Qingzhu Zhang
Liang Zhang
Yang Shen
Yunhao Fu
Liangyou Li
Pengfei Zhang
Yulan Wei ✉

https://doi.org/10.21278/TOF.492069824
ISSN 1333-1124
eISSN 1849-1391

DEVELOPMENT AND TESTING OF A SEMI-PHYSICAL SIMULATION SYSTEM FOR A PIPE END FORMING MACHINE

Summary

To address the challenges of personalised customisation and rapid delivery in pipe fitting production—such as time constraints and complexities in programming and debugging pipe end forming machines—this study proposes a semi-physical simulation system. The system integrates a programmable logic controller (PLC) as the upper control unit with a virtual 3D model of the pipe end forming machine as the lower control unit, enabling data exchange through an intermediate serial port. The test results demonstrate that the system can simulate pipe machining with a response latency of approximately 14 ms and data transmission rates ranging from 88.75% to 95.63%. This approach provides a cost-effective, safe, and reliable platform for program testing and debugging in the pipe manufacturing industry while also offering valuable insights for simulating other complex mechanical systems.

Key words: *semi-physical simulation, pipe end forming machine, serial port, Unity 3D, PLC*

1. Introduction

Pipe fittings, often referred to as the "blood vessels" of industry, are widely used in the aerospace, automotive, marine, agricultural machinery and military industries, among others [1-3]. The hollow structure of metal pipe fittings provides advantages such as lightweight construction, high strength, strong toughness, low consumption, and excellent impact resistance and cushioning [4-6].

To achieve connections between metal pipe fittings, it is necessary to process the pipe ends using a pipe end forming machine. This process includes operations such as upsetting [7,8], spinning [9], and chamfering [10] (Fig. 1). Upsetting is used to expand or reduce the length of the pipe ends, facilitating the interconnection of pipes. Spinning is employed to extrude grooves, making it easier to insert sealing rings. Chamfering is utilised to remove burrs, ensuring that pipes can be connected without damaging the material at the pipe ends.



Fig. 1 Pipe end shapes after processing

With the increasing specialisation of the market and advances in equipment manufacturing, customer demand regarding the shapes of pipe ends is showing the need for increasing diversification, personalisation, and small batch sizes [11,12]. For example, some customers may require only a specific function, such as upsetting, spinning, or chamfering, either individually or in combination, or they may have specific requirements for pipe material parameters [13]. To stand out in a competitive market, manufacturers of pipe end forming machines need to be able to swiftly customise equipment and promptly deliver it to customers.

Currently, the development of pipe end forming machines needs to proceed gradually through a process that includes mechanical modelling, electrical development, equipment assembly, and debugging of control programs [14]. The debugging of control programs is often completed through onsite testing only after machine assembly is finished. This incurs exceptionally high costs in terms of time, manpower, materials, and site resources [15]. Furthermore, the processes of program simulation and debugging focus solely on identifying logical errors after program development, and they may not accurately capture actual operating conditions [16]. In the case of a personalised, custom-designed pipe end forming machine, unforeseen circumstances may arise during the debugging process, potentially leading to equipment damage or even causing personnel injuries [17,18].

In semi-physical simulation technology, certain portions of simulation models are connected to parts of actual objects through communication devices to complete system simulation testing. Such techniques are commonly employed for simulation and debugging during secondary development [19-21]. Jakub Wnorowski [22] developed an anchor positioning system to address the ship positioning problem. This system, with a programmable logic controller (PLC) as the upper computer and a virtual simulation model as the lower computer, utilises an RS-232 serial port connection for data transmission, markedly improving the accuracy of ship positioning. On the basis of the bottle sealing process, Adrián Villarroel [23] developed a virtualised bottle sealing factory platform. This system uses a PLC as the upper computer, and configuration software, along with Unity 3D as the lower computer, and bidirectional communication is enabled through a TCP/IP Ethernet connection to constitute a virtual training platform.

Compared with traditional physical and fully virtual simulation methods, semi-physical simulation systems offer significant advantages by combining both virtual and physical components, leading to a more reliable testing environment for complex industrial processes. Although traditional physical simulations achieve 100% accuracy and highly dependable real-world feedback, they are constrained by high costs, lengthy setup times, and slower response rates, making them impractical for situations that require flexibility or real-time adjustments [24]. On the other hand, fully virtual simulations are more cost effective, with rapid response times and short setup durations [25]. However, these methods often fall short in accurately representing complex mechanical systems because of the absence of real-world feedback.

Semi-physical simulation technology addresses the limitations of traditional software simulations in accurately replicating the actual operating conditions of complex systems. This approach reduces the risk that errors may be identified only during the final stage of onsite testing and helps to circumvent the extensive consumption of manpower, materials, finances, and other resources associated with conducting numerous real experiments. The incorporation of both virtual and physical components significantly shortens the product development cycle [26-28]. However, like all simulation methods, it has limitations, particularly in real-time performance compared to fully hardware-based systems or high-fidelity virtual environments. Consequently, establishing a semi-physical simulation system for conducting experimental tests on the control systems of custom-designed pipe end forming machines prior to prototype testing is both economical and highly essential.

A semi-physical simulation system requires data exchange between the upper computer and the lower computer. However, for pipe end forming machines, there is no universal third-party software available for PLC communication. To address this issue, one of two solutions can be adopted: Ethernet communication or serial port communication [29,30]. However, pipe end forming machines often use small to medium-sized PLCs. For earlier series such as the Mitsubishi FX2N and FX3U series, the Ethernet communication modules can be expanded, but at additional cost. Consequently, in practical production, most assembly sites cannot achieve full Ethernet coverage. Furthermore, PLCs that use Ethernet communication are typically based on a bus-type architecture for local area networks (LANs) [31]. This communication method involves competitive transmission, collision detection, and carrier sensing mechanisms, leading to unstable transmission rates. In particular, when there is an excessive number of connected network devices within the LAN, the number of communication data packets sent may increase accordingly, potentially resulting in packet loss [32,33]. Therefore, in this paper, the data transmission issue is addressed via the serial port communication method, which is more commonly employed in industrial equipment. The primary advantage of serial port communication lies in its widespread use, and it remains a standard feature of current desktop PCs [34]. Most industrial equipment is also equipped with serial ports, mainly for convenience in connecting printers. In cases where devices lack serial ports, a common approach during development is to connect through a serial port of a computer [35]. Thus, serial port communication is considered the simplest and most straightforward method for enabling devices to communicate.

In summary, this paper focuses on the semi-physical simulation of a pipe end forming machine, which includes an upsetting module, a spinning module, and a chamfering module. A semi-physical simulation system has been developed in which a PLC is used as the upper computer and a virtual 3D model of the pipe end forming machine is used as the lower computer. Real-time interaction between the upper and lower computers is achieved through the development of communication interfaces. This allows the real-time data output from the upper PLC to control the actions of the virtual 3D model of the pipe end forming machine. The results of the actions of the virtual 3D model are then fed back to the upper PLC. The semi-physical simulation system of the pipe end forming machine developed in this paper can simulate pipe processing, which is convenient for engineers to find abnormal working conditions in time, modify and debug them in time, increase the safety factor of developers in the process of equipment debugging, and solve the problems of insufficient program test time and fault troubleshooting time in the development process of the pipe end forming machine.

2. Design of the semi-physical simulation system

The overall scheme of the developed semi-physical simulation system for a pipe end forming machine is illustrated in Fig. 2. This system comprises an upper computer (a PLC), an intermediate communication interface, and a lower computer (a virtual 3D model of the pipe end forming machine).

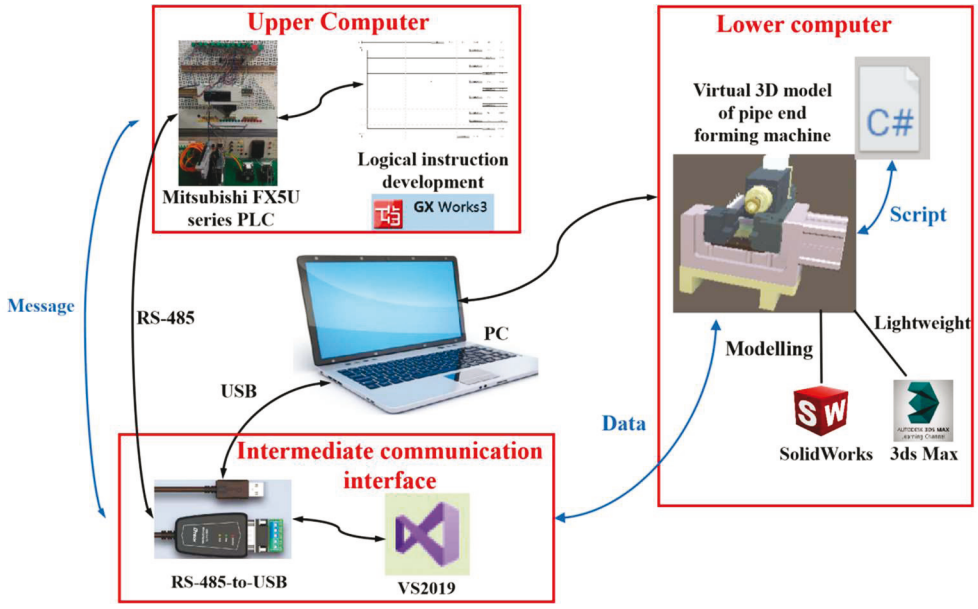


Fig. 2 Overall system structure

The upper computer is a Mitsubishi FX5U series PLC and its supporting equipment. It is the control core of the semi-physical simulation system. Instruction development, data exchange compilation and debugging are carried out using GX Works3, which is the development software for the Mitsubishi PLC series and is primarily utilised for logical instruction development for the upper computer in the semi-physical simulation system.

The lower computer is the virtual 3D model of the pipe end forming machine established by a PC. SolidWorks and 3ds Max are used separately for the construction and optimisation of the virtual 3D model. The lower computer is responsible for receiving the data sent from the upper computer PLC and then analysing and selecting the data and driving the lower computer through the C# driver control script.

The intermediate communication interface is used to communicate between the upper computer PLC and the lower computer virtual 3D model. The RS-485-to-USB communication serial port is used to connect the PLC to the PC. The communication format of the serial port was developed in Visual Studio 2019, including the communication protocol, format and basic parameters of the two sides, to achieve data exchange between the upper computer and the lower computer.

The data transmission process in the semi-physical simulation system is described below.

Data in the correct instruction format sent by the upper computer are concatenated to form a message. The intermediate communication interface parses the message, removes the frame header and footer, and extracts the transmitted data. The data are then analysed, and if the format and content are correct, the corresponding script is used to control the virtual 3D model of the pipe end forming machine to reach the specified position. At this time, the virtual 3D model of the pipe end forming machine sends data back to the upper computer. The sent data are validated through the intermediate communication interface, which creates a feedback message that is sent to the upper computer. The data are then processed and stored in the designated software.

3. Construction of the semi-physical simulation system

The construction of the semi-physical simulation system includes the development and compilation of the upper computer command, the design of the intermediate virtual communication interface, the modelling of the virtual 3D model of the lower computer, the environment construction of the virtual simulation system, the texture mapping and the development of the motion process and script.

3.1 Upper computer command development and compilation

The control system of the upper computer of the semi-physical simulation system of the pipe end forming machine employs the PLC as its core, which is responsible for the issuance and feedback of instructions. Consequently, this section first analyses the processing flow of the pipe end forming machine and then develops the upper computer logic instructions according to the processing flow.

3.1.1 Upper computer instruction development

In this work, the main development focus is on the upsetting module, spinning module, and chamfering module of the pipe end forming machine, as depicted in Fig. 3. The corresponding logical instructions are developed in accordance with their respective processes.



1-Upsetting module; 2-Spinning module; 3-Foundation; 4-Chamfering module; 5-Feeding device

Fig. 3 Pipe end forming machine

(1) Upsetting module

After the feeding device delivers the pipe material to the upsetting processing station, a cylinder controls a die to clamp the pipe. Once the pressure plate cylinder is secured, the process of punching the pipe material begins. Head dies of different shapes and sizes are used to perform one, two, three, four, or a maximum of five continuous direct upset extrusion processes on the pipe material, forming various required pipe end shapes. After the punching process is complete, the pressure plate and cylinder are released. Fig. 4 shows the processing operation of the upsetting module.

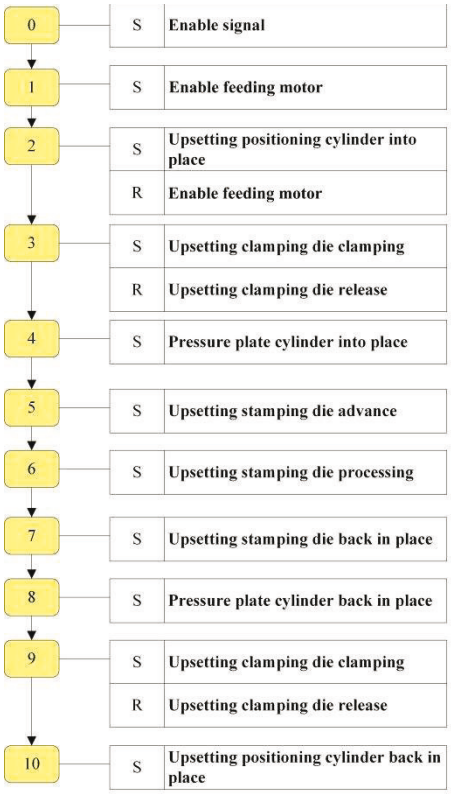


Fig. 4 Processing operation of the upsetting module

(2) Spinning module

After the feeding device delivers the pipe material from the upsetting processing station to the spinning processing station, the spinning positioning cylinder positions the pipe material. Once positioning is complete, the cylinder controls the die to clamp the pipe, and the rotary compression tool processes the pipe material to create grooves at the pipe end. After spinning is complete, the cylinder is released. Fig. 5 shows the processing operation of the spinning module.

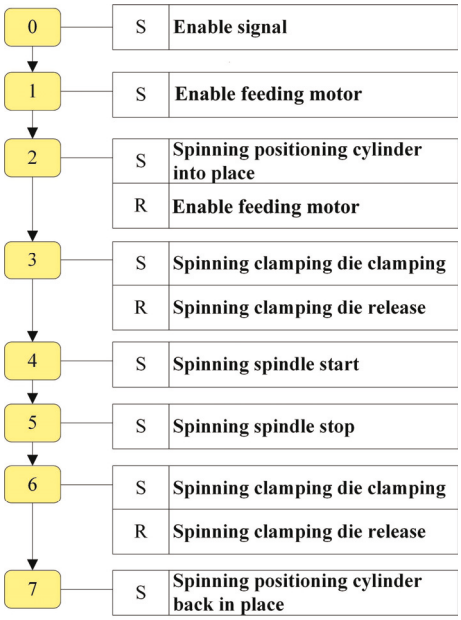


Fig. 5 Processing operation of the spinning module

(3) Chamfering module

After the feeding device delivers the pipe material from the spinning processing station to the chamfering processing station, the chamfering positioning cylinder positions the pipe material. Once positioning is complete, the die clamps the pipe, and the high-speed rotating chamfering tool disc chamfers and removes burrs from the pipe material. After chamfering is complete, the cylinder is released. Fig. 6 shows the processing operation of the chamfering module.

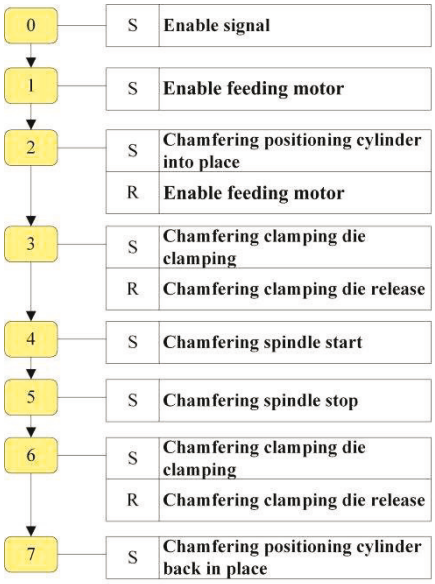


Fig. 6 Processing operation of the chamfering module

3.1.2 Upper computer instruction compilation

Because the software used for the virtual 3D model of the pipe end forming machine is third-party software and there is no mature protocol for communication between the PLC and this software, it is necessary to develop and compile the data transmission and reception of the upper computer program on the basis of the developed logic instructions.

A custom communication protocol is adopted in this work for Mitsubishi serial communication. Therefore, it is necessary to define the formats of the transmitted and received messages. We choose to send values consisting of 8 bytes in four registers. The original program is compiled using RS-2 instructions, and the output formats of the compiled messages are shown in Table 1.

The data executed in accordance with the RS-2 instructions are transmitted in a round-robin fashion, and the data to be output are placed in registers D1000, D1001, D1002, and D1003. In steps in which there is no position movement or speed output, 0000 is used as the output for registers D1002 and D1003. When the instruction reaches the current step, the key data are stored in the registers to be sent.

Table 1 Message output formats

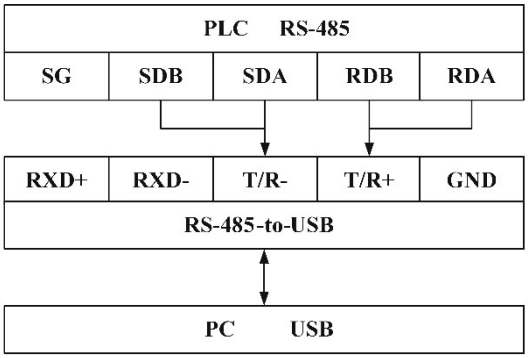
Bytes	Data definition	Data type	Scope	Remarks
1-2 (Instruction output register D1000）	Output position	Signed	0-20	Corresponding to all outputs
3-4 (Instruction output register D1001)	Instruction 1	Signed	2-byte signed integer	Displacement
5-6 (Instruction output register D1002)	Instruction 2	Signed	2-byte signed integer	Speed
7-8 (Instruction output register D1003)	Instruction 3	Signed	0 and 1	State change

The upper computer utilised in the semi-physical simulation system incorporates an FX5U-64MR/ES PLC as the control core in this work. The input signals are digitised to simulate sensor signals, and the output signals consist of action commands to be executed by various modules, which are processed through Mitsubishi's nonsequential protocol. These output action commands are used to control the motion of the virtual model of the pipe end forming machine.

3.2 Intermediate virtual communication interface design

Once the control instructions are in a form that can be used to control the virtual 3D model, a communication interface is needed both to pass those instructions to the virtual model and to return feedback data from the virtual 3D model to the PLC.

The intermediate virtual communication interface between the PLC and the virtual 3D model of the pipe end forming machine uses serial communication. Given that the virtual 3D model lacks available physical ports, an RS-485-to-USB converter is employed to connect the PLC to the PC on which the virtual model is run, as shown in Fig. 7.



SG: signal grounding; SDB: send data (B-); SDA: send data (A+); RDB: receive data (B-); RDA: receive data (A+); RXD+: receive data (+); RXD-: receive data (-); T/R-: send data (-); T/R+: send data (+); GND: ground

Fig. 7 Wiring diagram

To avoid system stalling caused by thread blocking and reduce the response time and errors in execution, multithread programming is used in this work to develop a virtual serial port. The steps are as follows:

(1) Basic parameter configuration and development of the virtual serial port

The basic parameters include the port number, baud rate, stop bit, data bit, check bit, sending frame head and sending frame tail. The configuration of these parameters needs to match that of the basic parameters of the physical port to ensure the fluency of data transmission and establish realistic communication conditions.

(2) Creation of a read thread

The functions of the read thread include opening the serial port, reading data from the serial port buffer, exception handling, thread dormancy and thread destruction. Its purpose is to receive data sent from the upper computer.

(3) Creation of a data analysis thread

The functions of the data analysis thread are to analyse the data received by the read thread, to analyse and determine the frame head and frame tail of the received data, to determine the correct format, and then to output the key data to a script for the virtual 3D model of the pipe end forming machine.

(4) Creation of a data output thread

The functions of the data output thread include writing data to the buffer, exception handling, thread dormancy and thread destruction. Its purpose is to feed back the execution results to the upper computer.

3.3 Design of the virtual 3D model simulation system for the lower computer

Once the control instructions have been passed to the virtual model via the communication interface, the virtual model needs to be used to simulate the operation of the pipe end forming machine in accordance with those control instructions.

The overall design of the simulation system for the virtual 3D model of the pipe end forming machine is shown in Fig. 8. The system is divided into three parts: a view layer, a logic control layer, and a management layer. These three layers perform their duties in parallel.

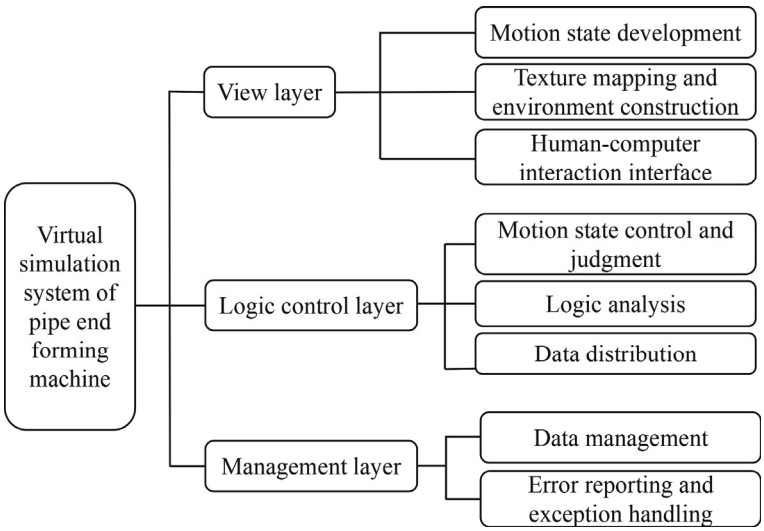


Fig. 8 Virtual 3D model simulation system

(1) The view layer is responsible for the interface through which the virtual 3D model of the pipe end forming machine is presented to the user. Its main functions include the construction of the environment of the virtual 3D model, the development of the motion process, texture mapping, light source simulation, and the design of the human–computer interaction interface.

(2) The logic control layer is responsible for the logical analysis and matching of the data transmitted from the PLC to control the movement of the virtual 3D model of the pipe end forming machine.

(3) The management layer manages the data of the entire virtual 3D model, performing functions such as abnormal state capture and maintenance of connection timeout records.

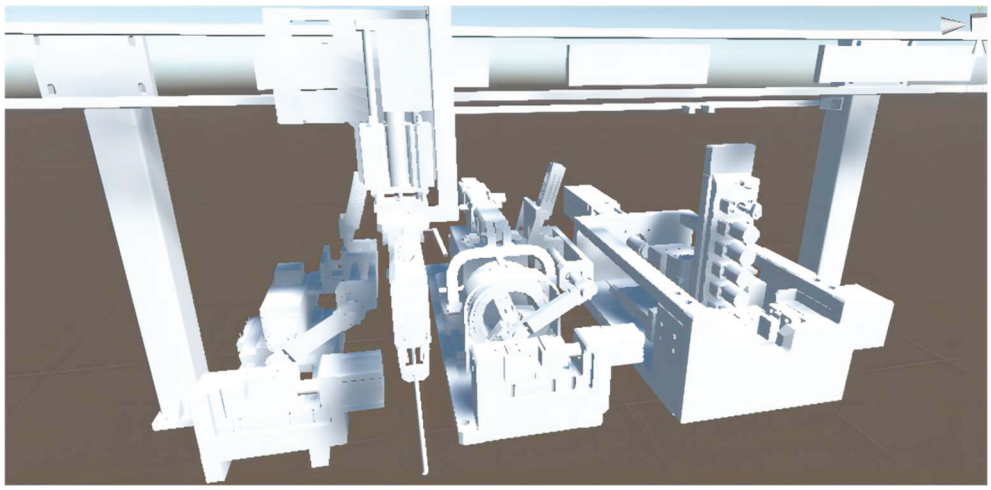
During the operation of the system, the operation status of each component of the pipe end forming machine can be comprehensively observed, and the logical relationship between the program under test and the actions of the machine can be tested.

3.3.1 Modelling process of the virtual 3D model

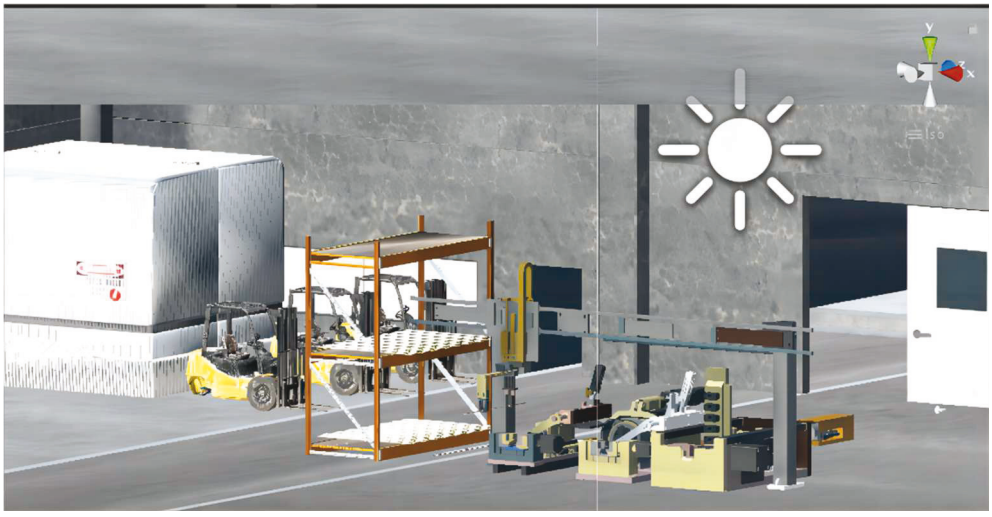
After the virtual 3D model of the pipe end forming machine is formed in SolidWorks to ensure system fluency during the software operation and to prevent the model from being too large, which could result in system jamming or motion control lag, the model is imported into 3ds Max for lightweight processing. A comparison of the number of surfaces and the memory usage before and after lightweight processing is shown in Table 2. It can be seen from this table that after lightweight processing, the number of surfaces and the amount of memory occupied are reduced by 86% and 99.8%, respectively; thus, this processing greatly improves the running speed of the virtual 3D model. The lightweight model is represented as a file in FBX format and then imported into Unity 3D, as shown in Fig. 9 (a). To increase the authenticity of the virtual 3D model, it is mapped to and integrated with models of other features of the factory environment, such as shelves and forklifts. The complete system is shown in Fig. 9 (b), from which it can be seen that the effect is relatively realistic.

Table 2 Comparison before and after lightweight processing

Lightweight processing of model	Before treatment	After treatment	Reduction ratio
Number of surfaces	1372.1K	192.1K	86%
Memory footprint	548.3 MB	1.5 MB	99.8%



(a) Model diagram before texture mapping



(b) Diagram of the complete environment after mapping

Fig. 9 Simulation system for the virtual 3D model of the pipe end forming machine

3.3.2 Motion process and script development

To make the virtual 3D model of the pipe end forming machine move according to the instructions of the upper computer, it is necessary to develop the motion process and script. In this work, the motion state of the virtual 3D model of the pipe end forming machine is simulated by an Animator state machine in Unity 3D. The Animator's interface development function allows more abstract logical relationships to be repaired and combined so that the process of logical control is more fluent and clearer, which is conducive to the integration of code and motion processes and can enhance the realism of the system.

Through the development of multiple key frames on the timeline of a model, Unity3D forms a continuous and smooth motion process according to the motion position of the object on the key frame. The development, management, and switching of the motion process are managed through the Animator Controller panel. The development process of the system

involves the movement of multiple parts, so there will be different motion processes of different parts. If all the motion actions are developed using the parent objects for the key frame, this increases the complexity and difficulty of system development. To solve this problem, this paper develops the motion process of the subobject on the basis of sorting out the parent–child relationship. In addition, because the system also requires a fast response time and precise control, it is necessary to set the parameters in the transition layer, the most important of which are the exit time, transition duration, and loop time.

The exit time is the ratio of the time transition between the previous motion process and the next one, which is affected by the transition duration. The transition duration involves the synchronous execution of two motion processes. Loop time is present in the initial definition of each motion process. After the script is executed by Animator, the virtual 3D model is controlled by default to move continuously. With this function, the motion will be stopped at the end of the last frame of the motion until the motion state of the virtual 3D model is switched through Animator. To improve the response time, this paper sets the exit time to 1 and cancels the transition duration and loop time. The motion process development and parameter configuration are shown in Fig. 10.

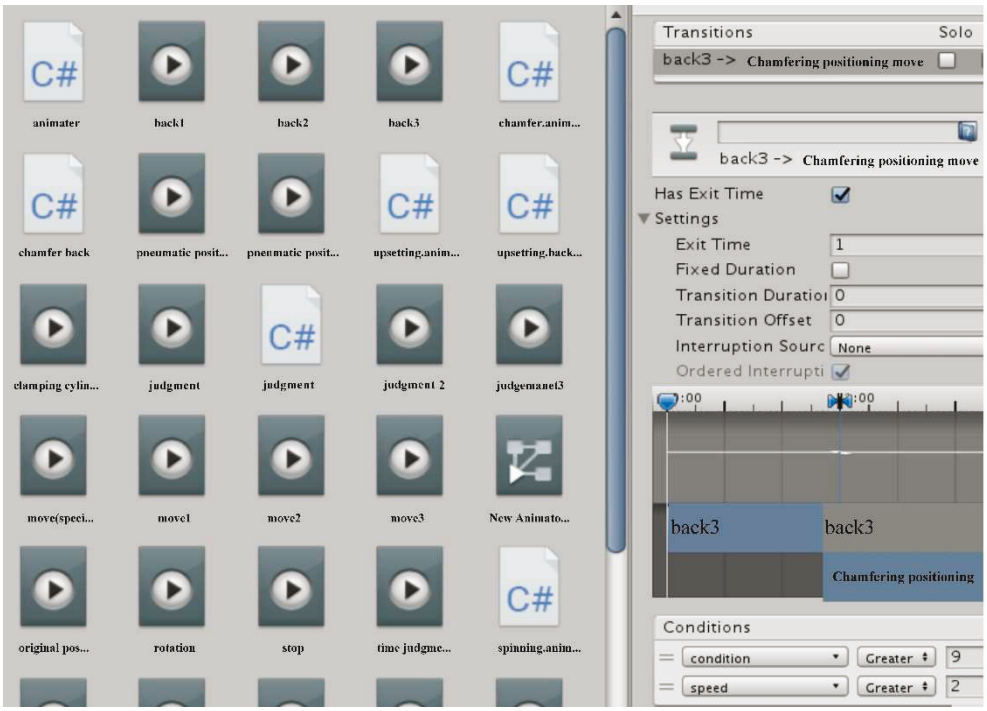
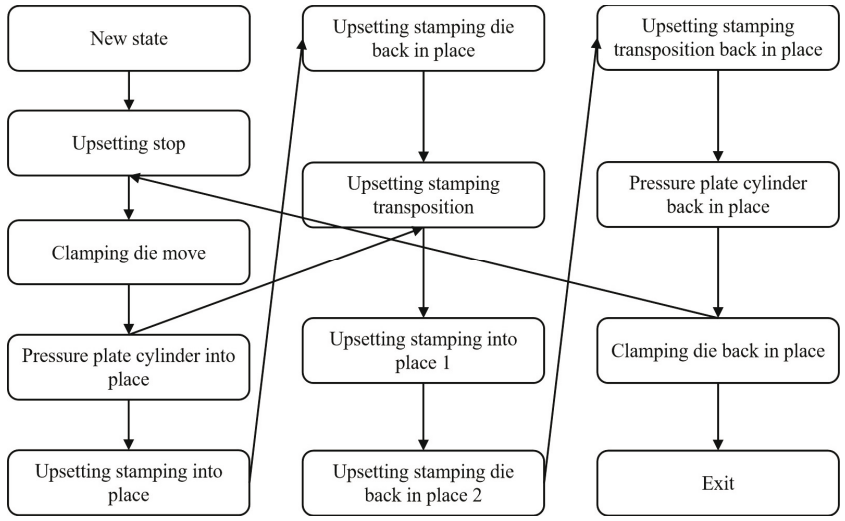


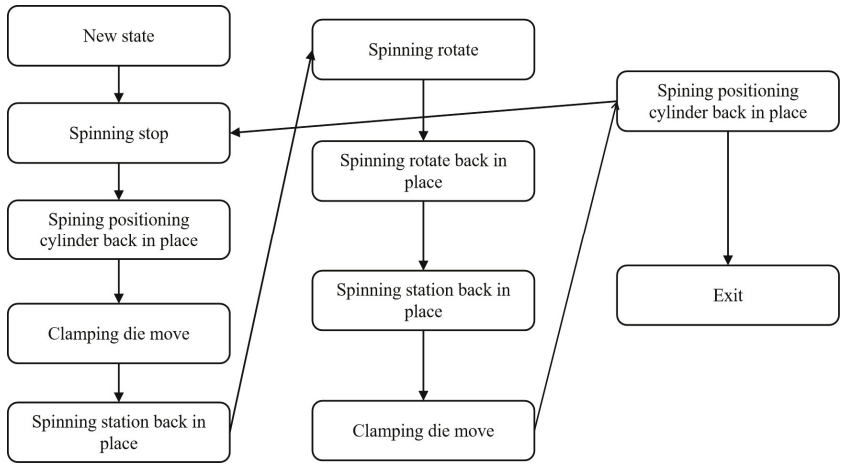
Fig. 10 Motion process development and parameter configuration

After completing the development of the motion process and parameter configuration, the logic analysis of the state transformation is carried out. The motion process is developed according to the motion process of the pipe end forming machine controlled by the upper computer. Diagrams illustrating the motion process control development of the upsetting, spinning, and chamfering modules are presented in Fig. 11 (a), Fig. 11 (b) and Fig. 11 (c), respectively.

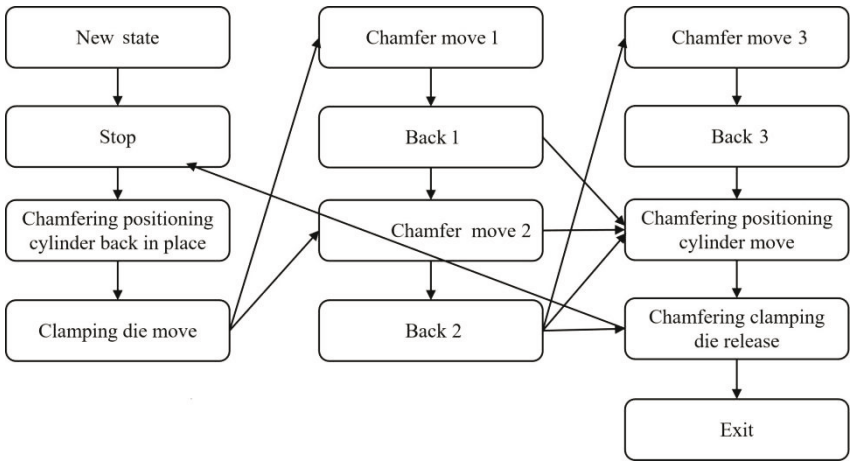
In Fig. 11, each box represents every motion process of the virtual 3D model of the pipe end forming machine. The indicator arrows depict the logical relationship between the current motion state and the subsequent motion state. This logical relationship corresponds directly to the one programmed in the upper computer. The conditions for transitioning between motion states are determined by scripts and calculations utilising data transmitted from the upper computer to the virtual 3D model of the pipe end forming machine.



(a) Upsetting module



(b) Spinning module



(c) Chamfering module

Fig. 11 Motion process control development diagrams of each module

A logical analysis of the RS-2 instructions of the Mitsubishi series PLC reveals that for each instruction sent by the PLC, the 8-byte data from the four continuous registers, which form a complete set of data, are sent to Unity 3D through the virtual communication serial port. After a set of data is received, the received data are processed and analysed in accordance with the custom communication format, and the data are parsed through the developed script. After the analysis is complete, the virtual model of the pipe forming machine is controlled as specified by the data.

In the semi-physical simulation system, the upper computer program corresponds closely with the script development in the Unity 3D system. To ensure clarity in the script development process, it is essential to analyse and set each message and feedback data transmitted from the upper computer to Unity 3D. The returned data rely on the current number of instruction steps executed by the upper computer. Once the movement process is complete, the data are sent to the upper computer's register D2003. Furthermore, by comparing the returned data, constraints can be placed on the messages sent by the upper computer, ensuring that only messages pertinent to the current step are transmitted. This approach guarantees a clearer operational process for the entire system, facilitating easier debugging and troubleshooting during the development phase.

The determination of actions for the upsetting, spinning, and chamfering modules in the pipe end forming machine is contingent upon the object requiring control. Precise invocation is achieved through a well-organised hierarchical structure. The script diagram of the logical analysis and determination is shown in Fig. 12. The logic analysis and determination method presented in this paper is used in the update process. Data can be obtained promptly through system frame calls, thereby enabling efficient and timely data analysis.

```
void DecisionmethoddD1(){...}   // Upsetting clamping die
void DecisionmethoddD2(){...}   // Upsetting stamping die
void DecisionmethoddD3(){...}   // Spinning clamping die
void DecisionmethoddD4(){...}   // Chamfering clamping die
void DecisionmethoddD5(){...}   // Pressure plate cylinder
void DecisionmethodD6(){...}    // Spinning positioning cylinder
void DecisionmethodD7(){...}    // Chamfering positioning cylinder
void DecisionmethodD8(){...}    // Upsetting station moving
void DecisionmethodD9(){...}    // Spinning  station moving
void DecisionmethodD10(){...}   // Spinning station rotation
void DecisionmethodD11(){...}   // Chamfering station moving
```

Fig. 12 Script diagram of logical analysis and determination

3.4 System parameter settings

To ensure the normal operation and real-time data transmission of the semi-physical simulation system, it is necessary to add a pair of virtual serial ports to both the device manager and Unity 3D and ensure consistency in the serial port parameters. In this work, a virtual serial port assistant is used to add a pair of virtual serial ports to the PC. COM1 is used as the virtual serial port for Unity 3D. For the device manager port, COM2 is added through the split function and matched with COM1, and the serial port parameters are set to be consistent.

The input signal is sent through the I/O port, and the output signal is used to control the movement of the virtual 3D model of the pipe end forming machine. When the virtual 3D model reaches the specified position, the corresponding data are sent back, and the returned data are placed in the specified register. The returned data are compared with the data of the specified register to verify whether they are correct; if so, then the next data are sent. In accordance with the above method, testing of the whole system can be completed.

The operation interface of the developed semi-physical simulation system for a pipe end forming machine is shown in Fig. 13. The upper part is the visual window, and the lower part is the camera visual window. The control interface of the system is shown on the left-hand side of the camera visual window and displays the basic parameters, the input and output data and the serial port communication status.



Fig. 13 System operation interface

To achieve personalised customisation and meet the different needs of customers for pipe parameters, it is necessary to change the processing program and working condition parameters of the upper computer. This paper tests the operation of the semi-physical simulation system under two working conditions, the parameters of which are shown in Table 3.

Table 3 Test conditions

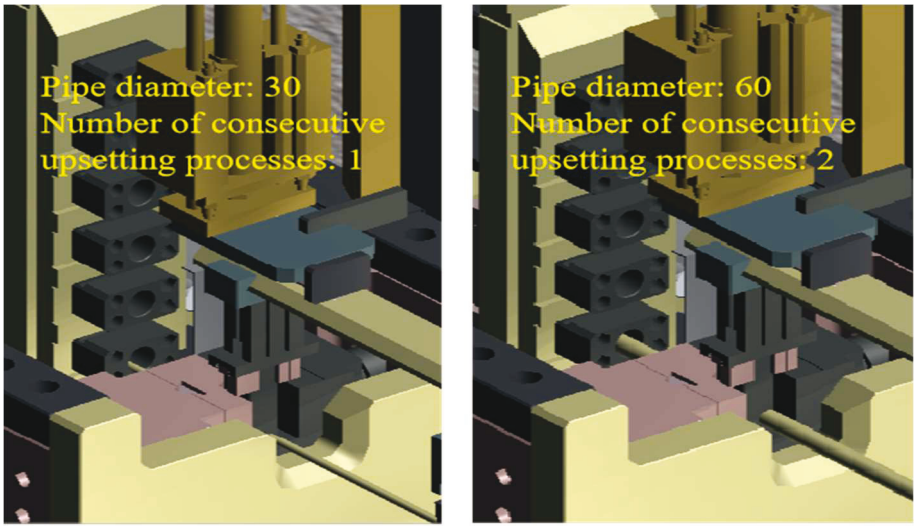
Working condition	Number of pipe fittings	Pipe diameter/mm	Pipe length /mm	Spinning stroke /mm	Chamfer angle /°	Chamfered edge length /mm	Number of consecutive upsetting processes
1	10	ϕ30	300	2	45	1	1
2	10	ϕ60	300	6	45	2	2

4. Construction of the semi-physical simulation system

Taking the running time, real-time performance and stability of the system as indexes, the semi-physical simulation system of the pipe end forming machine was tested, and the test results were analysed.

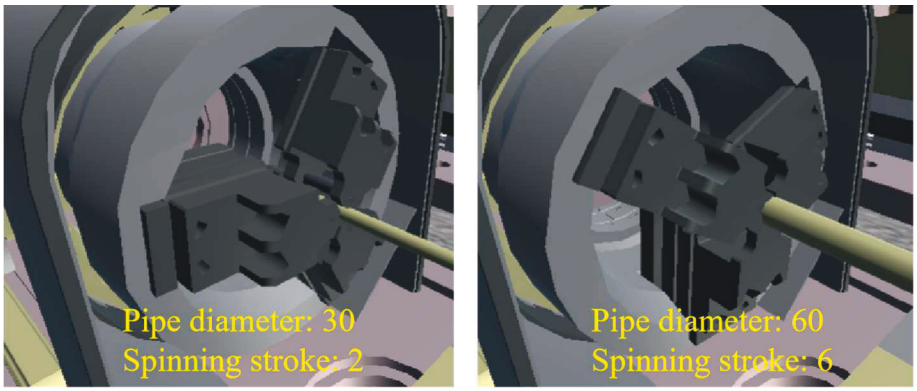
4.1 System running time

Fig. 14, Fig. 15 and Fig. 16 show images of the virtual 3D model during the machining processes of the upsetting module, the spinning module, and the chamfering module under the two working conditions. These images illustrate that the developed semi-physical simulation system can simulate the operation of the pipe end forming machine under different working conditions.



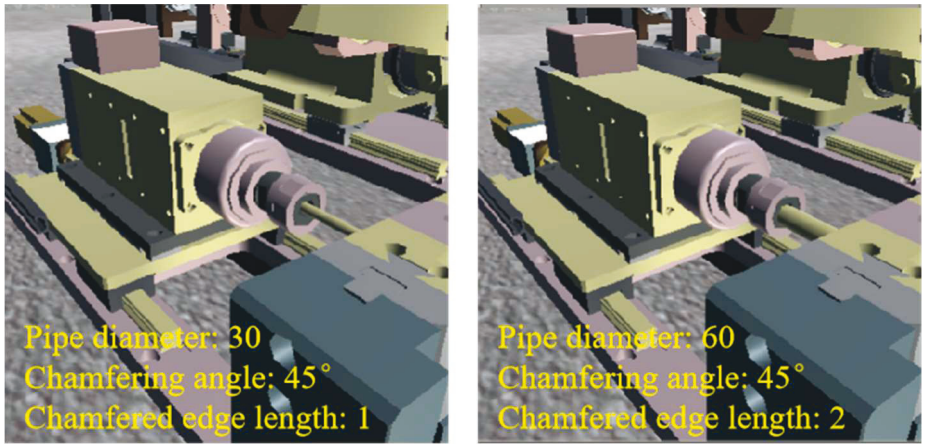
(a) Condition 1 (b) Condition 2

Fig. 14 State of the virtual model during processing with the upsetting module



(a) Condition 1 (b) Condition 2

Fig. 15 State of the virtual model during processing with the spinning module



(a) Condition 1 (b) Condition 2

Fig. 16 State of the virtual model during processing with the chamfering module

The processing times for the three modules under the two working conditions are presented in Table 4. The data show that an increase in the pipe diameter increases the complexity of machining, particularly in the upsetting module, which is highly sensitive to such changes. The processing time increases from 15 s to 45 s, a 200% rise, due to the greater resistance of larger pipes. This requires more frequent and precise adjustments in the simulation

to accurately model the increased forces and deformation. The processing time of the spinning module is notably affected by the stroke length, increasing from 33 s to 46 s, a 39.4% increase, reflecting the added complexity of simulating larger displacements and stress factors in the material. Although the chamfering module is less sensitive to parameter changes, it still shows a slight increase in processing time, from 25 s to 26 s, which impacts the overall system performance. Furthermore, multiple consecutive upsetting processes significantly increase the system load, causing the processing time to increase exponentially, particularly with the increasing pipe diameter.

In summary, the pipe diameter, spinning stroke length, chamfer length, and the number of upsetting processes significantly affect system performance, with larger diameters and longer strokes leading to increased processing time and system load. This indicates that specific process parameters directly influence the system's performance, leading to longer processing times under more demanding conditions.

Table 4 Processing time

Condition	Time/s		
	Upsetting module	Spinning module	Chamfering module
1	15	33	25
2	45	46	26

4.2 System real-time and stability tests

In these experiments, the same data used in the running time tests were selected as the instructions to send. The response delay time of the system and the proportion of effective data transmission were tested in a round-robin manner. Separate tests were run with the virtual serial port assistant connected to the virtual 3D model of the pipe end forming machine, with the PLC connected to that virtual 3D model, and with the PLC connected to the virtual serial port assistant. The receiving and sending processes of the serial port assistant are similar to the operation mode of the nonsequential communication protocol. A message is sent first, the data are sent back after reception is completed, and the returned data are sent the next time after being received. The response delay time of the system is calculated by comparing the measured number of transmissions (n_t), the measured number of receptions (n_r), and the theoretical number of transmissions (N).

The calculation formula for the theoretical number of transmissions is as follows:

$$N = \frac{t \times 1000}{t_1} \tag{1}$$

where N is the theoretical number of transmissions; t is the testing time, in s; and t_1 is the transceiving time, in ms. The calculation formula for the response delay time is as follows:

$$t_2 = \frac{t \times 1000}{n_t} - t_1 \tag{2}$$

where t_2 is the response delay time, in ms; and n_t is the measured number of transmissions.

The calculation formula for the proportion of effective data transmission is as follows:

$$R = \frac{n_t}{N} \times 100\% \tag{3}$$

where R is the proportion of effective data transmission, expressed as a percentage.

The test time t was 360 s, the baud rate was 9600, and three transceiving times t_1 of 250 ms, 500 ms and 750 ms were tested, each repeated ten times with the average values taken as the results. The resulting test data for the 250 ms, 500 ms and 750 ms system transmit and receive frames are shown in Table 5, Table 6 and Table 7, respectively.

Since the number of transmissions during the test is the number sent by the upper computer and the number of receptions is the number sent by the lower computer, the measured number of transmissions of the PLC and the virtual serial assistant, the virtual serial assistant and the virtual 3D model of the pipe end forming machine, and the PLC and the virtual 3D model of the pipe end forming machine in Table 5, Table 6 and Table 7 is one more than the number of transmissions received.

Table 5 Test data for t_1=250 ms transmit and receive frames

Data	PLC and virtual serial assistant	Virtual serial assistant and virtual 3D model of pipe end forming machine	PLC and virtual 3D model of pipe end forming machine
n_r	1278	1437	1277
n_t	1279	1438	1278
N	1440	1440	1440

Table 6 Test data for t_1 =500 ms transmit and receive frames

Data	PLC and virtual serial assistant	Virtual serial assistant and virtual 3D model of pipe end forming machine	PLC and virtual 3D model of pipe end forming machine
n_r	676	718	676
n_t	677	719	677
N	720	720	720

Table 7 Test data for t_1 =750 ms transmit and receive frames

Data	PLC and virtual serial assistant	Virtual serial assistant and virtual 3D model of pipe end forming machine	PLC and virtual 3D model of pipe end forming machine
n_r	459	479	458
n_t	460	480	459
N	480	480	480

Fig. 17 shows the response delay time between the PLC and the virtual serial assistant, the virtual serial assistant and the virtual 3D model of the pipe end forming machine, and the PLC and the virtual 3D model of the pipe end forming machine when the transceiving times are 250 ms, 500 ms and 750 ms. The results show that for the different transceiving times, the response delay times between the PLC and the virtual serial assistant range from 31.47 ms to 32.61 ms, whereas those between the virtual serial assistant and the virtual 3D model of the pipe end forming machine range from 0 ms to 0.70 ms. Accordingly, the response delay times between the PLC and the virtual 3D model are similar to those between the PLC and the virtual serial assistant, ranging from 31.69 ms to 34.31 ms. This delay is attributed primarily to the scanning time of the PLC and the initial setting delay timer for the physical serial port. In this system, the delay timer for physical serial port initialisation is set at 20 ms. By minimising this delay timer, the overall response time delay of the system can be reduced to approximately 14 ms. The PLC response time, as stated in the Mitsubishi series user manual, ranges from 0.5 ms to 100 ms. Consequently, a response delay of 14 ms is within acceptable operational parameters, indicating that the semi-physical simulation system developed in this study exhibits good real-time performance.

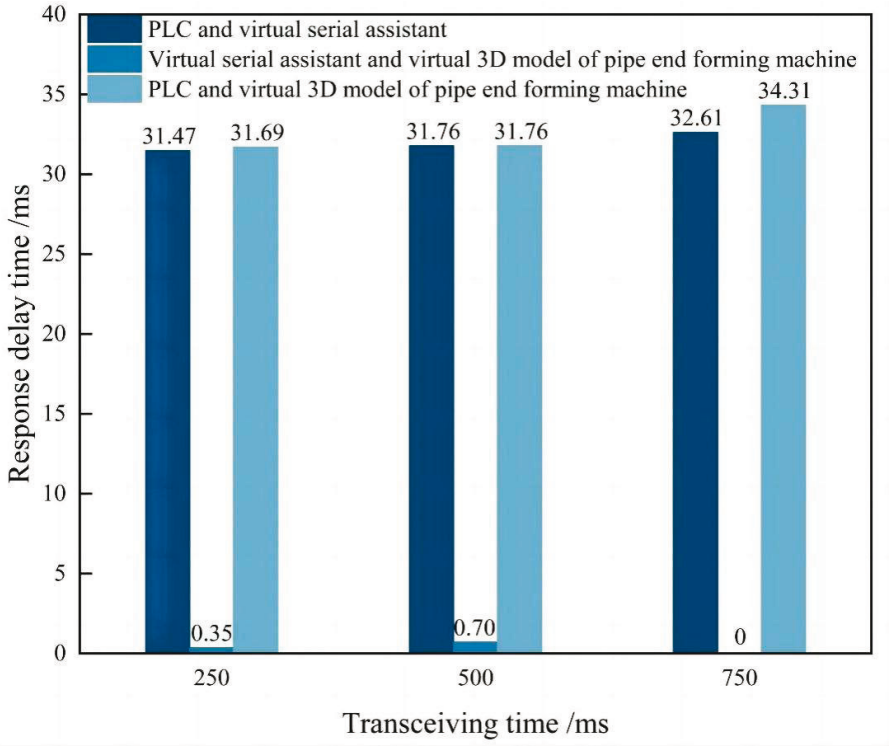


Fig. 17 Response delay time

Table 8 shows the proportion of effective data transmission between the PLC and the virtual serial assistant, the virtual serial assistant and the virtual 3D model of the pipe end forming machine, and the PLC and the virtual 3D model of the pipe end forming machine when the transceiving times are 250 ms, 500 ms and 750 ms. The results show that when the transceiving times are 250 ms, 500 ms and 750 ms, compared with the theoretical number of transmissions, the virtual serial assistant and virtual 3D model of the pipe end forming machine basically have no data loss, so the proportion of effective data transmission between the PLC and the virtual 3D model of the pipe end forming machine is similar to the proportion of effective data transmission between the PLC and the virtual serial assistant, ranging from 88.75% to 95.63%, which proves that the developed semi-physical simulation system has good stability. Furthermore, the data show that as the number of transmissions increases, the proportion of effective data transmissions also increases. When the transceiving times are 500 ms and 750 ms, the proportion of effective data transmission obtained by the two methods is almost the same, but it is greater than that when the transceiving time is 250 ms. When the transceiving time is 750 ms, the proportion of effective data transmission is 7.75% greater than that when the transceiving time is 250 ms. The results of the T-test analysis indicate that the significance probability $p < 0.05$, proving that the results are statistically significant.

Table 8 Proportion of effective data transmission

Transceiving time/ms	PLC and virtual serial assistant	Virtual serial assistant and virtual 3D model of pipe end forming machine	PLC and virtual 3D model of pipe end forming machine
250	88.82%	99.86%	88.75%
500	94.03%	99.86%	94.03%
750	95.83%	100%	95.63%

In sectors where latency, real-time data integrity, and system responsiveness are critical, even small percentage gains can lead to significant enhancements in overall system performance. These gains may manifest as reduced downtime, faster machine operation, and fewer data transmission errors. By significantly reducing latency, a 7.75% improvement in

transmission performance enables the system to respond more swiftly and accurately to real-time commands. This is particularly beneficial in semi-physical simulations or industrial automation systems, such as the pipe end forming machine. Additionally, this also provides a basis for setting the transceiving time when the semi-physical simulation system developed in this study is applied to the actual machine.

The above results shows that when the transceiving time is set to 750 ms, the system achieves the highest levels of latency and data transmission efficiency. To comprehensively evaluate the system's performance, a series of rigorous tests were conducted, aimed at assessing its reliability under extended periods of continuous use. The experiments specifically focused on evaluating the system's performance with a 750 ms transceiving time over a continuous 24-hour period. The key metrics analysed during this evaluation were data transmission efficiency and latency.

Table 9 indicates that with a 750 ms transceiving time, the system maintains a high level of reliability and stability over a continuous 24-hour period. The response time shows a slight but steady increase, increasing from 13.9 ms at the start to 14.4 ms after 24 hours, reflecting a total increase of 3.60%. Similarly, the proportion of effective data transmission begins at 94.5% and gradually decreases to 90.2%, indicating a 4.3% decline in efficiency. These changes, though moderate, suggest that the system experiences minor latency and a gradual reduction in data transmission efficiency over time. However, the overall performance remains within acceptable limits, indicating that the system is well suited for extended use in industrial applications where continuous operation is essential, with minimal performance degradation observed.

These observations prove that the semi-physical simulation system developed in this study has good real-time performance and stability.

Table 9 Key performance metrics during a 24-hour continuous operation test

Time/hours	Response time/ms	Percentage increase in response time/%	Proportion of effective data transmission/%
0	13.9	0	94.5
4	14.0	0.72	94.1
8	14.1	1.44	93.8
12	14.3	2.88	92.9
16	14.3	2.88	91.7
20	14.4	3.60	90.8
24	14.4	3.60	90.2

5. Conclusion

To address the problems of insufficient debugging time and difficulties in the debugging process for pipe end forming machines due to the increasingly prevalent development mode of personalised customisation and rapid delivery, a pipe end forming machine is taken as the research object in this study to develop a semi-physical simulation system.

(1) In accordance with the processing flow of a pipe end forming machine, PLC logic instructions are developed for the upsetting module, the spinning module, and the chamfering module. The Mitsubishi nonsequential communication protocol is used, and the data output format is defined. On the basis of the developed logic instructions, an upper computer program is developed and compiled for data transmission and reception.

(2) A virtual communication interface is developed for establishing the connection between the PLC upper computer and the virtual 3D model of the pipe end forming machine. This virtual 3D model is imported into 3ds Max for lightweight processing and subsequently imported into Unity 3D in FBX format. In Unity 3D, the virtual model of the pipe end forming

machine is employed for system functions such as environment construction, texture mapping, motion processing, and script execution.

(3) The results of experimental tests of the developed semi-physical simulation system show that the system can successfully simulate different working conditions. For different transceiving times, the response delay times between the PLC and the virtual 3D model range from 31.69 ms to 34.31 ms. By minimising the delay timer, the overall response time delay of the system can be reduced to approximately 14 ms. Compared with the theoretical number of transmissions, there is almost no data loss between the virtual serial assistant and the virtual 3D model of the pipe end forming machine. The proportion of effective data transmission between the PLC and the virtual 3D model of the pipe end forming machine varies from 88.75% to 95.63%, indicating that the system exhibits excellent real-time performance and stability. Extended 24-hour testing further confirmed the system's reliability, with only slight performance degradation over time. These findings indicate that the system is well suited for continuous industrial applications, offering reliable real-time monitoring and control.

(4) While the system was designed specifically for pipe end forming machines, its modular design suggests potential scalability. With necessary modifications to the PLC logic and virtual models, the system could be adapted to other types of machinery. However, the extent to which the system can scale up to larger industrial operations or different machines requires further research. For example, in larger-scale manufacturing environments, issues such as communication bandwidth, system processing capacity, and the handling of more complex machine geometries need to be addressed to ensure the system's continued efficiency.

(5) In this paper, the main processing modules of the pipe end forming machine are studied. In future work, the feeding device will be incorporated, and the semi-physical simulation system will undergo further development. The subsequent step is to apply the developed system in a real manufacturing facility for a pipe end forming machine to assess its actual impact on the production cycle.

The semi-physical simulation system developed in this paper can be used to test and verify the programming of a pipe end forming machine to ensure its quality and reliability. This approach enhances the safety factor for developers during equipment debugging, aids in preventing equipment damage caused by development or debugging errors, and reduces development costs. Consequently, the developed system can significantly shorten the development and debugging time for pipe end forming machines, solve the problems of debugging difficulties and insufficient debugging safety during the development process, and assist engineers who are unfamiliar with pipe end forming machines to swiftly acquire a profound understanding of this equipment.

Acknowledgements

This project was supported by the Scientific Research Fund of the Zhejiang Provincial Education Department (Grant Nos. Y202351058 and Y202454545), the Postgraduate Research and Innovation Project of Huzhou University (Grant No. 2024KYCX58), the Huzhou City Public Welfare Application Research Project (Grant Nos. 2024GZ19 and 2022GZ13), the Huzhou City "Unveiling and Commanding" Project (Grant No. 2022JB03), and the Zhejiang Provincial "Jianbing" "Lingyan" Research and Development Program of China (Grant No. 2022C01132).

REFERENCES

[1] Zhang, S.; Li, Y.; Yue, Z.; et al. 3D pipe forming of a new bending machine with a 3PUU-3RPS hybrid mechanism, Machines 2022, 10 (6), 470-488. https://doi.org/10.3390/machines10060470

[2] Zhou, W.; Shao, Z.; Yu, J.; et al. Advances and trends in forming curved extrusion profiles, Materials 2021, 14 (7), 1603. https://doi.org/10.3390/ma14071603

[3] Kuruppu, K.A.; Zhou, W.; Shi, Z.; et al. Review and Analysis of Manufacturing Curved Extrusion Components, International Conference on the Technology of Plasticity. Cham: Springer Nature Switzerland 2023, 392-403. https://doi.org/10.1007/978-3-031-42093-1_38

[4] Zhou, W.; Yu, J.; Lin, J.; et al. Manufacturing a curved profile with fine grains and high strength by differential velocity sideways extrusion, International Journal of Machine Tools and Manufacture 2019, 140, 77-88. https://doi.org/10.1016/j.ijmachtools.2019.03.002

[5] Hu, S.; Cheng, C.; Abd El-Aty, A.; et al. Forming characteristics of thin-walled tubes manufactured by free bending process-based nontangential rotation bending die, Thin-Walled Structures 2024, 194, 111313. https://doi.org/10.1016/j.tws.2023.111313

[6] Li Y. Q.; Li, F.; Kang, F.W.; et al. Recent research and advances in extrusion forming of magnesium alloys: A review, Journal of Alloys and Compounds 2023, 170080. https://doi.org/10.1016/j.jallcom.2023.170080

[7] Zhu, S.; Zhuang, X.; Xu, D.; et al. Flange forming at an arbitrary tube location through upsetting with a controllable deformation zone, Journal of Materials Processing Technology 2019, 273, 116230. https://doi.org/10.1016/j.jmatprotec.2019.05.011

[8] Ali, K.B.; Demir, B.; Gürün, H.; et al. Influence of Punch Shapes on Forces and Part Quality in the Punching of Explosively Welded DP600, Al1100, and DP600-Al1100 Composite Plates, Transactions of FAMENA 2022, 46 (2), 55-68. https://doi.org/10.21278/TOF.462039222

[9] Han, Z.; Li, H.; Xu, X.; et al. Crushing characteristics of aluminum/CFRP/aluminum hybrid tubes prepared by spinning forming, Composite Structures 2020, 249, 112551. https://doi.org/10.1016/j.compstruct.2020.112551

[10] Ning, F.; Shi, Q.; Kong, S.; et al. Experiment and analysis on edge chamfering rolling of sheet with various wedge angles, Materials Chemistry and Physics 2024, 313, 128699. https://doi.org/10.1016/j.matchemphys.2023.128699

[11] Gronostajski, Z.; Pater, Z.; Madej, L.; et al. Recent development trends in metal forming, Archives of Civil and Mechanical Engineering 2019, 19 (3), 898-941. https://doi.org/10.1016/j.acme.2019.04.005

[12] Yang, D.Y.; Bambach, M.; Cao, J.; et al. Flexibility in metal forming, Cirp Annals 2018, 67 (2), 743-765. https://doi.org/10.1016/j.cirp.2018.05.004

[13] Wu, W.J.; Chen, W.Z.; Zhang, L.X.; et al. Improvement of tension/compression asymmetry for high-performance ZK61 magnesium alloy rod via tailoring deformation parameters: Upsetting-extrusion temperature and upsetting ratio, Materials Science and Engineering: A 2021, 823, 141767. https://doi.org/10.1016/j.msea.2021.141767

[14] Miller, G.G. Tube forming processes: a comprehensive guide, Society of Manufacturing Engineers 2003.

[15] Lee, C.G.; Park, S.C. Survey on the virtual commissioning of manufacturing systems, Journal of Computational Design and Engineering 2014, 1 (3), 213-222. https://doi.org/10.7315/JCDE.2014.021

[16] Striffler, N.; Voigt, T. Concepts and trends of virtual commissioning–A comprehensive review, Journal of Manufacturing Systems 2023, 71, 664-680. https://doi.org/10.1016/j.jmsy.2023.10.013

[17] Scheifele, C.; Verl, A.; Riedel, O. Real-time co-simulation for the virtual commissioning of production systems, Procedia CIRP 2019, 79, 397-402. https://doi.org/10.1016/j.procir.2019.02.104

[18] Wang, J.; Niu, X.; Gao, R.X.; et al. Digital twin-driven virtual commissioning of machine tool, Robotics and Computer-Integrated Manufacturing 2023, 81, 102499. https://doi.org/10.1016/j.rcim.2022.102499

[19] Li, M. Semi-physical Virtual Simulation for Ship Straight Pipe Processing, 2022 IEEE 6th Advanced Information Technology, Electronic and Automation Control Conference (IAEAC). IEEE 2022, 1396-1401. https://doi.org/10.1109/IAEAC54830.2022.9929475

[20] Ergan, S.; Zou, Z.; Bernardes, S.D.; et al. Developing an integrated platform to enable hardware-in-the-loop for synchronous VR, traffic simulation and sensor interactions, Advanced Engineering Informatics 2022, 51, 101476. https://doi.org/10.1016/j.aei.2021.101476

[21] Leng, J.; Zhou, M.; Xiao, Y.; et al. Digital twins-based remote semi-physical commissioning of flow-type smart manufacturing systems, Journal of Cleaner Production 2021, 306, 127278. https://doi.org/10.1016/j.jclepro.2021.127278

[22] Wnorowski, J.; Łebkowski, A. Verification of the system for ship position keeping equipped with a set of anchors in Unity3d, Sensors 2022, 22 (19), 7421. https://doi.org/10.3390/s22197421

[23] Villarroel, A.; Toapanta, D.; Naranjo, S.; et al. Hardware in the loop simulation for bottle sealing process virtualized on Unity 3D, Electronics 2023, 12 (13), 2799. https://doi.org/10.3390/electronics12132799

[24] Salah, B.; Alsamhan, A.M.; Saleem, W.; et al. 3D Simulation of a Yogurt Filling Machine Using Grafcet Studio and Factory IO: Realization of Industry 4.0, Transactions of FAMENA 2023, 47 (3), 15-30. https://doi.org/10.21278/TOF.473049922

[25] Gao, H.; Fan, Q.; Chu, Z. Simulation Research on the Forming Process of Large Axles Rolled by Cross-Wedge Rolling, Transactions of FAMENA 2022, 46 (3), 63-80. https://doi.org/10.21278/TOF.463043422

[26] Chen, J.; Huo, D.; Liu, C.; et al. Design and modeling of hardware-in-loop test bench for hydraulic excavator based on dynamic load emulation, Automation in Construction 2022, 137, 104196. https://doi.org/10.1016/j.autcon.2022.104196

[27] Hu, W.; Cong, W.; Chen, X.; et al. Design and implementation of UAV semi-physical simulation system based on VxWorks, Chinese Intelligent Systems Conference. Singapore: Springer Nature Singapore 2023, 687-698. https://doi.org/10.1007/978-981-99-6847-3_60

[28] Zribi, F.; Sidhom, L.; Krama, A.; et al. Enhancement of drill string system operations with adaptive robust controller and hardware in-the-loop validation, Journal of Sound and Vibration 2023, 556, 117716. https://doi.org/10.1016/j.jsv.2023.117716

[29] Danielis, P.; Skodzik, J.; Altmann, V.; et al. Survey on real-time communication via ethernet in industrial automation environments, Proceedings of the 2014 IEEE Emerging Technology and Factory Automation (ETFA). IEEE 2014, 1-8. https://doi.org/10.1109/ETFA.2014.7005074

[30] Dawoud, D.S.; Dawoud, P. Serial communication protocols and standards, River Publishers 2022. https://doi.org/10.1201/9781003339496

[31] Mellado, J.; Núñez, F. Design of an IoT-PLC: A containerized programmable logical controller for the industry 4.0, Journal of Industrial Information Integration 2022, 25, 100250. https://doi.org/10.1016/j.jii.2021.100250

[32] Xu, L.; Wang, B.; Wang, L.; et al. PLC-SEIFF: A programmable logic controller security incident forensics framework based on automatic construction of security constraints, Computers & Security 2020, 92, 101749. https://doi.org/10.1016/j.cose.2020.101749

[33] Ramirez, R.; Chang, C.K.; Liang, S.H. PLC cybersecurity test platform establishment and cyberattack practice, Electronics 2023, 12 (5), 1195. https://doi.org/10.3390/electronics12051195

[34] Yuksel, H.; Altunay, Ö. Host-to-host TCP/IP connection over serial ports using visible light communication, Physical Communication 2020, 43, 101222. https://doi.org/10.1016/j.phycom.2020.101222

[35] Xie, C.; Zhang, D.; Yang, L.; et al. Precision seeding parameter monitoring system based on laser sensor and wireless serial port communication, Computers and Electronics in Agriculture 2021, 190, 106429. https://doi.org/10.1016/j.compag.2021.106429

Submitted: 26.7.2024

Accepted: 22.11.2024

Qingzhu Zhang[123]
Liang Zhang[2]
Yang Shen[2]
Yunhao Fu[4]
Liangyou Li[3]
Pengfei Zhang[23]
Yulan Wei[2*]
[1] College of Engineering, Northeast Agricultural University, Harbin, PR China
[2] School of Engineering, Huzhou University, Huzhou, PR China
[3] Zhejiang Changxing Heliang Intelligent Equipment Co. Ltd., Huzhou, PR China
[4] Sichuan Low Altitude Safety Technology Co. Ltd., Chengdu, PR China
*Corresponding author:
weiyl@zjhu.edu.cn