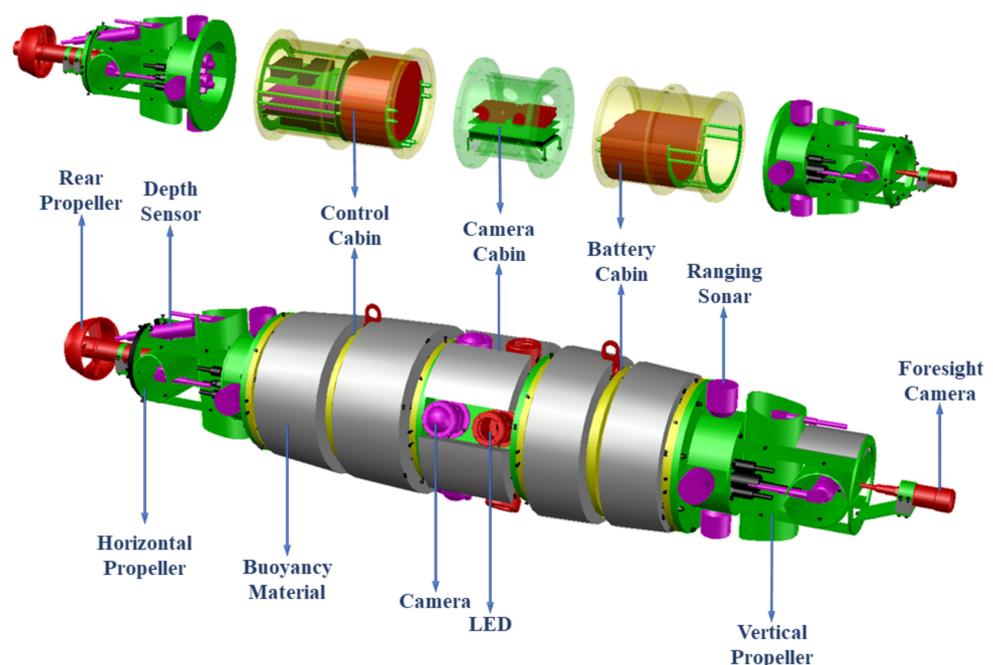
### 2.1. Hardware Composition

As shown in Figure 1, the AUV-T is bound to collide with walls due to sailing in the narrow tunnel. Therefore, the exterior is wrapped in a carbon fiber shell and multiple sets of collision avoidance bars have been added to the shell to ensure the safety of the AUV.



**Figure 1.** Tunnel-detection AUV (AUV-T).

The main body of the AUV-T consists of three watertight compartments—namely, a control cabin, a camera cabin, and a battery cabin—outside which annular buoyancy material is equipped, as shown in Figure 2. The AUV-T is equipped with a main thruster at the stern and two vertical and two horizontal thrusters for the control of surge, sway, heave, pitch, and yaw motion. A high-precision depth gauge and four ranging sonars are equipped in the stern, four ranging sonars are equipped around the bow, and five lights and five HD cameras are equipped in the camera cabin in a circular distribution. The cameras are mainly responsible for shooting the video of the wall in the tunnel. The brightness of the lights is adjusted by a PWM (pulse width modulation) signal to provide a light source for the camera. In addition, a forward-looking camera is equipped at the bow of the AUV to recognize the optical signals of the branch hole and assist the AUV in exiting the hole.



**Figure 2.** Structure of the AUV-T.

The navigation, fault diagnosis, and wall following the control of AUV-T is controlled by an onboard PC104 Pentium III-class computer (Computer\_Ctrl, Sheng-Bo Technology

Corp., Shenyang, China) which is configured in the control cabin. Fiber-optic gyro (Fizoptika, Mosta, Malta), TCM5 (PNI Corporation, Santa Rosa, CA, USA), and leak detection sensors (Huakongxingye, Beijing, China) are also located in the control cabin.

There are three video processing PC104 (Computer\_Video1, Computer\_Video1, Computer\_Video3, Sheng-Bo Technology Corp., Shenyang, China) in the camera cabin. Computer\_Video1 is responsible for processing the data captured by circumferentially distributed cameras 1 and 2 and communicating with Computer\_Ctrl regularly according to the communication protocol to ensure the operation of the dual-CPU hot redundancy system. Computer\_Video2 is responsible for processing the video data of circumferentially distributed cameras 3, 4, and 5. Computer\_Video3 is responsible for processing the forward-looking camera data and communicating with Computer\_Ctrl, sending the command of driving out of the main tunnel to Computer\_Ctrl when an exit signal is detected. In addition, a solid storage device for data storage is equipped in the camera cabin. Other parameters of the AUV-T are shown in Table 1.

**Table 1.** The parameters of the AUV-T.

Parameters	Value
Length	2060 mm
Maximum Diameter	400 mm
Weight	174.6 kg
Maximum Depth	200 m
Cruising Speed	$\leq 2$ m/s
Endurance Time	8 h
Video Storage Time (1080 p/30 fps)	$\geq 10$ h
Sensors	Fiber Optic Gyroscope (FOG)
	TCM5 magnetic compass
	Water Leak Sensor
	Power Supply Monitoring
	Ranging Sonar
	Depth Sensor
	Video Camera
Operating System	VxWorks5.5 (bottom layer) (Wind River Systems Inc., Alameda, CA, USA)
	Windows 7 (surface layer) (Microsoft Corporation, One Microsoft Way Redmond, WA, USA)

## 2.2. Architecture Design

The architecture of the AUV-T adopts the idea of layered design, which is divided into three levels from the bottom to the top: execution level, task level, and management level [10], as shown in Figure 3. The management level consists of three systems: a navigation system, a mission system and a motion control system.

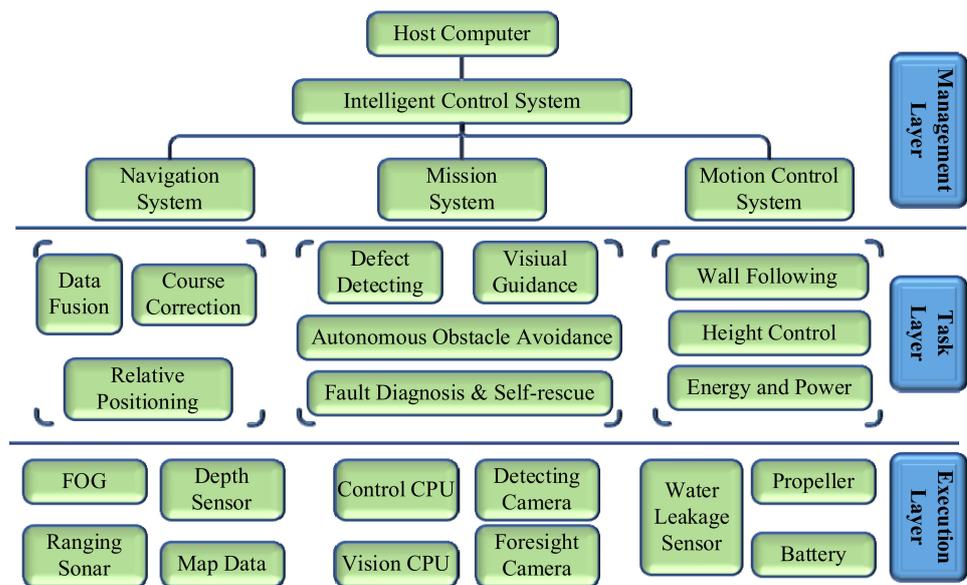


Figure 3. The layered structure.

According to the special requirements of the tunnel detection task, the task layer is divided into ten modules.

(1) Data fusion.

AUV-T is equipped with ranging sonars on the fore and stern. In order to avoid echo interference from sonars in narrow tunnels [11], eight ranging sonars adopt a weighted round robin mode, so it is impossible to obtain all the sonar data at the same time. The data fusion module performs time registration on the sonar data in the same direction. First, the least square method is used to estimate the distance data in the direction where no echo is obtained at the current moment, and then the data in the direction is fused. Through the sensors carried by the AUV-T, important data such as the width, height, depth, heading, and wall video of different positions of the water conveyance tunnel can be obtained, which provides an important reference for the detection of the water conveyance tunnel.

(2) Course correction.

Since the water conveyance tunnel is a reinforced concrete structure, the magnetic compass cannot work normally in the tunnel and so the fiber optic gyroscope is used to obtain the heading change of the AUV. However, the fiber optic gyroscope will have a drift problem under continuous operation [12]. The course correction module uses the fusion of magnetic compass data and ranging sonar data to generate an estimate of the heading change, thereby completing the correction of the drift of the fiber optic gyroscope.

(3) Relative positioning.

Conventional navigation methods deal with the problem of failure in the tunnel. Therefore, using the known terrain feature data of the tunnel, combined with real-time heading, depth, and ranging sonar data, the relative positioning is found based on the entrance to obtain the relative position of the AUV in the tunnel.

(4) Wall defect detection.

The five circumferential cameras on the AUV-T are used to take pictures of the tunnel wall in real time. Then, the system performs real-time crack detection in the video data and uses the relative positioning data to mark the current crack position. When the video data need to be transmitted, the AUV can be connected via WIFI or optical fiber for data transmission. The WIFI module adopts 802.11 n protocol and the transmission speed can reach 150 Mbps. The transmission speed of the optical transceiver and router in the

optical fiber communication is 1000 Mbps, which can meet the demand for high-speed data transmission.

(5) Autonomous obstacle avoidance.

As shown in Figure 4, the obstacle avoidance in the water conveyance tunnel mainly refers to: First, avoiding the vessel meeting holes in the tunnel to prevent the AUV from entering the meeting hole, affecting the relative positioning of the AUV and even causing damage to the AUV. The second is to avoid non-outlet branches. Since there are many branch exits in the water conveyance tunnel and the AUV needs to be recovered from the designated branch exits [13], it is necessary to identify the branch exits based on the positioning data [14] to avoid driving out from the wrong branch exit and affecting the completion of the work task and the AUV recycling.

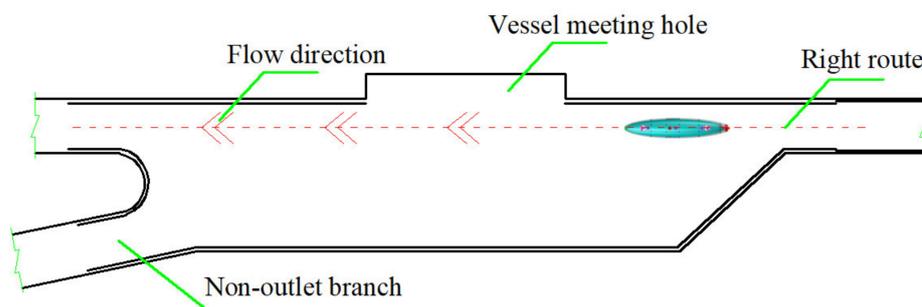


Figure 4. Schematic diagram of obstacle avoidance.

(6) Visual guidance.

In order to prevent the AUV from missing the exit and to ensure that the AUV can be successfully recovered, a light source with a specified frequency is placed at the exit. The light blinks according to the preset color and frequency. Red and blue blink alternately, the frequency is 1 Hz, and the frequency can be adjusted according to the actual situation. The AUV-T uses the forward-looking camera onboard to identify the light source of the specified frequency and exit the tunnel.

(7) Fault diagnosis and self-rescue.

The AUV operating in the pressurized water tunnel is in a closed and confined environment. When a failure or danger occurs, the conventional self-rescue measures of releasing the ballast from floating cannot be adopted [15]. In order to ensure the safety of the AUV-T in tunnels, it is extremely necessary to study reliable fault diagnosis and self-rescue algorithms. Utilizing multiple CPUs that can communicate with each other in the AUV-T, this paper proposes a heterogeneous dual-CPU hot redundancy system based on dual lines, the details of which will be introduced in Chapter 3.

(8) Wall following.

In order to shoot a clear video, the AUV-T needs to maintain a fixed distance from the tunnel wall [16]. The AUV-T transforms the wall following into heading control by adjusting the distance between the heading and the wall [17]. The backstepping method is used to design an adaptive sliding mode controller to control the heading [18].

The kinematic model of heading motion is as follows:

$$\dot{\psi} = r. \tag{1}$$

The dynamic model [19] is as follows:

$$\dot{r} = \frac{\tau_N}{I_z - N_r} - \frac{(X_{\dot{u}} - Y_{\dot{v}})uv + f_N}{I_z - N_r} + \frac{(N_r + N_{r|r}|r|)r}{I_z - N_r} \tag{2}$$

The S-plane function is:

$$s = k_2 z_1 + z_2, k_2 > 0 \quad (3)$$

Adaptive law is:

$$\dot{\hat{f}} = \lambda s \quad (4)$$

The controller is:

$$\tau_N = \frac{1}{b} [-k_2(z_2 - k_1 z_1) - f(x_2) + \dot{x}_{2d} - k_3 s - k_4 \text{sat}(s) - \hat{f}] \quad (5)$$

where  $\psi$  is the heading angle;  $r$  is the heading angular velocity;  $\tau_N$  is the moment about the GZ-axis;  $u, v$  are linear velocities;  $f_N$  is the external disturbance moment;  $I_Z, N_{\dot{r}}, X_{\dot{u}}, Y_{\dot{v}}, N_{r|r}$  are the hydrodynamic coefficients;  $b = 1/(I_Z - N_{\dot{r}})$ ,  $f(x_2) = (N_{\dot{r}} + N_{r|r}|r|)r/(I_Z - N_{\dot{r}})$ ,  $f = -((X_{\dot{u}} - Y_{\dot{v}})uv + f_N)/(I_Z - N_{\dot{r}})$ , and  $k_1, k_2, z_1, z_2, \lambda$  are the control parameters;  $\dot{x}_2 = f(x_2) + b\tau_N + \dot{f}$ ;  $x_{2d}$  is the expected value of  $x_2$ ;  $s$  is the sliding surface function, which can be expressed as  $s = k_2 z_1 + z_2$ ;  $\hat{f}$  is the estimated value of  $f$ .

#### (9) Height control.

There is water pressure in the water conveyance tunnel, and the depth sensor obtains the depth of the AUV from the free surface. In order to maintain the stability of the distance between the AUV and the wall in the vertical direction, an adaptive S-plane controller is designed [20]. The distance between the AUV and the bottom of the tunnel obtained by the ranging sonar is used as the height value to input into the controller so as to realize the control of the distance between the AUV and the bottom of the tunnel.

The control model of S-plane [21] is:

$$\tau_Z = 2.0/(1.0 + \exp(-k_1 e - k_2 \dot{e})) - 1.0 + \Delta\tau_Z \quad (6)$$

where  $\tau_Z$  is the output value of the height controller.  $k_1, k_2$  are the controller parameters.  $e, \dot{e}$  are the height deviation and deviation change rates.  $\Delta\tau_Z$  is the adaptive adjustment term, which is defined as:

$$\Delta\tau_Z = \begin{cases} 2.0/(1.0 + \exp(-k_3 \Delta e)) - 1.0 & 0 < \dot{e} < \dot{e}_{\max} \\ 0 & \dot{e} \geq \dot{e}_{\max}, \text{ or } \dot{e} \leq 0 \end{cases} \quad (7)$$

where  $\dot{e}_{\max}$  is the deviation rate threshold, which is used to control the adjustment quantity.  $\Delta e$  is the height deviation adaptive adjustment quantity, which is defined as:

$$\Delta e = (1 - \gamma)e_t + \gamma\eta_t \sum_{i=t-6}^t e_i \quad (8)$$

where  $\gamma$  is the parameter of low-pass filter.  $e_t$  is the height deviation at time  $t$ .  $\eta_t$  is the instant elimination factor at time  $t$ , which is defined as following:

$$\eta_t = \begin{cases} 0.25\eta_{t-1} & \Delta e_t \cdot e_t < 0 \\ 1 & \Delta e_t \cdot e_t \geq 0 \end{cases} \quad (9)$$

The distance between the AUV and the bottom of the tunnel varies due to the gradient of the tunnel. The integral part of the adaptive adjustment item can improve the response ability of the controller to a small deviation and improve the response to the height control. The design of weakening link can improve the control shock brought by an integral term. A low-pass filter is added to smooth the output of the height controller.

#### (10) Energy and power.

To meet the needs of long-term detecting tasks, two rechargeable Li-ion batteries are equipped in AUV-T, as shown in Figure 5. Battery1 is located in the control cabin

with an output voltage of 24 V and is used to power sensors, cameras, LEDs, and PC104. Battery2 is located in the battery cabin with a 48 V output and is used to power five thrusters. Both batteries are equipped with an intelligent detection system, which can output battery voltage, current, temperature, SOC (state of charge), and other information to the control system in real time. When the SOC value of one of the batteries reaches the critical threshold, AUV-T will stop the detection task and go out the tunnel nearby to ensure that the AUV can complete the recovery safely.

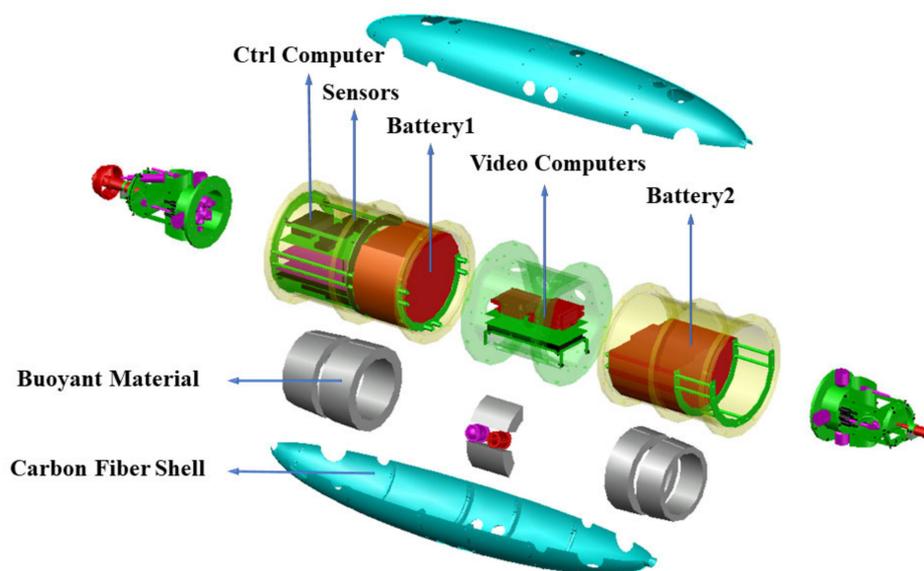


Figure 5. Energy and power system of the AUV-T.

### 3. Results

#### 3.1. Multi CPU Hot Redundancy System Modeling

In order to find an optimal multi-CPU hot redundancy system suitable for the AUV-T, a mathematical model of the multi CPU hot redundancy system was established, as shown in Figure 6. System “a” consists of two CPUs, and system “b” consists of three CPUs, and so on.

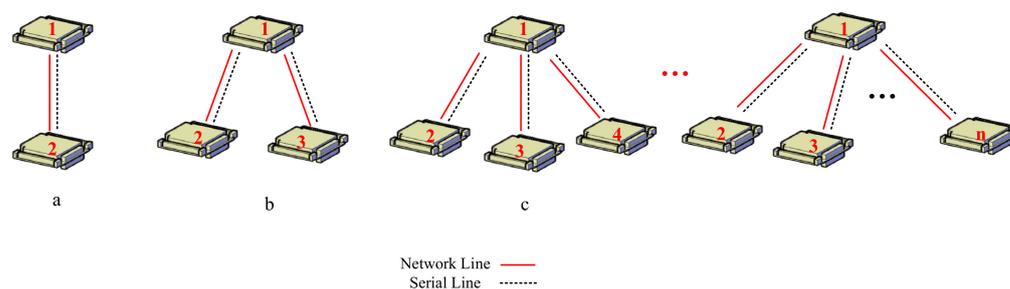


Figure 6. Model of a multi-CPU hot redundancy system.

There is a main CPU (number 1) and multiple standby CPUs (number 2 to  $n$ ) in each system, and each standby CPU communicates with the main CPU through the dual-communication line (including a network line and a serial line). To simplify the calculation, it is assumed that the failure rate of each CPU is the same, which is  $\alpha$ , and the failure rate of the network line and the serial line are the same, which is  $\beta$ . Taking the actual situation into account, the following assumptions are also made.

- (1) When the PC104 fails, the serial communication fails, or the network communication fails, it is considered that the module fails. When the module fails, it cannot recover by itself.
- (2) The working status of the system is divided into three types:
  - ① Normal working state ( $R_1$ ): All modules are working normally, or the network line or serial line in one or more dual-communication fails (that is, there is no failure of both the network line and the serial line in a dual-communication line). At this time, normal communication can still be guaranteed between all CPUs, and the AUV performs tasks normally.
  - ② Communication failure status ( $R_2$ ): One or more dual-communication lines fail (both the network line and serial line fail), or one of the CPU fails. At this time, although the AUV can accept control commands and work normally, some CPUs cannot perceive the status of each other. To ensure safety, the AUV-T should exit the tunnel nearby. After exiting the tunnel, a manually check and fault restoring is needed.
  - ③ Fatal failure status ( $F$ ): As long as the CPU can start the nearby exit program, it is considered that the system has no fatal failure. Only when all the CPUs fail is it considered that the entire system has a fatal failure. At this time, the AUV loses control and sails in an unknown state. The failure of this mission may cause the AUV to fail to recover or even cause damage to the AUV.

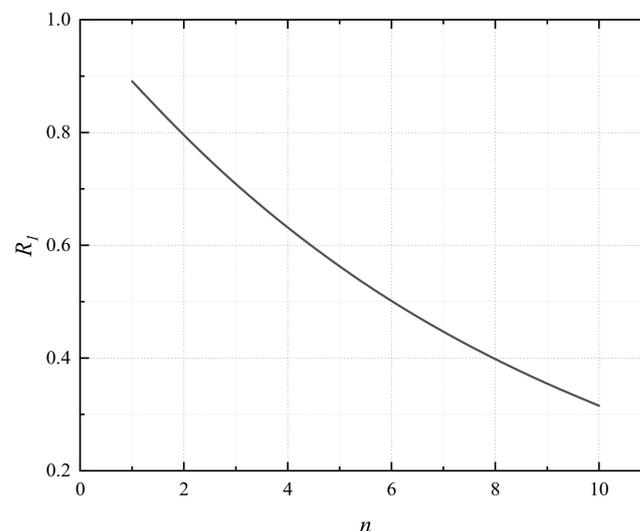
When there are  $n$  ( $n \geq 2$ ) CPUs in the system, the probability of the normal working state of the system can be obtained as:

$$R_1 = (1 - \alpha)^{n-1}(1 - \beta)^{n-1} \quad (10)$$

The probability of a fatal failure in the system is:

$$F = \alpha^n \quad (11)$$

When  $\alpha = 0.1$ ,  $\beta = 0.01$  the curve of  $R_1$  and  $n$  is shown in Figure 7.



**Figure 7.** The curve of  $R_1$  and  $n$ .

Obviously, the increase in the number of CPUs ( $n$ ) will reduce the probability of normal working ( $R_1$ ), but it will also reduce the probability of fatal failures of the system.

Taking the following factors into account:

A fatal failure of the system is a rare event.

An AUV can only perform tasks normally when it is in an  $R_1$  state, so increasing the probability of  $R_1$  should be given priority.

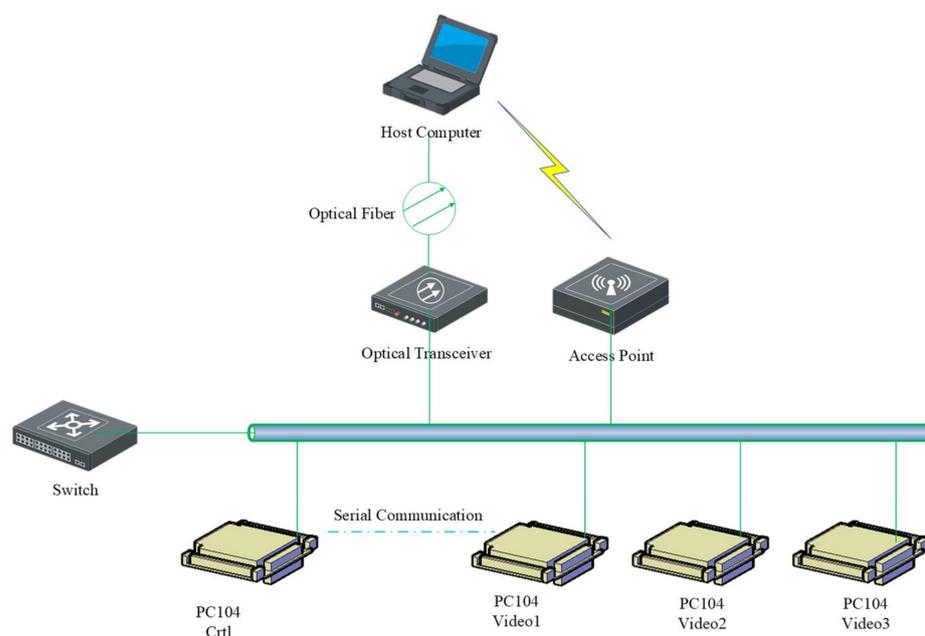
The AUV needs at least two CPUs (Computer\_Ctrl and Computer\_Video, respectively) to perform inspection tasks normally.

Computer\_Ctrl needs to communicate with each other CPUs simultaneously with the network and serial lines. Each communication method requires the establishment of two independent task processes. The increase in CPU will increase the operation load of Computer\_Ctrl, thereby further increasing its failure rate.

Considering the above factors comprehensively, combined with the actual operation requirements of AUV, it is finally decided to adopt the dual CPU hot redundancy system as the optimal multi CPU hot redundancy system for AUV-T.

### 3.2. Modeling of Heterogeneous Dual-Line Dual-CPU Hot Redundancy System

The heterogeneous dual-line dual-CPU hot redundancy system is divided into four modules, as shown in Figure 8, including two different CPUs (Computer\_Ctrl, Computer\_Video), one serial line, and one network line. The two CPUs communicate through the network and serial lines.



**Figure 8.** Model of the heterogeneous dual-line dual-CPU hot redundancy system.

Computer\_Ctrl as well as Computer\_Video are connected by a serial line. If there is any congestion or bottleneck on the network line, it helps the computers to stay connected. Both the CPUs execute a health checkup routine in parallel. If an emergency happens such as low battery power, the breakdown of the internal ethernet bus, the malfunctioning of navigational sensors, or some program execution fault, any or both of them executes the same emergency routine.

Two CPUs run a VxWorks system and run the same program. The CPU receives the same input data and establishes fault diagnosis tasks with a high priority. This task is responsible for real-time serial communication and network communication between two PC104 (TCP/IP, Computer\_Ctrl is the server and Computer\_Video is the client). Computer\_Video continuously updates the internal variable value according to the communication content. In the initial state, the output status bit of Computer\_Ctrl is true, and the output status bit of Computer\_Video is false—that is, the AUV finally executes the control command from Computer\_Ctrl. When the fault diagnosis system determines that a fault has occurred, Computer\_Video sets the output status bit to true and executes the nearest exit task in the current state.

The Markov model is proposed to analyze the reliability of the system under the assumption that any failure is independent. The transition probabilities are time invariant and depend only on the current state.

The interrelationship of failures in the dual-line heterogeneous dual-CPU hot redundancy system is shown in Figure 9.

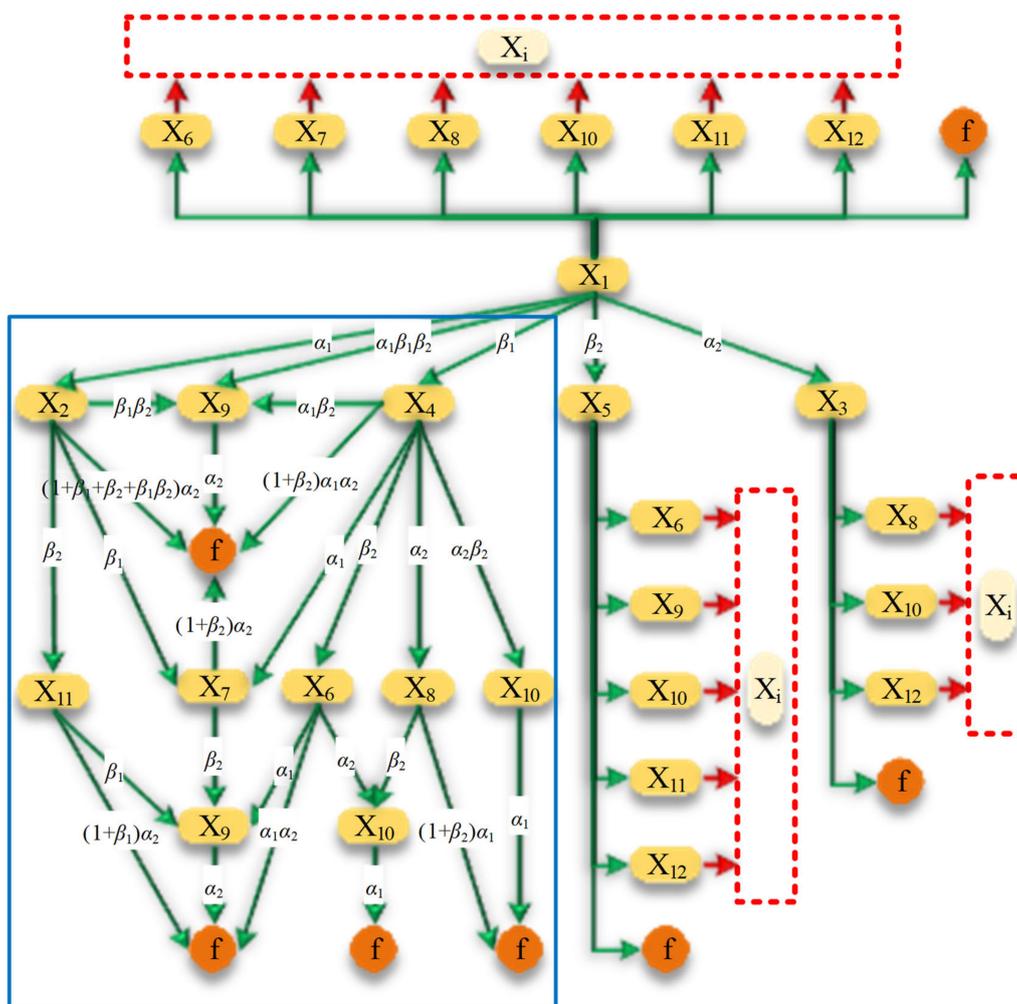


Figure 9. State transition of the Markov process.

Because the two CPUs have different models and working loads, the probability of failure is different. On the other hand, the communication mechanism between the serial line and the network determines that the probability of failure is different.  $\alpha_1$  is the constant failure rate of computer1,  $\alpha_2$  is the constant failure rate of computer2,  $\beta_1$  is the constant failure rate of the serial line, and  $\beta_2$  is the constant failure rate of the network line.  $X_i (i = 1, 2 \dots 12, 13)$ , ( $f = X_{13}$ ) is the process states of Markov.  $X_1$  is the initial normal working state of the system.  $X_2, X_3$  are the state of a single CPU failure.  $X_4, X_5$  are the state of a single communication line failure.  $X_6$  is the state in which both communication lines have failed.  $X_7, X_8, X_{11}, X_{12}$  are the state in which both a computer and a line have failed.  $X_9, X_{10}$  is the state in which both a computer and two communication lines have failed.  $f$  is the state of a fatal failure of the system. Among them,  $X_1, X_4, X_5$  belong to the normal working state, and  $X_2, X_3, X_6, X_7, X_8, X_9, X_{10}, X_{11}, X_{12}$  belong to the communication failure state. Figure 7 shows the transition probability and transition process between each state  $X_i$ . The blue box on the left completely displays the transition probability and transition

process between some states. The state transition process in the red dashed frame on the right and upper side is similar to that on the left, so it is omitted.

Therefore, the state transition probability matrix can be obtained:

$$P_{i_n, i_{n+1}} = p(X_{n+1} = s_{i_{n+1}} | X_n = s_{i_n})$$

$$= \begin{bmatrix}
 \gamma_1 & \alpha_1 & \alpha_2 & \beta_1 & \beta_2 & \beta_1\beta_2 & \alpha_1\beta_1 & \alpha_2\beta_1 & \alpha_1\beta_1\beta_2 & \alpha_2\beta_1\beta_2 & \alpha_1\beta_2 & \alpha_2\beta_2 & \chi_1 \\
 0 & \gamma_2 & 0 & 0 & 0 & 0 & \beta_1 & 0 & \beta_1\beta_2 & 0 & \beta_2 & 0 & \chi_2 \\
 0 & 0 & \gamma_3 & 0 & 0 & 0 & 0 & \beta_1 & 0 & \beta_1\beta_2 & 0 & \beta_2 & \chi_3 \\
 0 & 0 & 0 & \gamma_4 & 0 & \beta_2 & \alpha_1 & \alpha_2 & \alpha_1\beta_2 & \alpha_2\beta_2 & 0 & 0 & \chi_4 \\
 0 & 0 & 0 & 0 & \gamma_5 & \beta_1 & 0 & 0 & \alpha_1\beta_1 & \alpha_2\beta_1 & \alpha_1 & \alpha_2 & \chi_5 \\
 0 & 0 & 0 & 0 & 0 & \gamma_6 & 0 & 0 & \alpha_1 & \alpha_2 & 0 & 0 & \chi_6 \\
 0 & 0 & 0 & 0 & 0 & 0 & \gamma_7 & 0 & \beta_2 & 0 & 0 & 0 & \chi_7 \\
 0 & 0 & 0 & 0 & 0 & 0 & 0 & \gamma_8 & 0 & \beta_2 & 0 & 0 & \chi_8 \\
 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \gamma_9 & 0 & 0 & 0 & \chi_9 \\
 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \gamma_{10} & 0 & 0 & \chi_{10} \\
 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \beta_1 & 0 & \gamma_{11} & \chi_{11} \\
 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \beta_1 & 0 & \gamma_{12} & \chi_{12} \\
 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1
 \end{bmatrix} \tag{12}$$

where,  $p$  is the conditional probability,  $X = s_i, s_i \in \mathbf{s}$  is the state, and  $\mathbf{s}$  is the state space.

$$\begin{aligned}
 \chi_1 &= (1 + \beta_1 + \beta_2 + \beta_1\beta_2)\alpha_1\alpha_2, \\
 \gamma_1 &= 1 - (1 + \alpha_1 + \alpha_2)(\beta_1 + \beta_2 + \beta_1\beta_2) - \chi_1, \\
 \chi_2 &= (1 + \beta_1 + \beta_2 + \beta_1\beta_2)\alpha_2, \\
 \gamma_2 &= 1 - \beta_1 - \beta_1\beta_2 - \beta_2 - \chi_2, \\
 \chi_3 &= (1 + \beta_1 + \beta_2 + \beta_1\beta_2)\alpha_1, \\
 \gamma_3 &= 1 - \beta_1 - \beta_1\beta_2 - \beta_2 - \chi_3, \\
 \chi_4 &= (1 + \beta_2)\alpha_1\alpha_2, \\
 \gamma_4 &= 1 - \beta_2 - (\alpha_1 + \alpha_2)(1 + \beta_2) - \chi_4, \\
 \chi_5 &= (1 + \beta_1)\alpha_1\alpha_2, \\
 \gamma_5 &= 1 - \beta_1 - (\alpha_1 + \alpha_2)(1 + \beta_1) - \chi_5, \\
 \chi_6 &= \alpha_1\alpha_2, \\
 \gamma_6 &= 1 - \alpha_1 - \alpha_2 - \chi_6, \\
 \chi_7 &= (1 + \beta_2)\alpha_2, \\
 \gamma_7 &= 1 - \beta_2 - \chi_7, \\
 \chi_8 &= (1 + \beta_2)\alpha_1, \\
 \gamma_8 &= 1 - \beta_2 - \chi_8, \\
 \chi_9 &= \alpha_2, \\
 \gamma_9 &= 1 - \chi_9, \\
 \chi_{10} &= \alpha_1, \\
 \gamma_{10} &= 1 - \chi_{10}, \\
 \chi_{11} &= (1 + \beta_1)\alpha_2, \\
 \gamma_{11} &= 1 - \beta_1 - \chi_{11}, \\
 \chi_{12} &= (1 + \beta_1)\alpha_1, \\
 \gamma_{12} &= 1 - \beta_1 - \chi_{12}.
 \end{aligned}$$

The n-step transition probability in the Markov chain is:

$$P_{i_n, i_{n+1}}^{(n)} = p(X_n = s_{i_n} | X_0 = s_{i_0}) \tag{13}$$

Furthermore, from the Chapman–Kolmogorov equation, the n-step transition matrix is:

$$P^{(n)} = P^{(n-1)} \dots P^{(1)} I \tag{14}$$

There is no fault condition in the initial state—that is, the initial state distribution is:

$$X^{(0)} = [1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0] \tag{15}$$

$$p(t) = [p_1(t), p_2(t), p_3(t) \dots p_{13}(t)] = X^{(0)} \cdot P^{(t)} \tag{16}$$

According to Equations (12)–(16), we could get that the value of  $p_i(t)$ .  $p_1(t), p_2(t), p_3(t) \dots p_{13}(t)$  is the probability that the system is in state  $X_1, X_2, X_3 \dots X_{13}$  at time  $t$ . When the system is in the  $X_{13}(i.e.f)$  state, it is considered that a fatal fault has occurred, otherwise the system is considered to be a reliable state. Therefore, the fatal failure rate of the system at time  $t$  is defined as:

$$F(t) = p_{13}(t) \tag{17}$$

The probability that the system is in a reliable state is:

$$R(t) = \sum_{i=1}^{12} p_i(t) = 1 - F(t) \tag{18}$$

The system reliability status includes normal working status and the communication failure status. The probability of a normal working state is defined as:

$$R_1(t) = p_1(t) + p_4(t) + p_5(t) \tag{19}$$

The probability of a communication failure state is:

$$R_2(t) = p_2(t) + p_3(t) + p_6(t) + p_7(t) + p_8(t) + p_9(t) + p_{10}(t) + p_{11}(t) + p_{12}(t) \tag{20}$$

That is:

$$R(t) = R_1(t) + R_2(t) \tag{21}$$

### 3.3. Reliability Analysis

In order to verify the reliability of the heterogeneous dual-line dual-CPU hot redundancy system, let  $\alpha_1, \alpha_2, \beta_1, \beta_2$  take different values. As far as we know, there is no accurate value for the failure rate used in this paper. Therefore, the failure rate is set in accordance with empirical values and other papers [8]. In order to ensure the reliability of the data, we appropriately increased the failure rate in the paper. When  $t$  takes a different value, the values of  $p_i(t), R(t), F(t)$  change accordingly. According to Equations (16)–(20), we could calculate the probability of the system in different states under different working hours. The results are shown in Table 2. In Figure 10a–c represent three failure rate states from high to low respectively.

**Table 2.** Probability of the system in different states.

Failure Rate	$R(t)$ and $F(t)$	$t = 24$ h	$t = 72$ h	$t = 240$ h	$t = 720$ h
$\alpha_1 = 2 \times 10^{-2}/h$ $\alpha_2 = 1 \times 10^{-2}/h$ $\beta_1 = 1 \times 10^{-3}/h$ $\beta_2 = 5 \times 10^{-3}/h$	$F(t)$	0.0841	0.3999	0.9051	0.9993
	$R_1(t)$	0.4756	0.1062	0.0005	$1 \times 10^{-10}$
	$R_2(t)$	0.4403	0.4939	0.0944	0.0007
	$R(t)$	0.9159	0.6001	0.0949	0.0007
$\alpha_1 = 2 \times 10^{-3}/h$ $\alpha_2 = 1 \times 10^{-3}/h$ $\beta_1 = 1 \times 10^{-4}/h$ $\beta_2 = 5 \times 10^{-4}/h$	$F(t)$	0.0011	0.0094	0.0816	0.3925
	$R_1(t)$	0.9303	0.8050	0.4845	0.1122
	$R_2(t)$	0.0685	0.1856	0.4340	0.4953
	$R(t)$	0.9989	0.9906	0.9184	0.6075
$\alpha_1 = 2 \times 10^{-4}/h$ $\alpha_2 = 1 \times 10^{-4}/h$ $\beta_1 = 1 \times 10^{-5}/h$ $\beta_2 = 5 \times 10^{-5}/h$	$F(t)$	$1 \times 10^{-5}$	0.0001	0.0011	0.0093
	$R_1(t)$	0.9928	0.9786	0.9305	0.8055
	$R_2(t)$	0.0072	0.0213	0.0684	0.1852
	$R(t)$	0.99999	0.9999	0.9989	0.9907

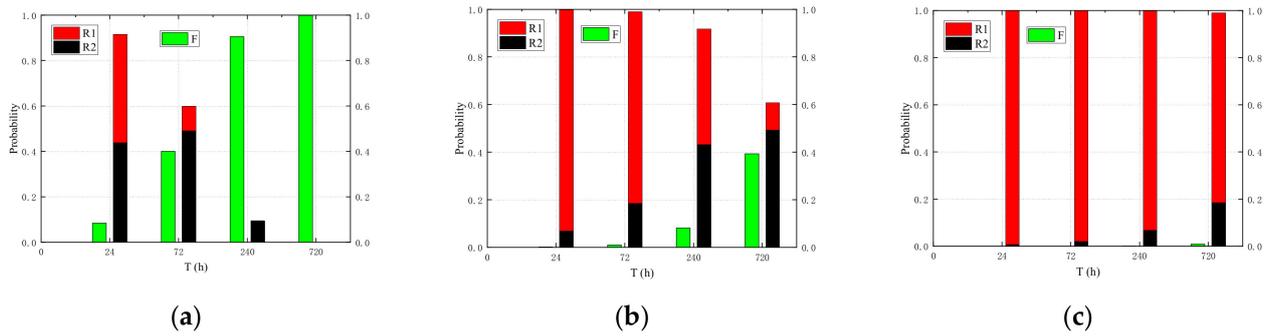


Figure 10. Probability of the system in different states.

On the other hand, the working hours of the system under different fatal failure probability can be calculated, as shown in Table 3. The results are shown in Figure 11.

Table 3. Working hours under different fatal failure probabilities.

Time/Failure Rate		$F(t) = 0.1$	$F(t) = 0.5$	$F(t) = 0.9$
$t(h)$	P1 $\alpha_1 = 2 \times 10^{-2}/h$ $\alpha_2 = 1 \times 10^{-2}/h$ $\beta_1 = 1 \times 10^{-3}/h$ $\beta_2 = 5 \times 10^{-3}/h$	27	90	236
	P2 $\alpha_1 = 2 \times 10^{-3}/h$ $\alpha_2 = 1 \times 10^{-3}/h$ $\beta_1 = 1 \times 10^{-4}/h$ $\beta_2 = 5 \times 10^{-4}/h$	272	908	2380
	P3 $\alpha_1 = 2 \times 10^{-4}/h$ $\alpha_2 = 1 \times 10^{-4}/h$ $\beta_1 = 1 \times 10^{-5}/h$ $\beta_2 = 5 \times 10^{-5}/h$	2721	9086	23,827

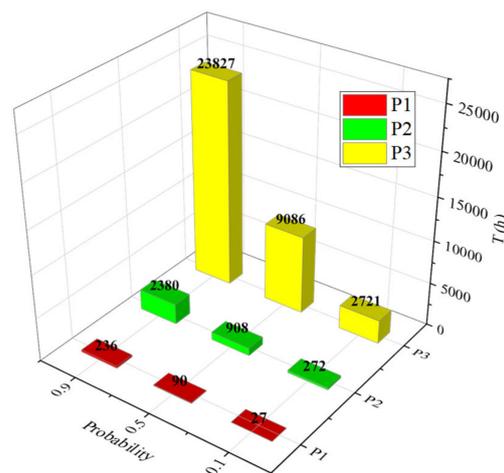


Figure 11. Working hours under different fatal failure probabilities.

P1, P2, and P3 are the corresponding system states under different failure rates. It can be seen that the dual-CPU hot redundancy system can greatly improve the reliability of the system compared with the system of one CPU. Even if the CPU failure rate is as high as  $10^{-2}/h$ , the AUV-T can still be guaranteed to be in a reliable working condition (R) for more than 27 h. The high reliability of the VxWorks system can ensure that the CPU has a very low failure rate. When the failure rate of the CPU is  $10^{-3}/h$ , the probability of

the AUV maintaining a normal working state ( $R_1$ ) for 72 h is 80.5%, and the probability of it maintaining a reliable working state ( $R$ ) is 99.1%. When the failure rate of the CPU is  $10^{-4}/\text{h}$ , the probability of the AUV maintaining a normal working state ( $R_1$ ) for 72 h continuously is 97.9%, and the probability of it maintaining a reliable working state ( $R$ ) is greater than 99.9%. In practical engineering applications, due to the limitations of the power supply carried by the AUV and the length of the tunnel, the continuous working time of the AUV for tunnel detection usually does not exceed 24 h. Therefore, the use of the dual-CPU hot redundancy system can ensure that AUV-T could complete the detection of the water conveyance tunnel in a reliable working state.

#### 4. Discussion

In order to verify the AUV-T's ability to detect tunnels and the reliability of the system, we used the AUV-T to conduct an automatic inspection of a water conveyance tunnel in Hangzhou, Zhejiang Province, China, in August 2019, as shown in Figure 12. Figure 12a is AUV-T. In Figure 12b, AUV-T was sailing in the tunnel. The length of the water conveyance tunnel is about 4 km. The sailing time of the AUV-T is 90 min, and the average speed is about 0.74 m/s.

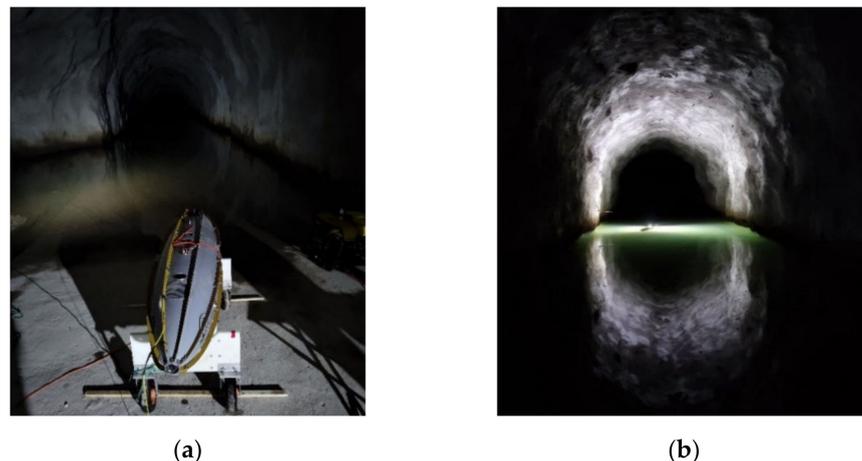


Figure 12. Experiment in Zhejiang, China.

The yaw, height data, and distance from the wall during the voyage are shown in Figures 13–15.

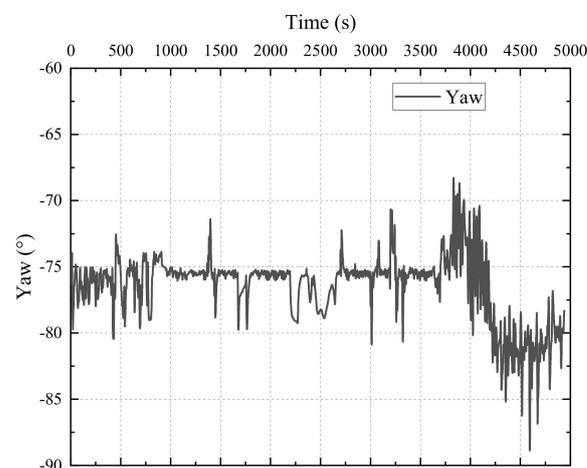


Figure 13. Yaw during voyage.

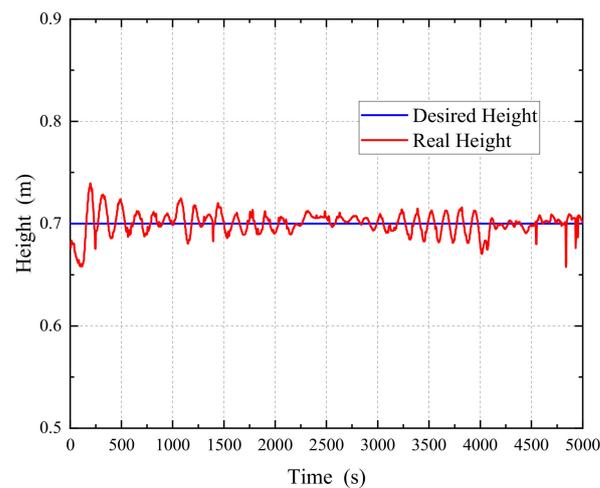


Figure 14. Height during voyage.

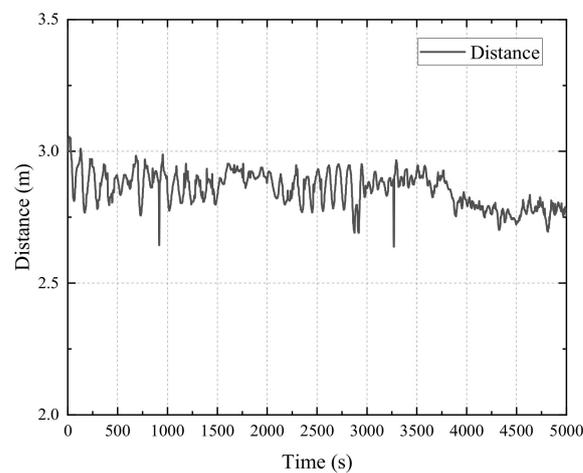


Figure 15. Distance from the wall during voyage.

It can be seen that the AUV-T can successfully detect the tunnel. During the selected sailing time, the AUV-T maintained a heading angle of about  $75^\circ$ , and the overshoot was less than  $5^\circ$ . At 4000 s, due to the change in the tunnel direction, the AUV-T realized the automatic adjustment of the heading angle after recognizing the tunnel direction change. On the one hand, the distance between the AUV and the bottom of the tunnel was always stable at 0.7 m, and the height fluctuation was less than 0.1 m. Using the controller proposed in this paper, the AUV-T could maintain a stable heading and height during the sailing in accordance with instructions to ensure the stability of the AUV-T during sailing, so as to realize the autonomous observation of the tunnel.

## 5. Conclusions

- (1) This paper designed a system structure suitable for a water conveyance tunnel inspection AUV. The task layer includes ten task modules: data fusion, course correction, relative positioning, wall defect detection, autonomous obstacle avoidance, visual guidance, fault diagnosis and self-rescue, wall following, height control, and energy and power. It can adapt to the harsh environment in a water conveyance tunnel and realize the detection of the tunnel.
- (2) The heterogeneous dual-line dual-CPU hot redundancy system was proposed and its reliability was analyzed under different failure rates and different working hours.
- (3) A stability analysis showed that the dual-CPU hot redundancy system proposed in this paper can significantly improve the reliability of the system, and the probability

of maintaining a reliable working state can reach more than 99%. In addition, the tunnel experiment showed that an AUV-T with the architecture designed in this paper can successfully and effectively complete an autonomous inspection of a water conveyance tunnel.

- (4) In the next step, the data collected this time will be compared with known tunnel data to evaluate the detection effect and provide a basis for water conveyance tunnel detection. In the not-too-distant future, the AUV-T will be used to conduct longer-distance autonomous inspections on more water tunnels.

**Author Contributions:** X.W.: Methodology, Writing—original draft, Writing—review & editing; Y.S.: Funding acquisition, Supervision; L.W.: Resources; H.B.: Resources; X.R.: Data curation. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research was funded by the Natural Science Foundation of Heilongjiang Province, grant number ZD2020E005, Financial support for Shaanxi Provincial Water Conservancy Science and technology program, grant number 2020slkj-5, and the China National Natural Science Foundation, grant number 51779057 and 51709061.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

- Wang, X.; Zhang, G.; Sun, Y.-S.; Wan, L.; Cao, J. Research on autonomous underwater vehicle wall following based on reinforcement learning and multi-sonar weighted round robin mode. *Int. J. Adv. Robot. Syst.* **2020**, *17*, 10. [[CrossRef](#)]
- Glushko, I.; Olenew, E.; Komar, M.; Kniese, L.; Sokolovskiy, R.; Kebkal, O.; Bannasch, R.; Kebkal, K. Software Control Architecture for the BOSS Manta Ray AUV Actuation System. In Proceedings of the IEEE/OES Autonomous Underwater Vehicle Workshop (AUV), Porto, Portugal, 6–9 November 2018. [[CrossRef](#)]
- Tipsuwan, Y.; Hoonsuwan, P. Design and implementation of an AUV for petroleum pipeline inspection. In Proceedings of the 2015 7th International Conference on Information Technology and Electrical Engineering (ICITEE), Chiang Mai, Thailand, 29–30 October 2015. [[CrossRef](#)]
- Pidić, A.; Aasbøe, E.; Almankaas, J.S.; Wulvik, A.S.; Steinert, M. Low-Cost Autonomous Underwater Vehicle (AUV) for Inspection of Water-Filled Tunnels During Operation. In Proceedings of the Volume 5B: 42nd Mechanisms and Robotics Conference, Quebec, QC, Canada, 26–29 August 2018. [[CrossRef](#)]
- Jia, C.; Chen, Z.; Zhang, G.-C.; Shi, L.; Sun, Y.; Pang, S. AUV Tunnel Tracking Method Based on Adaptive Model Predictive Control. *IOP Conf. Ser.* **2018**, *428*, 012070. [[CrossRef](#)]
- Freire, L.O.; Oliveira, L.M.; Vale, R.T.; Medeiros, M.; Diana, R.E.; Lopes, R.M.; Pellini, E.L.; De Barros, E.A. Development of an AUV control architecture based on systems engineering concepts. *Ocean Eng.* **2018**, *151*, 157–169. [[CrossRef](#)]
- Das, S.K.; Pal, D.; Nandy, S.; Kumar, V.; Shome, S.N.; Mahanti, B. Control Architecture for AUV-150: A Systems Approach. In *Trends in Intelligent Robotics, Proceedings of 13th FIRA Robot World Congress, FIRA 2010, Bangalore, India, 15–17 September 2010*; Springer: Heidelberg/Berlin, Germany, 2010; Volume 103, pp. 41–48.
- Wang, F.; Wan, L.; Jiang, D.-P.; Xu, Y.-R. Design and reliability analysis of DP-3 dynamic positioning control architecture. *China Ocean Eng.* **2011**, *25*, 709–720. [[CrossRef](#)]
- Wang, F.; Lv, M.; Xu, F. Design and implementation of a triple-redundant dynamic positioning control system for deepwater drilling rigs. *Appl. Ocean Res.* **2016**, *57*, 140–151. [[CrossRef](#)]
- Palomeras, N.; Carreras, M.; Ridao, P.; Hernandez, E. Mission control system for dam inspection with an AUV. In Proceedings of the 2006 IEEE/RSJ International Conference on Intelligent Robots and Systems, Beijing, China, 9–15 October 2006. [[CrossRef](#)]
- Singh, R.; Katal, A.; Kumar, M.; Singh, K.; Bhola, D. Removal of sonar wave interference in multi-robot system for the efficient SLAM by randomized triggering time technique. *World J. Eng.* **2020**, *17*, 535–542. [[CrossRef](#)]
- Liu, J.-H.; Li, R.-C. Analysis of thermal drift in high performance interferometric fiber-optic gyroscopes. *Chin. Opt.* **2020**, *13*, 333–343.
- Sun, Y.; Ran, X.; Zhang, G.; Wang, X.; Xu, H. AUV path following controlled by modified Deep Deterministic Policy Gradient. *Ocean Eng.* **2020**, *210*, 107360. [[CrossRef](#)]
- Sun, Y.-S.; Ran, X.; Zhang, G.; Xu, H.; Wang, X. AUV 3D Path Planning Based on the Improved Hierarchical Deep Q Network. *J. Mar. Sci. Eng.* **2020**, *8*, 145. [[CrossRef](#)]
- He, B.; Sha, Q.; Luo, J. Self-rescue system based on behavior decision-making and computed torque control for AUV. In Proceedings of the 2nd International Asia Conference on Informatics in Control, Automation and Robotics, Wuhan, China, 6–7 March 2010. [[CrossRef](#)]
- Yen, W.-K.; Guo, J. Wall following control of a robotic fish using dynamic pressure. Proceedings of OCEANS 2016—Shanghai, Shanghai, China, 10–13 April 2016; pp. 1–7.

17. Wang, X.; Zhang, G.; Sun, Y.-S.; Cao, J.; Wan, L.; Sheng, M.; Liu, Y. AUV near-wall-following control based on adaptive disturbance observer. *Ocean Eng.* **2019**, *190*, 106429. [[CrossRef](#)]
18. Xu, H.; Zhang, G.-C.; Cao, J.; Pang, S.; Sun, Y.-S. Underactuated AUV Nonlinear Finite-Time Tracking Control Based on Command Filter and Disturbance Observer. *Sensors* **2019**, *19*, 4987. [[CrossRef](#)] [[PubMed](#)]
19. Park, J.; Rhee, S.H.; Yoon, H.K.; Lee, S.; Seo, J. Effects of a propulsor on the maneuverability of an autonomous underwater vehicle in vertical planar motion mechanism tests. *Appl. Ocean Res.* **2020**, *103*, 102340. [[CrossRef](#)]
20. Xu, H.; Zhang, G.-C.; Sun, Y.-S.; Pang, S.; Ran, X.; Wang, X. Design and Experiment of a Plateau Data-Gathering AUV. *J. Mar. Sci. Eng.* **2019**, *7*, 376. [[CrossRef](#)]
21. Lei, Z.; Yongjie, P.; Ye, L.; Lei, W. Motion control of AUV based on embedded operation system. In Proceedings of the 2009 4th IEEE Conference on Industrial Electronics and Applications, Xi'an, China, 25–27 May 2009; pp. 1167–1172.