

Control Strategies for Digital Twin Systems

Guo-Ping Liu , Fellow, IEEE

Abstract—With the continuous breakthrough in information technology and its integration into practical applications, industrial digital twins are expected to accelerate their development in the near future. This paper studies various control strategies for digital twin systems from the viewpoint of practical applications. To make full use of advantages of digital twins for control systems, an architecture of digital twin control systems, adaptive model tracking scheme, performance prediction scheme, performance retention scheme, and fault tolerant control scheme are proposed. Those schemes are detailed to deal with different issues on model tracking, performance prediction, performance retention, and fault tolerant control of digital twin systems. Also, the stability of digital twin control systems is analysed. The proposed schemes for digital twin control systems are illustrated by examples.

Index Terms—Digital twin control systems, fault tolerant control, model tracking, performance prediction, performance retention.

I. INTRODUCTION

WITH the rapid development of information technology, the application scope of digital twins continues to expand. Simply speaking, the digital twin, which is a concept originating from industry, refers to the construction of completely consistent corresponding models, dynamic simulation, monitoring, analysis and control of physical entities in the real world through digital means. The digital twin emphasizes simulation, modelling, analysis, and decision-making of physical entities in the data world. Since this concept was proposed, digital twin technology has been evolving rapidly and had a huge driving effect on product design, manufacturing, and service. The digital twin is an innovative application that integrates various technologies, such as the Internet of Things, cloud computing, big data and artificial intelligence [1]. It can widely be applied in various sectors, e.g., manufacturing, energy, transportation, healthcare and agriculture, to improve efficiency, reduce costs, and enhance competitiveness.

The concept of digital twins can be traced back to the mirrored spaces model proposed in 2003, which is defined as a three-dimensional model that includes physical products, virtual products, and the connection between the physical and virtual products [2]. Due to limitations in technology and cog-

nition level at the time, this concept was not given much attention [3], and no related achievements were published in the following decade. In 2010, National Aeronautics and Space Administration (NASA) first introduced the concept of digital twins in the space technology roadmap with the aim of using digital twins to achieve comprehensive diagnostic maintenance of flight systems [4]. In 2011, the US Air Force Laboratory clearly proposed a digital twin example for future aircraft, pointing out that a complete virtual mapping of aircraft should be built, based on the high-precision simulation model, historical data and real-time sensor data of aircraft, so as to achieve the prediction of aircraft health status, remaining life and mission reachability [5]. Afterwards, the concept of digital twins began to receive widespread attention, and some research organisations began to study related key technologies [6], [7]. The applications of digital twins also expanded from aircraft operation and maintenance to rich scenarios, such as product development, equipment manufacturing, and smart cities [8]–[10]. As a rapidly developing emerging technology, digital twins have extensively been studied in terms of data fusion, modelling, control and other aspects.

Accurate modelling and control of digital twins require sampling data. In practice, digital twin systems are located in complex environments, and their sensing data have characteristics, such as multi-sources, heterogeneous, multi-scales, and high noises. Therefore, it is necessary to clean the data and handle issues, such as loss, redundancy, conflicts, and errors of data through various algorithms, e.g., machine learning algorithms with rule constraints. At the same time, the data collected by multiple sensors should be fused to improve the robustness and reliability of twin data and expand the modelling dimension of virtual entities [11]. The common multi-sensor fusion methods include fuzzy logic, neural networks, wavelet analysis, support vector machines [12], [13], etc. In digital twins, sensor data and models are typically fused and mapped on the basis of methods such as IoT middleware, feature extraction, and information fusion. For example, a model fusion method has been proposed to fuse multi-sensor data into a single model [14], based on semantic feature fusion, combining the characteristics of computer-aided design systems, computer-aided manufacturing systems, etc. Digital twin modelling is to create a high-precision virtual model to truly reproduce the geometry, attributes, behaviour and rules of a physical entity [15]. These models should not only be consistent with a physical entity in geometric structure, but also be able to simulate its spatiotemporal states, behaviours, and functions [16]. A five-dimensional digital twin model has been proposed [17], consisting of physical entities, virtual entities, connections, twin data and services. It emphasizes the driving role of twin data composed of physical data, virtual

Manuscript received June 6, 2023; revised July 22, 2023; accepted August 19, 2023. This work was supported in part by Shenzhen Key Laboratory of Control Theory and Intelligent Systems (ZDSYS20220330161800001) and the National Natural Science Foundation of China (62173255, 62188101). Recommended by Associate Editor Qing-Long Han.

Citation: G.-P. Liu, “Control strategies for digital twin systems,” *IEEE/CAA J. Autom. Sinica*, vol. 11, no. 1, pp. 170–180, Jan. 2024.

G.-P. Liu is with the Center for Control Science and Technology, Southern University of Science and Technology, Shenzhen 518055, China (e-mail: liugp@sustech.edu.cn).

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/JAS.2023.123834

data, service data and knowledge on physical entities, virtual devices and services. The application ideas and schemes of the five-dimensional digital twin model in multiple fields are widely recognized.

Digital twin-based control has been studied to improve the control performance of a system in recent years. A digital twin-based optimal state control method has been proposed to help a synchronized production operation system keep in an optimal state when uncertainties affect the system for obtaining the full element information needed for decision making [18]. A transparent digital twin has been designed for output control using the belief rule base, which is not only to model the complex relationships between the system inputs and output but also to conduct output control by identifying and optimizing the key parameters in the model inputs so that iterative optimizations are not necessarily needed [19]. A physics-based digital twin allows a digital twin instance for model predictive control of autonomous unmanned aerial vehicle landing to be informed by a truly dynamic flight model, rather than a less accurate set of steady-state aerodynamic force and moment data points [20]. A digital twin-driven method for online quality control in process industry has been presented to realise real-time monitoring, evaluation, and optimization of process parameters that are strongly related to product quality [21]. A web-based digital twin thermal power plant has been developed to provide a feasible and practicable route for monitoring and control of a thermal power plant via web browsers, processing data, visualizing processes, and sending commands to the control systems of the plant [22]. There are some similar concepts to digital twin control, for example, the optimal-tuning control [23], [24] and parallel control [25], [26]. The research work above has accelerated the use of digital twins in control systems.

Digital twins have been applied in scenarios, such as the service industry, supply chain management, intelligent workshops, product research and development, fault diagnosis, and smart cities. The research based on service theory enables digital twins to provide optimal services to customers. For example, a digital twin model was constructed for specific service applications based on a metamodel in unified modelling language (UML) format, providing services for the full lifecycle management of batteries [27]. Introducing the digital twin model into supply chain management monitors and predicts real-time changes in the supply chain to simplify business processes and improve decision-making efficiency [28]. The world's largest bearing manufacturer has applied the digital twin model to its distribution network [29]. Being driven by twin data in a workshop achieves the production management, activity planning, process control of the workshop [30], etc., effectively improving the transparency of the workshop production and optimizing the production process [31], [32]. Digital twins can improve the efficiency of equipment development and production in manufacturing enterprises through virtual simulation of manufacturing equipment and processes [33], [34], and provide the product quality inspection and support for managing and upgrading products throughout their entire lifecycles [35]. Fault prediction and maintenance of equipment are carried out, based on digital twins. By updat-

ing the operation data, fault data and maintenance data of a physical entity, its losses are calculated, the remaining life of equipment is predicted, and maintenance decisions are guided. In addition to the aforementioned fields, digital twin technology has potential applications in satellite/space communication networks, environmental protection, military operations, and other fields.

Although digital twins have made significant progress in recent years, there still exist a number of challenges on modelling, prediction, control of digital twin systems. Generally, digital twins are dynamic and need to interact with the data of the physical entity layer in real time, integrate the virtual and real, and evolve the model iteratively. In the future, the digital twin model will still need to strengthen research in several areas, such as aligning with industry standard architectures, establishing unified description methods and specifications. How to effectively and deeply combine high-precision sensing data with system mechanisms to achieve better state evaluation and system characterization is an urgent problem to be solved. Also, for faults of a real system, how to improve fault tolerance of the system using digital twin techniques is an important issue. To tackle those challenging issues above, this paper proposes various control strategies for digital twin systems, including the architecture of digital twin control systems, adaptive model tracking scheme, performance prediction scheme, performance retention scheme, and fault tolerant control scheme. The stability of digital twin systems is also analysed. Comprehensive simulations are provided to demonstrate the effectiveness of the proposed control strategies for digital twin systems.

II. ARCHITECTURE OF DIGITAL TWIN CONTROL SYSTEMS

Following the previous work on the optimal-tuning proportional integral derivative (PID) control method [23], [24], [36], a digital twin control system is presented, as shown in Fig. 1. This system mainly consists of three parts: 1) The physical subsystem that is a physical closed-loop feedback control system with a real plant and a controller; 2) The virtual subsystem that is a virtual closed-loop feedback control system with a digital plant and a controller; 3) The information exchange between the physical and virtual subsystems. In this digital twin control system, the real plant represents the practical process or plant to be controlled, and the digital plant is the model of the real plant, which can be identified off-line and/or on-line to track the dynamic variations of the real plant. The controller for the digital plant, denoted as D-controller, has the same structure as the controller for the real plant, denoted as R-controller, and the parameters of the former can be passed to the latter if needed. In fact, the real plant and digital plant are twin, and the R-controller and D-controller are twin as well. So, the physical subsystem and virtual subsystem are also twin.

Generally speaking, the digital twin control system is aimed to construct a corresponding digital control system with the same dynamic characteristics as a physical control system in the real world, fully utilizing physical mechanisms, sensing information and operational data. Through digital means, the physical control system can dynamically be simulated,

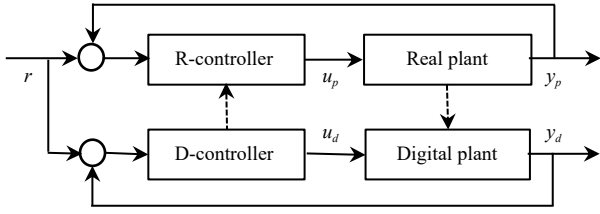


Fig. 1. The digital twin control system.

tracked, monitored, diagnosed, predicted, analyzed, synthesized, etc.

The real plant (or process) is described in a discrete form of

$$x_p(t+1) = f(x_p^{[n-1]}(t), u_p^{[m]}(t), \theta_p) \quad (1)$$

$$y_p(t) = g(x_p^{[n-1]}(t), \varphi_p) \quad (2)$$

where $x_p(t)$, $y_p(t)$ and $u_p(t)$ are the state vector, output vector and control input vector of the real plant, respectively, θ_p and φ_p denote the parameters of the real plant, $f(\cdot)$ and $g(\cdot)$ are the nonlinear function vectors, n and m are positive integers, and $v^{[k]}(t)$ of a signal $v(t)$ is defined as

$$v^{[k]}(t) = (v(t), v(t-1), \dots, v(t-k))$$

for $k \geq 0$.

In most cases, the nonlinear function vectors $f(\cdot)$ and $g(\cdot)$ of the real plant are unknown. Thus, the digital plant is modelled as

$$x_d(t+1) = \hat{f}(x_d^{[n-1]}(t), u_d^{[m]}(t), \theta_d) \quad (3)$$

$$y_d(t) = \hat{g}(x_d^{[n-1]}(t), \varphi_d) \quad (4)$$

where $x_d(t)$, $y_d(t)$ and $u_d(t)$ are the state vector, output vector and control input vector of the digital plant, respectively, θ_d and φ_d are its parameters, and $\hat{f}(\cdot)$ and $\hat{g}(\cdot)$ are the nonlinear function vectors. In the digital twin control system, the digital plant should track the dynamics of the real plant as accurately as possible.

Utilising nonlinear modelling methods, e.g., neural networks, the nonlinear function vectors $\hat{f}(\cdot)$ and $\hat{g}(\cdot)$ of the digital plant can be approximated by

$$\hat{f}(x_d^{[n-1]}(t), u_d^{[m]}(t), \theta_d) = \sum_{i=1}^{n_f} \theta_{d,i} f_i(x_d^{[n-1]}(t), u_d^{[m]}(t)) \quad (5)$$

$$\hat{g}(x_d^{[n-1]}(t), \varphi_d) = \sum_{i=1}^{n_g} \varphi_{d,i} g_i(x_d^{[n-1]}(t)) \quad (6)$$

where $f_i(x_d^{[n-1]}(t), u_d^{[m]}(t))$ and $g_i(x_d^{[n-1]}(t))$ are basis function vectors, $\theta_{d,i}$ and $\varphi_{d,i}$ are the weight matrices of the i -th basis function vector, n_f and n_g are the positive integers, $\theta_d = (\theta_{d,1}, \theta_{d,2}, \dots, \theta_{d,n_f})$ and $\varphi_d = (\varphi_{d,1}, \varphi_{d,2}, \dots, \varphi_{d,n_g})$. For example, for a linear state space system, let $n_f = 2$ and $n_g = 1$. The basis function vectors can be chosen as

$$f_1(x_d^{[n-1]}(t), u_d^{[m]}(t)) = x_d(t)$$

$$f_2(x_d^{[n-1]}(t), u_d^{[m]}(t)) = u_d(t)$$

$$g_1(x_d^{[n-1]}(t)) = x_d(t)$$

and the weight matrices of the basis function vectors are set to $\theta_{d,1} = A$, $\theta_{d,2} = B$ and $\varphi_{d,1} = C$, where A , B and C denote the matrices of a state space model.

The controller for the plant can be any kind of controllers, e.g., a PID controller, predictive controller and robust controller. A generic R-controller is described in the form of

$$z_p(t+1) = H(z_p^{[q-1]}(t), y_p^{[l]}(t), r(t), h_p) \quad (7)$$

$$u_p(t) = K(u_p^{[m-1]}(t-1), z_p^{[q-1]}(t), y_p^{[l]}(t), r(t), k_p) \quad (8)$$

where $z_p(t)$ is the state vector of the controller, h_p and k_p are the controller parameters, $H(\cdot)$ and $K(\cdot)$ are the nonlinear function vectors of the R-controller, l and q are positive integers, and $r(t)$ is the reference input.

Normally, the controller of the digital plant should have exactly the same structure as the one of the real plant. So, the D-controller is

$$z_d(t+1) = H(z_d^{[q-1]}(t), y_d^{[l]}(t), r(t), h_d) \quad (9)$$

$$u_d(t) = K(u_d^{[m-1]}(t-1), z_d^{[q-1]}(t), y_d^{[l]}(t), r(t), k_d) \quad (10)$$

where $z_d(t)$ is the state vector of the D-controller, h_d and k_d are the controller parameters. This controller should be designed to satisfy certain specifications, which include both system stability and control performance, using linear or nonlinear control methods.

For a linear proportional integral (PI) controller, the controller functions $H(\cdot)$ and $K(\cdot)$ can be designed in the form of

$$H(z^{[q-1]}(t), y^{[l]}(t), r(t), h) = h_z z(t) + h_e(r(t) - y(t)) \quad (11)$$

$$K(u^{[m-1]}(t-1), z^{[q-1]}(t), y^{[l]}(t), r(t), k) = k_z z(t) + k_e(r(t) - y(t)) \quad (12)$$

where $z(t)$ represents the variables $z_p(t)$ and $z_d(t)$, $y(t)$ represents the variables $y_p(t)$ and $y_d(t)$, h denotes the parameters h_z and h_e , and k denotes the parameters k_z and k_e .

In the digital twin control system, there exists the information exchange between the physical subsystem and virtual subsystem. Firstly, the relationship between the real plant and digital plant can be represented by

$$\theta_d = \hat{\theta}_d(\theta_p, \varphi_p) \quad (13)$$

$$\varphi_d = \hat{\varphi}_d(\theta_p, \varphi_p) \quad (14)$$

where $\hat{\theta}_d(\theta_p, \varphi_p)$ and $\hat{\varphi}_d(\theta_p, \varphi_p)$ are the functions of the parameters θ_p and φ_p of the real plant, which means that the model of the digital plant is related to the dynamic variations of the real plant. Generally speaking, the relationship (13) and (14) between the real plant and digital plant is unknown, but can be estimated by the method in Section III.

Secondly, the relationship between the R-controller and D-controller is generally denoted by

$$h_p = \hat{h}_p(h_d, k_d) \quad (15)$$

$$k_p = \hat{k}_p(h_d, k_d) \quad (16)$$

where $\hat{h}_p(h_d, k_d)$ and $\hat{k}_p(h_d, k_d)$ are the functions of the parameters h_d and k_d of the D-controller, which implies the R-controller parameters are updated through the D-controller. If the parameters of the R-controller and D-controller are the same, then $\hat{h}_p(h_d, k_d) = h_d$ and $\hat{k}_p(h_d, k_d) = k_d$.

For the digital twin control system shown in Fig. 1, compared with conventional control systems, there are several advantages in the following aspects: dynamics tracking, performance prediction, performance retention, and fault tolerant control of the real plant. These will be addressed in the following sections.

III. ADAPTIVE MODEL TRACKING

As the dynamics of a real plant may change with time or there may exist some uncertainties in the real plant (1) and (2), the model of the digital twin system should track the variations taking place in the plant. To keep both the digital and real plants having the same dynamics, an adaptive model tracking scheme is presented in Fig. 2, where the model parameters of the digital plant are adaptively updated with the dynamic variations of the real plant.

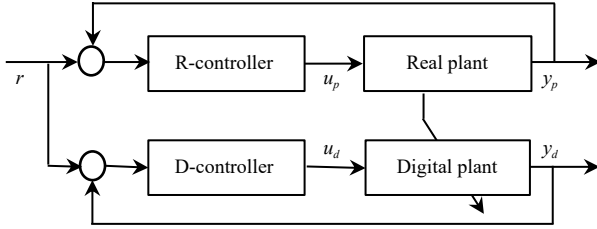


Fig. 2. Adaptive model tracking of the real plant.

Here, it is assumed that the structure and parameters of the R-controller and D-controller are exactly the same. Let the cost function of the model tracking be defined as

$$J_M(\theta_d, \varphi_d, t) = \sum_{k=1}^t J(\|\tilde{x}(k)\|, \|\tilde{y}(k)\|, k) \quad (17)$$

where the state error and output error are defined as

$$\tilde{x}(k) = x_p(k) - x_d(k) \quad (18)$$

$$\tilde{y}(k) = y_p(k) - y_d(k) \quad (19)$$

for $k = 1, 2, \dots, t$, $J(\cdot)$ is a function of measuring the model tracking performance, and $\|\cdot\|$ is the Euclidean norm.

The parameters of the digital model that minimize the cost function $J_M(\theta_d, \varphi_d, t)$ are

$$(\hat{\theta}_d(t), \hat{\varphi}_d(t)) = \operatorname{argmin} J_M(\theta_d, \varphi_d, t) \quad (20)$$

which can be determined by

$$\frac{\partial J_M(\theta_d, \varphi_d, t)}{\partial \theta_d} = 0 \quad (21)$$

$$\frac{\partial J_M(\theta_d, \varphi_d, t)}{\partial \varphi_d} = 0. \quad (22)$$

Generally, it is hard from the above to obtain the explicit expressions of the model parameters for the digital plant. Actually, the adaptive model reference control method can be

employed to update the model of the digital plant. In this case, the reference model is the real plant.

The model tracking can also be realised by system identification methods. Now, a single-input single-state single-output real plant is considered. Let the function $J(\cdot)$ be

$$J(\tilde{x}(k), \tilde{y}(k), k) = \alpha_k \tilde{x}(k)^2 + \beta_k \tilde{y}(k)^2 \quad (23)$$

where α_k and β_k are the time-varying weighting factors, which are chosen to balance the requirements on both the state error and the output error of the digital twin control system at time k . Also, let the model of the digital plant be expressed by (5) and (6) with a single input and a single output, which can be re-written as

$$x_d(t) = \theta_d^T F(t-1) \quad (24)$$

$$y_d(t) = \varphi_d^T G(t) \quad (25)$$

where

$$F(t-1) = [f_1(x_d^{[n-1]}(t-1), u_d^{[m]}(t-1)), \\ f_2(x_d^{[n-1]}(t-1), u_d^{[m]}(t-1)), \dots, \\ f_{n_f}(x_d^{[n-1]}(t-1), u_d^{[m]}(t-1))]^T$$

$$G(t) = [g_1(x_d^{[n-1]}(t)), g_2(x_d^{[n-1]}(t)), \dots, g_{n_g}(x_d^{[n-1]}(t))]^T$$

$$\theta_d = [\theta_{d,1}, \theta_{d,2}, \dots, \theta_{d,n_f}]^T, \quad \varphi_d = [\varphi_{d,1}, \varphi_{d,2}, \dots, \varphi_{d,n_g}]^T.$$

For the model tracking cost function (23), application of (21) and (22) to (17) results in

$$\sum_{k=1}^t \left(\alpha_k \tilde{x}^T(k) \frac{\partial x_d(k)}{\partial \theta_d} + \beta_k \tilde{y}^T(k) \frac{\partial y_d(k)}{\partial \theta_d} \right) = 0 \quad (26)$$

$$\sum_{k=1}^t \left(\alpha_k \tilde{x}^T(k) \frac{\partial x_d(k)}{\partial \varphi_d} + \beta_k \tilde{y}^T(k) \frac{\partial y_d(k)}{\partial \varphi_d} \right) = 0. \quad (27)$$

Let

$$\alpha_k = \lambda_{\theta}^{t-k}, \quad \beta_k = \lambda_{\varphi}^{t-k} \quad (28)$$

where $\lambda_{\theta} \in (0, 1]$ and $\lambda_{\varphi} \in (0, 1]$ are the forgetting factors. Following the recursive least squares algorithm [37], the parameter θ_d can recursively be estimated by

$$\hat{\theta}_d(t) = \hat{\theta}_d(t-1) + \left(\lambda_{\theta} + F^T(t-1) T_{\theta}(t-2) F(t-1) \right)^{-1} \\ \times T_{\theta}(t-2) F(t-1) (x_p(t) - \hat{\theta}_d^T(t-1) F(t-1)) \quad (29)$$

$$T_{\theta}(t-1) = \lambda_{\theta}^{-1} (T_{\theta}(t-2) (\lambda_{\theta} + F^T(t-1) T_{\theta}(t-2) \\ \times F(t-1))^{-1} T_{\theta}(t-2) F(t-1) \\ \times F^T(t-1) T_{\theta}(t-2)) \quad (30)$$

where $T_{\theta}(-2)$ is any positive definite matrix.

Similarly, the parameter φ_d can also be estimated utilising the above algorithm, that is

$$\hat{\varphi}_d(t) = \hat{\varphi}_d(t-1) + \left(\lambda_{\varphi} + G^T(t) T_{\varphi}(t-1) G(t) \right)^{-1} \\ \times T_{\varphi}(t-1) G(t) (y_p(t) - \hat{\varphi}_d^T(t-1) G(t)) \quad (31)$$

$$T_\varphi(t) = \lambda_\varphi^{-1} \left(T_\varphi(t-1) (\lambda_\varphi + G^T(t) T_\varphi(t-1) G(t))^{-1} \right. \\ \left. \times T_\varphi(t-1) G(t) G^T(t) T_\varphi(t-1) \right) \quad (32)$$

where $T_\varphi(-1)$ is any positive definite matrix.

IV. PERFORMANCE PREDICTION

The performance prediction of the real plant plays a very important role in control and scheduling of a digital twin system. Using the virtual subsystem, the future behaviour of the physical subsystem can be simulated in advance before the next sampling time. A performance prediction scheme is presented, as shown in Fig. 3. In this scheme, based on the information available up to time t , the future states, outputs and control inputs of the real plant from $t+1$ to $t+N$ can be predicted, where N is the prediction length.

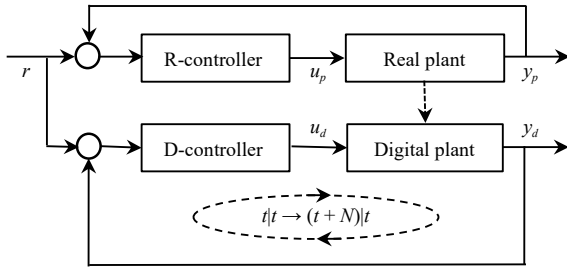


Fig. 3. Performance prediction of the real plant.

To make the presentation simple, it is assumed that the digital plant tracks the real plant very closely, and the structure and parameters of the R-controller and D-controller are exactly the same. Two cases are considered here.

A. The Plant States are Measurable

Given the state vector $x_p(t)$, output vector $y_p(t)$ and control input vector $u_p(t)$ of the real plant up to t , and reference input $r(t)$ up to $t+N$, the multi-step predictions of the state, output and control input vectors of the digital plant [38] are recursively calculated by

$$\hat{x}_d(t+i+1|t) = \hat{f}(\hat{x}_d^{[n-1]}(t+i|t), \hat{u}_d^{[m]}(t+i|t), \theta_d) \quad (33)$$

$$\hat{z}_d(t+i+1|t) = H(\hat{z}_d^{[q-1]}(t+i|t), \hat{y}_d^{[l]}(t+i|t), r(t+i), h_d) \quad (34)$$

$$\hat{y}_d(t+i+1|t) = \hat{g}(\hat{x}_d^{[n-1]}(t+i+1|t), \varphi_d) \quad (35)$$

$$\hat{u}_d(t+i+1|t) = K(\hat{u}_d^{[m-1]}(t+i|t), \hat{z}_d^{[q-1]}(t+i+1|t), \hat{y}_d^{[l]}(t+i+1|t), r(t+i+1), k_d) \quad (36)$$

for $i = 0, 1, 2, \dots, N-1$, where

$$\hat{x}_d^{[n-1]}(t|t) = x_p^{[n-1]}(t)$$

$$\hat{z}_d^{[q-1]}(t|t) = z_p^{[q-1]}(t)$$

$$\hat{u}_d^{[m-1]}(t|t) = u_p^{[m-1]}(t)$$

$\hat{v}(t+i|t)$ represents the i -th step ahead prediction of the signal $v(t)$ at time t ,

$$\hat{v}^{[k]}(t+i|t) = (\hat{v}(t+i|t), \hat{v}(t+i-1|t), \dots, \hat{v}(t+i-k|t)) \\ \text{and } \hat{v}(t+i|t) = v(t+i) \text{ if } i \leq 0.$$

B. The Plant States are Unmeasurable

If the states of the real plant are not measurable, a state observer is employed, i.e.,

$$\hat{x}_p(t+1|t) = f(\hat{x}_p^{[n-1]}(t|t-1), u_p^{[m]}(t), \theta_p) \\ + L(y_p(t) - g(\hat{x}_p^{[n-1]}(t|t-1), \varphi_p)) \quad (37)$$

where L is the observer gain matrix. If the nonlinear functions $f(\cdot)$ and $g(\cdot)$ of the plant are unknown, the state observer is structured by

$$\hat{x}_p(t+1|t) = \hat{f}(\hat{x}_p^{[n-1]}(t|t-1), u_p^{[m]}(t), \theta_d) \\ + L(y_p(t) - \hat{g}(\hat{x}_p^{[n-1]}(t|t-1), \varphi_d)). \quad (38)$$

In this case, the prediction calculations of the states, outputs and control inputs of the digital plant are the same as those used in Case 1 except that

$$\hat{x}_d(t|t) = \hat{x}_p(t|t-1).$$

In the two cases above, since the virtual subsystem tracks the physical subsystem, the predictions of the states, outputs and control inputs of the real plant can approximately be estimated by the following:

$$\hat{x}_p(t+i|t) = \hat{x}_d(t+i|t)$$

$$\hat{y}_p(t+i|t) = \hat{y}_d(t+i|t)$$

$$\hat{u}_p(t+i|t) = \hat{u}_d(t+i|t)$$

$$\text{for } i = 1, 2, \dots, N.$$

In this way, the performance prediction scheme can predict the performance (the states, outputs and control inputs) of the real plant through the virtual subsystem in advance. The prediction accuracy depends on how the model of the digital plant tracks the dynamics and environment of the real plant accurately.

V. PERFORMANCE RETENTION

Since the dynamics and environment of a real plant often vary with time, the system performance may degrade from its desired one. To deal with this issue, a performance retention scheme is proposed, as shown in Fig. 4.

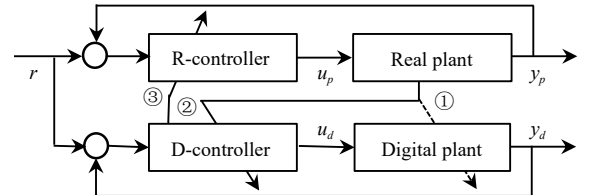


Fig. 4. Performance retention of the real plant.

In practice, there are several objectives which a digital twin control system needs to achieve. The corresponding cost function vector of the control performance for the physical subsystem is denoted by

$$J_p(t, \alpha_p) = [\phi_1(t, \alpha_p), \phi_2(t, \alpha_p), \dots, \phi_{n_r}(t, \alpha_p)]^T \quad (39)$$

where $\phi_i(t, \alpha_p)$ is the i -th individual cost function of the real plant, $\alpha_p = \{h_p, k_p\}$ is the R-controller, n_r is a positive integer.

The performance criterion of the physical subsystem may practically be stated as

$$|J_p(t, \alpha_p) - J^*| \leq \varepsilon \quad (40)$$

where J^* is the desired performance vector and ε is the tolerable performance error vector.

If the real plant is unknown, it is hard to search the controller parameters to satisfy criterion (40). To avoid this difficulty, an alternative cost function vector of the control performance is defined via the virtual subsystem as follows:

$$J_d(t, \alpha_d) = [\psi_1(t, \alpha_d), \psi_2(t, \alpha_d), \dots, \psi_{n_r}(t, \alpha_d)]^T \quad (41)$$

where $\psi_i(t, \alpha_d)$ is the i -th individual cost function of the digital plant, $\alpha_d = \{h_d, k_d\}$ are the D-controller parameters. Generally, the individual cost functions $\phi_i(t, \alpha_p)$ and $\psi_i(t, \alpha_d)$ should be similar.

In addition, there may still exist some freedom for the controller design after satisfying criterion (40). Taking advantages of the digital twin control system, the optimal parameters of the controller may be formulated as

$$\alpha_d^* = \operatorname{argmin}_{\alpha_d} (J_d(t, \alpha_d) - J^*)^T W_d (J_d(t, \alpha_d) - J^*) \quad (42)$$

$$\text{s.t. } |\psi_i(t, \alpha_d) - J_i^*| \leq \varepsilon_i \quad (43)$$

for $i = 1, 2, \dots, n_r$, where W_d is the weighting matrix and is normally chosen as

$$W_d = \operatorname{diag}\{\varepsilon_1^{-2}, \varepsilon_2^{-2}, \dots, \varepsilon_{n_r}^{-2}\}.$$

The optimization problem given in (42) and (43) can be solved by employing optimization methods [39], for example, the method of inequalities.

When one of the following criteria fails:

$$|\phi_i(t, \alpha_p) - J_i^*| \leq \varepsilon_i \quad (44)$$

for $i = 1, 2, \dots, n_r$, to maintain the performance of the physical subsystem, the performance retention scheme needs to take the following procedure:

- ① The model of the digital plant needs to be updated;
- ② The D-controller parameters $\alpha_d = \{h_d, k_d\}$ should be re-tuned to be $\alpha_d^* = \{h_d^*, k_d^*\}$ via solving the optimization problem (42) and (43);
- ③ Let the R-controller parameters $\alpha_p = \{h_p, k_p\}$ be replaced by the optimal parameters $\alpha_d^* = \{h_d^*, k_d^*\}$.

The three steps of the performance retention listed above are also marked in Fig. 4.

VI. FAULT TOLERANT CONTROL

The digital twin control system has more advantages than a conventional control system on coping with system faults. This is because there exists a virtual subsystem in the digital twin control system. The fault parts of the physical subsystem can be replaced by the corresponding parts of the virtual subsystem. Two faults of a real practical system are considered here: one is the sensor fault and the other is the controller fault.

A. Sensor Faults

In this case, it assumes that the sensor of measuring the output of the real plant fails. If a conventional control method is employed, the physical subsystem becomes an open-loop control system, the stability and performance of the system will change significantly, and it may collapse in a bad situation. Here, a sensor-fault tolerant control scheme is presented, as shown in Fig. 5. In this scheme, the output of the real plant measured by the failed sensor is replaced by the output of the digital plant. At the same time, the adaptive model tracking scheme may be needed to ensure the difference between the real plant and the digital plant is within a tolerated bound.

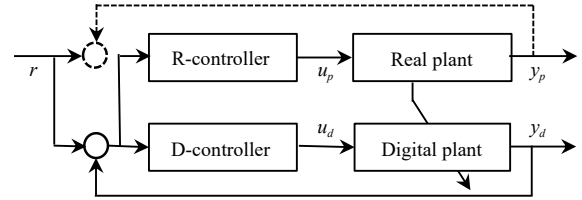


Fig. 5. Sensor-fault tolerant control of the real plant.

The output $y_p(t)$ of the real plant is set to be the output $y_d(t)$ of the digital plant in the R-controller, that is

$$z_p(t+1) = H(z_p^{[q-1]}(t), y_d^{[l]}(t), r(t), h_p) \quad (45)$$

$$u_p(t) = K(u_p^{[m-1]}(t-1), z_p^{[q-1]}(t), y_d^{[l]}(t), r(t), k_p). \quad (46)$$

In this way, the real plant is controlled with the support of the virtual subsystem running in parallel so that the stability and performance of the real plant can be maintained within a tolerant range for at least a certain period of time.

B. Controller Faults

In a conventional control system, if the controller of the real plant has a fault, the system has to stop. For example, in the case that the hardware of the controller is damaged or the software of the controller is crashed, the system has to stop. For a practical system, this may lead to enormously economical loss or catastrophe. So, a controller-fault tolerant control scheme for a digital twin control system with a plant controller failure is presented, as shown in Fig. 6.

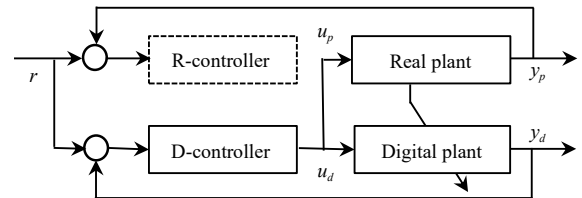


Fig. 6. Controller-fault tolerant control of the real plant.

When the R-controller does not work, the control input of the real plant is set to the output of the D-controller, i.e.,

$$u_p(t) = u_d(t) \quad (47)$$

which also leads to similar performance to the sensor-fault tolerant control scheme.

The modelling accuracy of the digital plant is crucial for the

fault tolerant control schemes introduced in the two cases above because it will impact the performance of the real plant significantly. Thus, the model tracking scheme plays a very important role in the fault tolerant control of digital twin systems.

VII. STABILITY ANALYSIS OF DIGITAL TWIN SYSTEMS

For the schemes discussed in the previous sections, their closed-loop digital twin control systems are different because the different control schemes are employed. But, the stability analysis procedure of those systems is quite similar. To simplify this procedure, based on the general structure of the digital twin control system proposed in Section II, its stability is discussed here.

The outputs (2) and (4) of the real plant and digital plant lead to

$$y_p^{[l]}(t) = g^{[l]}(x_p^{[n-1]}(t), \varphi_p) \quad (48)$$

$$y_d^{[l]}(t) = \hat{g}^{[l]}(x_d^{[n-1]}(t), \varphi_d). \quad (49)$$

Substituting (48) into the R-controller (7) and (8) results in

$$z_p(t+1) = H(z_p^{[q-1]}(t), g^{[l]}(x_p^{[n-1]}(t), \varphi_p), r(t), h_p) \quad (50)$$

$$u_p(t) = K(u_p^{[m-1]}(t-1), z_p^{[q-1]}(t), g^{[l]}(x_p^{[n-1]}(t), \varphi_p), r(t), k_p) \quad (51)$$

and also substituting (49) into the D-controller (9) and (10) yields

$$z_d(t+1) = H(z_d^{[q-1]}(t), \hat{g}^{[l]}(x_d^{[n-1]}(t), \varphi_d), r(t), h_d) \quad (52)$$

$$u_d(t) = K(u_d^{[m-1]}(t-1), z_d^{[q-1]}(t), \hat{g}^{[l]}(x_d^{[n-1]}(t), \varphi_d), r(t), k_d) \quad (53)$$

Combining the equations of the real plant (1), digital plant (3), R-controller (50) and (51) and D-controller (52) and (53) leads to a high-order nonlinear system

$$X(t+1) = F(X^{[M-1]}(t), r(t), \Theta) \quad (54)$$

where

$$X(t) = [x_p^T(t), x_d^T(t), z_p^T(t), z_d^T(t), u_p^T(t), u_d^T(t)]^T \quad (55)$$

$F(X(t), r(t), \Theta)$

$$= \begin{bmatrix} f(x_p^{[n-1]}(t), u_p^{[m]}(t), \theta_p) \\ \hat{f}(x_d^{[n-1]}(t), u_d^{[m]}(t), \theta_d) \\ H(z_p^{[q-1]}(t), g^{[l]}(x_p^{[n-1]}(t), \varphi_p), r(t), h_p) \\ H(z_d^{[q-1]}(t), \hat{g}^{[l]}(x_d^{[n-1]}(t), \varphi_d), r(t), h_d) \\ K(u_p^{[m-1]}(t-1), z_p^{[q-1]}(t), g^{[l]}(x_p^{[n-1]}(t), \varphi_p), r(t), k_p) \\ K(u_d^{[m-1]}(t-1), z_d^{[q-1]}(t), \hat{g}^{[l]}(x_d^{[n-1]}(t), \varphi_d), r(t), k_d) \end{bmatrix} \quad (56)$$

$$\Theta = [\theta_p, \varphi_p, \theta_d, \varphi_d, h_p, k_p, h_d, k_d]$$

$$M = \max\{n, m+1, l, p, q, n\}. \quad (57)$$

Clearly, the stability of the digital twin control system is determined by the closed-loop equation given in (54). Further stability analysis on (54) can be done using nonlinear system methods [40], for example, the nonlinear Lyapunov method.

VIII. EXAMPLES

To demonstrate the performance of the control strategies for digital twin systems proposed in this paper, the following nonlinear plant [41] is considered:

$$y_p(t+1) = \frac{\theta_{p,1}y_p(t)y_p(t-1)}{1+y_p^2(t-1)} + \theta_{p,2}u_p(t) + \theta_{p,3}u_p(t-1)$$

where $y_p(t)$ and $u_p(t)$ are the output and control input of the real plant, respectively, and the plant parameters are $\theta_{p,1} = 0.5$, $\theta_{p,2} = 1.5$ and $\theta_{p,3} = 0.3$. In terms of the real plant form in (1) and (2), the nonlinear functions $f(\cdot)$ and $g(\cdot)$ can be chosen as

$$f(x_p^{[1]}(t), u_p^{[1]}(t), \theta_p) = \frac{\theta_{p,1}x_p(t)x_p(t-1)}{1+x_p^2(t-1)} + \theta_{p,2}u_p(t) + \theta_{p,3}u_p(t-1)$$

$$g(x_p^{[1]}(t), \varphi_p) = \varphi_p x_p(t)$$

where $\varphi_p = 1$, which means that the output of the plant is also the state of the plant.

A linear model for the digital plant is introduced to approximate the real plant with basis functions chosen to be $x_d(t)$, $x_d(t-1)$, $u_d(t)$ and $u_d(t-1)$. The functions $\hat{f}(\cdot)$ and $\hat{g}(\cdot)$ of the digital plant are constructed as

$$\hat{f}(x_d^{[1]}(t), u_d^{[1]}(t), \theta_d) = \theta_{d,1}x_d(t) + \theta_{d,2}x_d(t-1) + \theta_{d,3}u_d(t) + \theta_{d,4}u_d(t-1)$$

$$\hat{g}(x_d^{[1]}(t), \varphi_d) = \varphi_d x_d(t)$$

where $\varphi_d = 1$ and parameters $\theta_d = [\theta_{d,1}, \theta_{d,2}, \theta_{d,3}, \theta_{d,4}]^T$ need to be estimated.

Using the high-order fully actuated control method [41], the D-controller for the virtual subsystem is designed as

$$z_d(t+1) = z_d(t) + h_d(r(t) - y_d(t)) \quad (58)$$

$$u_d(t) = ((K_{dp} - \theta_{d,1})y_d(t) - \theta_{d,2}y_d(t-1) + K_{di}z_d(t) - \theta_{d,4}u_d(t-1))/\theta_{d,3} \quad (59)$$

where $h_d = 1$, K_{dp} and K_{di} are the PI control parameters to be tuned.

Following the D-controller, the R-controller is constructed as:

$$z_p(t+1) = z_p(t) + h_p(r(t) - y_p(t)) \quad (60)$$

$$u_p(t) = ((K_{pp} - \theta_{d,1})y_p(t) - \theta_{d,2}y_p(t-1) + K_{pi}z_p(t) - \theta_{d,4}u_p(t-1))/\theta_{d,3} \quad (61)$$

where $h_p = 1$, K_{pp} and K_{pi} are the PI control parameters.

The desired control performance is assumed to be the step response of a standard second-order system with damping

ratio of 0.7 and undamped frequency of 5 rad/s. Thus, the transfer function between the desired output $y_r(t)$ and the reference input $r(t)$ is

$$\frac{Y_r(s)}{R(s)} = \frac{25}{s^2 + 7s + 25}$$

which can be discretised with a sampling period of 0.04 s to be

$$\begin{aligned} y_r(t+2) - 1.7210y_r(t+1) + 0.7558y_r(t) \\ = 0.0182r(t+1) + 0.0166r(t). \end{aligned} \quad (62)$$

The reference input $r(t)$ is assumed to be a square wave signal

$$r(t) = \begin{cases} 1, & t \in [0, 50) \text{ or } [100, 150) \\ 0, & t \in [50, 100) \text{ or } [150, 200). \end{cases}$$

The initial conditions of the states, outputs and control inputs of the digital twin control system are zeros. For the real plant with $\theta_p = [0.5, 1.5, 0.3]^T$, the initial parameters of the digital plant are set to

$$\theta_d = [1.3178, -0.3866, 1.4238, -1.2606]^T$$

which are estimated using an offline least squares method.

Employing a PI control tuning method (e.g., the trial-and-error method), a set of controller parameters that achieve the above desired control performance given by (62) with an acceptable error is obtained below:

$$K_{dp} = 0.75, K_{di} = 0.035, K_{pp} = 0.75, K_{pi} = 0.035. \quad (63)$$

Here, based on the proposed digital twin control system, the five cases are considered, including adaptive model tracking, performance prediction, performance retention, sensor-fault tolerant control, and controller-fault tolerant control.

Case 1: Adaptive model tracking

Assume the parameters of the real plant change from $\theta_p = [0.5, 1.5, 0.3]^T$, for $t \in [0, 100)$ to $\theta_p = [0.8, 0.7, 0.5]^T$, for $t \in [100, 200]$.

The digital plant is redescribed by

$$x_d(t) = \theta_d^T F(t-1)$$

where

$$F(t-1) = [x_d(t-1), x_d(t-2), u_d(t-1), u_d(t-2)]^T$$

$$\theta_d = [\theta_{d,1}, \theta_{d,2}, \theta_{d,3}, \theta_{d,4}]^T.$$

The adaptive model tracking scheme in Section III is applied to the digital twin control system in Fig. 1. The parameters θ_d of the digital model are recursively estimated online by (29) and (30) with $\lambda_\theta = 0.95$ and $T_\Theta(-2) = 10^6 I$. The output responses of the physical subsystem with controller (58) and (59) and virtual subsystem with controller (60) and (61) are almost the same, as shown in Fig. 7, where the parameters used in the digital plant, the R-controller and D-controller are updated with time but the PI parameters are fixed as in (63). The parameter estimation of the digital plant is shown in Fig. 8. Clearly, the adaptive model tracking scheme performs well even if the parameters of the real plant vary more than 50%.

Case 2: Performance prediction

In terms of the predictor (33)–(36) introduced in Section IV,

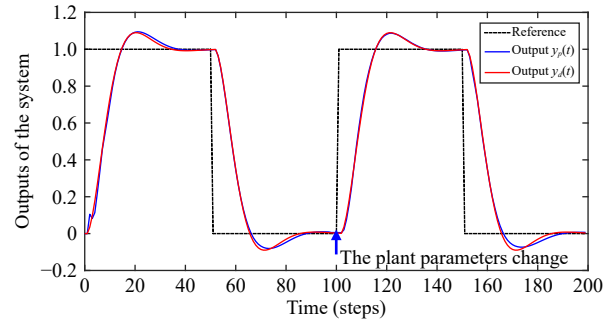


Fig. 7. The output responses of the system (Case 1).

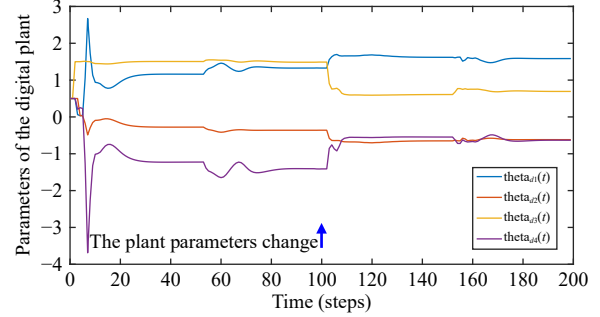


Fig. 8. Parameter estimation of the digital plant (Case 1).

the 10-step ahead predictions of the states, output and control input of the data plant are calculated via the virtual subsystem, i.e.,

$$\begin{aligned} \hat{x}_d(t+i+1|t) &= \theta_{d,1}\hat{x}_d(t+i|t) + \theta_{d,2}\hat{x}_d(t+i-1|t) \\ &\quad + \theta_{d,3}\hat{u}_d(t+i|t) + \theta_{d,4}\hat{u}_d(t+i-1|t) \end{aligned}$$

$$\hat{z}_d(t+i+1|t) = \hat{z}_d(t+i|t) + r(t+i) - \hat{y}_d(t+i|t)$$

$$\hat{y}_d(t+i+1|t) = \hat{x}_d(t+i+1|t)$$

$$\begin{aligned} \hat{u}_d(t+i+1|t) &= ((K_p - \theta_{d,1})\hat{y}_d(t+i+1|t) \\ &\quad - \theta_{d,2}\hat{y}_d(t+i|t) + K_i\hat{z}_d(t+i+1|t) \\ &\quad - \theta_{d,4}\hat{u}_d(t+i|t))/\theta_{d,3} \end{aligned}$$

for $i = 0, 1, \dots, 9$, where $\hat{x}_d(t|t) = y_p(t)$.

Let $\theta_p = [0.5, 1.5, 0.3]^T$. The parameters θ_d of the digital model are estimated using the adaptive tracking scheme in Case 1. The 10th-step ahead output predictions of the digital plant and the output of the real plant with controller (60) and (61) are shown in Fig. 9. The control input of the real plant and the control input predictions of the digital plant are given in Fig. 10. It is clear that the difference between the output and its prediction is very small except the beginning part ($t < 10$) because the first 10-step output predictions are set to zero.

Case 3: Performance retention

To evaluate the control performance, two simple cost functions are introduced as

$$J_p(t, \alpha_p) = \phi_1(t, \alpha_p) = \frac{1}{N+1} \sum_{k=t-N}^t (y_r(k) - y_p(k))^2$$

$$J_d(t, \alpha_d) = \psi_1(t, \alpha_d) = \frac{1}{N+1} \sum_{k=t-N}^t (y_r(k) - y_d(k))^2$$

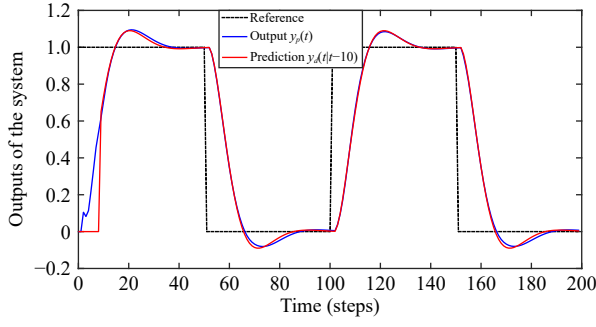


Fig. 9. The output responses of the system (Case 2).

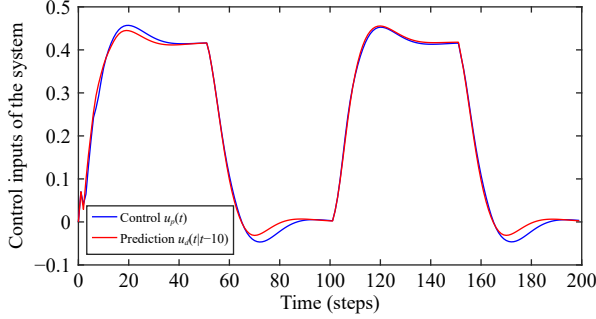


Fig. 10. The control inputs of the system (Case 2).

where $\alpha_p = (K_{pp}, K_{pi})$ and $\alpha_d = (K_{dp}, K_{di})$, the desired output $y_r(k)$ is calculated by the desired system (62), N is a positive integer and should not be less than the settling time of the desired system. Here let $N = 200$, the desired performance and tolerable performance error be

$$J^* = 0 \text{ and } \varepsilon = 0.0003.$$

Let the parameters $\theta_p = [0.5, 1.5, 0.3]^T$ of the real plant be fixed and the parameters of the digital plant be set to $[1.3178, -0.3866, 1.4238, -1.2606]^T$. At the beginning, the controller parameters are still $K_{pp} = K_{dp} = 0.75$ and $K_{pi} = K_{di} = 0.035$. Clearly, the control performance of both the physical and virtual subsystems is significantly different from the desired performance given by (62) before $t = 100$, as shown in Fig. 11.

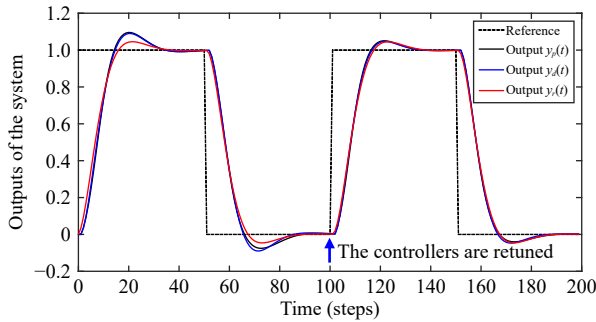


Fig. 11. The output responses of the system (Case 3).

At $t = 100$, employing the performance retention scheme in Section V and the genetic-algorithm-based optimization method [38], the parameters of the D-controller are optimally returned to be

$$K_{dp}^* = 0.7, K_{di}^* = 0.04$$

so that the performance function $\psi_1(t, \alpha_d)$ is minimised with $\psi_1(t, \alpha_d^*) \leq 0.0003$. At the same time, the parameters of the R-controller are updated to be

$$K_{pp}^* = 0.7, K_{pi}^* = 0.04 \text{ with } \phi_1(t, \alpha_p^*) \leq 0.0003.$$

The control performance is provided in Fig. 11. The results show that both the physical subsystem and virtual subsystem have the desired control performance after time at $t = 100$.

Case 4: Sensor-fault tolerant control

In this case, the parameters of the real plant are still $\theta_p = [0.5, 1.5, 0.3]^T$, the parameters θ_d of the digital model are estimated online using the adaptive model tracking scheme with $\lambda_\theta = 0.99$.

When the output sensor of the real plant is out of order at $t = 80$, let $y_p(t) = 0$ in the R-controller. Then, the output of the real plant diverges from the desired output $y_r(t)$ gradually, as shown in Fig. 12. If the sensor-fault tolerant control scheme in Section VI is applied, the output of the digital plant replaces the real output in the R-controller (60) and (61), i.e.,

$$u_p(t) = ((K_{dp} - \theta_{d,1})y_d(t) - \theta_{d,2}y_d(t-1) + K_{di}z_p(t) - \theta_{d,4}u_p(t-1)) / \theta_{d,3}$$

$$z_p(t+1) = z_p(t) + r(t) - y_d(t).$$

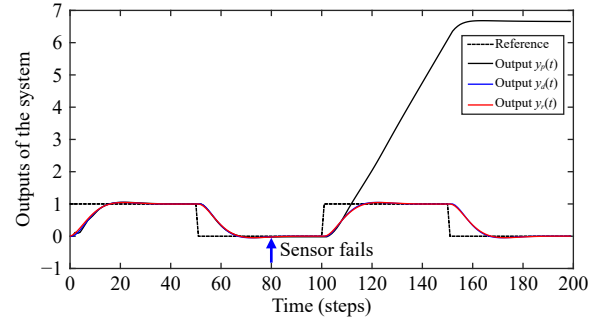


Fig. 12. The output responses of the system without sensor-fault tolerant control (Case 4).

The results in Fig. 13 illustrate that the control performance of the physical subsystem is still close to the desired one when the output sensor of the real plant fails at $t = 80$.

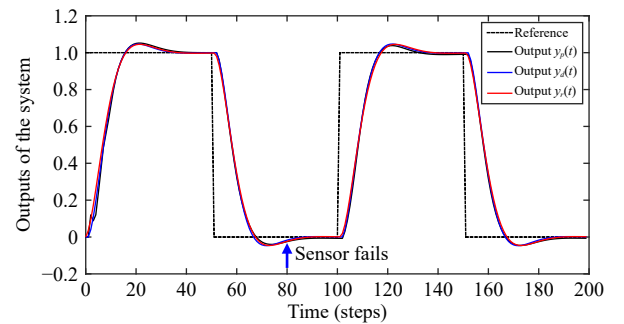


Fig. 13. The output responses of the system with sensor-fault tolerant control (Case 4).

Case 5: Controller-fault tolerant control

The system setup is the same as in Case 4. When the R-controller does not work at time $t = 80$, let the output of the R-controller be a random signal between -0.5 and 0.5 . The physical subsystem collapses immediately, as shown in Fig. 14. If the controller-fault tolerant control scheme in Section VI is employed, the output of the D-controller is directly applied to the control input of the real plant, i.e.,

$$u_p(t) = u_d(t) = ((K_{dp} - \theta_{d,1})y_d(t) - \theta_{d,2}y_d(t-1) + K_{di}z_d(t) - \theta_{d,4}u_d(t-1))/\theta_{d,3}$$

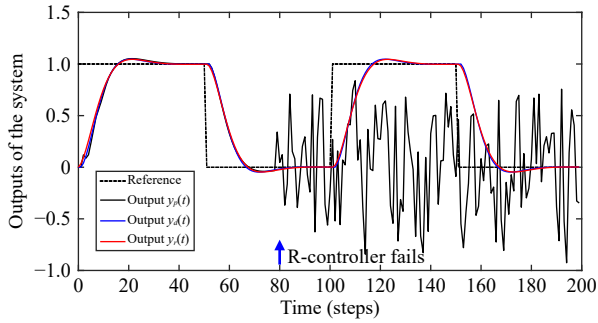


Fig. 14. The output responses of the system without controller-fault tolerant control (Case 5).

$$z_d(t+1) = z_d(t) + h_d(r(t) - y_d(t)).$$

The results in Fig. 15 demonstrates that the physical subsystem still follows the desired control performance well after the R-controller fails $t = 80$.

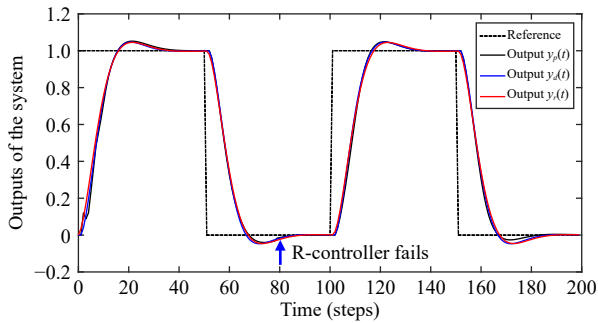


Fig. 15. The output responses of the system with controller-fault tolerant control (Case 5).

IX. CONCLUSIONS

This paper has addressed some challenging issues in modelling, prediction and control of digital twin systems. A novel architecture of digital twin control systems has been proposed to provide a fundamental structure for practical applications of digital twins in control systems. To make full use of the advantages of digital twins, several schemes for digital twin control systems have been presented, including adaptive model tracking scheme, performance prediction scheme, performance retention scheme, and fault tolerant control scheme. The adaptive model tracking scheme can track the dynamics variations of real plants in an iterative online estimation way. The performance prediction scheme can predict the future per-

formance of the physical subsystem via the virtual subsystem. The performance pretention scheme can maintain the desired control performance when the dynamics of the real plant varies. The fault tolerant control scheme can cope with a sensor fault and controller fault of the physical subsystem with the support provided by the virtual subsystem. The stability of digital twin control systems has been analysed as well. A number of simulation results have demonstrated the performance of the digital twin control systems with different schemes above.

To put the proposed schemes into practice, more research work on physical experiments and practical applications is needed and will be carried out in the near future. Also, there are other issues that are not discussed in this paper, for example, uncertainties, modelling errors, disturbances and noises, which impact the stability and robustness of a digital twin control system. In addition, compared with other conventional control systems, there are some disadvantages, for example, increased consumption of computing resources and reduced data security. A digital twin control system requires a large number of computing resources, which may become limiting factors in large-scale applications, leading to an increase in computing hardware cost. It involves much system information in the process of data collection and use, and the protection of this information is an important issue in the application of the digital twin control system. Although there still exist a number of challenging issues above, the proposed architecture and schemes have provided fundamental base for further study on digital twin control systems.

REFERENCES

- [1] A. Redelinghuys, A. Basson, and K. Kruger, "A six-layer architecture for the digital twin: A manufacturing case study implementation," *J. Intelligent Manufacturing*, vol. 31, no. 6, pp. 1383–1402, 2020.
- [2] M. W. Grieves, "Product lifecycle management: The new paradigm for enterprises," *Int. J. Product Development*, vol. 2, no. 1–2, pp. 71–84, 2005.
- [3] Y. H. Kim, C. M. Kim, Y. H. Han, Y. S. Jeong, and D. S. Park, "An efficient strategy of nonuniform sensor deployment in cyber physical systems," *The J. Supercomputing*, vol. 66, no. 1, pp. 70–80, 2013.
- [4] E. J. Tuegel, A. R. Ingraffea, T. G. Eason, and S. M. Spottswood, "Reengineering aircraft structural life prediction using a digital twin," *Int. J. Aerospace Engineering*, vol. 2011, p. 154798, 2011.
- [5] E. Glaessgen and D. Stargel, "The digital twin paradigm for future NASA and U.S. Air Force vehicles," in *Proc. 53rd AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conf.*, 2012, pp. 1818–1832.
- [6] R. Soni, M. Bhatia, and T. Singh, "Digital twin: Intersection of mind and machine," *Int. J. Computational Intelligence and IoT*, vol. 2, no. 3, pp. 667–670, 2019.
- [7] S. Haag and R. Anderl, "Digital twin-proof of concept," *Manufacturing Letters*, pp. 64–66, 2018.
- [8] P. D. U. Coronado, R. Lynn, W. Louhichi, M. Parto, E. Wescoat, and T. Kurfess, "Part data integration in the shop floor digital twin: Mobile and cloud technologies to enable a manufacturing execution system," *J. Manufacturing Systems*, vol. 48, pp. 25–33, 2018.
- [9] Y. Fang, C. Peng, P. Lou, Z. Zhou, J. Hu, and J. Yan, "Digital-twin based job shop scheduling towards smart manufacturing," *IEEE Trans. Industrial Informatics*, vol. 15, no. 2, pp. 6425–6435, 2019.
- [10] I. M. Cavalcante, E. M. Frazzon, F. A. Forcellini, and D. Ivanov, "A supervised machine learning approach to data-driven simulation of resilient supplier selection in digital manufacturing," *Int. J. Information Management*, vol. 49, pp. 86–97, 2019.

- [11] A. El Saddik, "Digital twins: The convergence of multimedia technologies," *IEEE MultiMedia*, vol. 25, no. 2, pp. 87–92, 2018.
- [12] D. Smith and S. Singh, "Approaches to multisensor data fusion in target tracking: A survey," *IEEE Trans. Knowledge and Data Engineering*, vol. 18, no. 12, pp. 1696–1710, 2006.
- [13] Y. He, J. Guo, and X. Zheng, "From surveillance to digital twin: Challenges and recent advances of signal processing for industrial internet of things," *IEEE Signal Processing Magazine*, vol. 35, no. 5, pp. 120–129, 2018.
- [14] B. Yi, X. Li, and Y. Yang, "Heterogeneous model integration of complex mechanical parts based on semantic feature fusion," *Engineering Computers*, vol. 33, no. 4, pp. 797–805, 2017.
- [15] G. N. Schroeder, C. Steinmetz, C. E. Pereira, and D. B. Espindola, "Digital twin data modeling with AutomationML and a communication methodology for data exchange," *IFAC-PapersOnLine*, vol. 49, no. 30, pp. 12–17, 2016.
- [16] B. Schleich, N. Anwer, L. Mathieu, and S. Wartzack, "Shaping the digital twin for design and production engineering," *CIRP Annals*, vol. 66, no. 1, pp. 141–144, 2017.
- [17] F. Tao, H. Zhang, A. Liu, and A. Y. C. Nee, "Digital twin in industry: State-of-the-art," *IEEE Trans. Industrial Informatics*, vol. 15, no. 4, pp. 2405–2415, 2019.
- [18] K. Zhang, T. Qu, D. Zhou, H. Jiang, Y. Lin, P. Li, H. Guo, Y. Liu, C. Li, and G. Huang, "Digital twin-based opti-state control method for a synchronized production operation system," *Robotics and Computer-Integrated Manufacturing*, vol. 63, p. 101892, 2020.
- [19] L. Chang, L. Zhang, C. Fu, and Y. Chen, "Transparent digital twin for output control using belief rule base," *IEEE Trans. Cyber.*, vol. 52, no. 10, pp. 10364–10378, 2022.
- [20] A. McClellan, J. Lorenzetti, M. Pavone, and C. Farhat, "A physics-based digital twin for model predictive control of autonomous unmanned aerial vehicle landing," *Philosophical Trans. the Royal Society A-Mathematical Physical and Engineering Sciences*, vol. 380, p. 2229, 2022.
- [21] X. Zhu and Y. Ji, "A digital twin-driven method for online quality control in process industry," *Int. J. Advanced Manufacturing Technology*, vol. 119, no. 5–6, pp. 3045–3064, 2022.
- [22] Z. Lei, H. Zhou, W. Hu, G. Liu, S. Guan, and X. Feng, "Towards a web-based digital twin thermal power plant," *IEEE Trans. Industrial Informatics*, vol. 18, no. 3, pp. 1716–1725, 2022.
- [23] G. Liu and S. Daley, "Optimal-tuning PID controller design in the frequency domain with application to a rotary hydraulic system," *Control Engineering Practice*, vol. 7, pp. 821–830, 1999.
- [24] G. Liu and S. Daley, "Optimal-tuning nonlinear PID control for hydraulic systems," *Control Engineering Practice*, vol. 8, no. 9, pp. 1045–1053, 2000.
- [25] Q. Wei, H. Li, and F. Wang, "Parallel control for continuous-time linear systems: A case study," *IEEE/CAA J. Autom. Sinica*, vol. 7, no. 4, pp. 919–928, 2020.
- [26] J. Yang, X. Wang, and Y. Zhao, "Parallel manufacturing for industrial metaverses: A new paradigm in smart manufacturing," *IEEE/CAA J. Autom. Sinica*, vol. 9, no. 12, pp. 2063–2070, 2022.
- [27] L. Merkle, A. S. Segura, J. T. Grummel, and M. Lienkamp, "Architecture of a digital twin for enabling digital services for battery systems," in *Proc. IEEE Int. Conf. Industrial Cyber Physical Systems*, 2019, pp. 155–160.
- [28] T. Defraeye, G. Tagliavini, W. Wu, *et al.*, "Digital twins probe into food cooling and biochemical quality changes for reducing losses in refrigerated supply chains," *Resources, Conservation and Recycling*, vol. 149, pp. 778–794, 2019.
- [29] A. Dalstam, M. Engberg, D. Nafors, B. Johansson, and A. Sundblom, "A stepwise implementation of the virtual factory in manufacturing industry," in *Proc. Winter Simulation Conf.*, 2018, pp. 3229–3240.
- [30] F. Tao and M. Zhang, "Digital twin shop-floor: A new shop-floor paradigm towards smart manufacturing," *IEEE Access*, vol. 5, pp. 20418–20427, 2017.
- [31] N. Nikolakis, K. Alexopoulos, E. Xanthakis, and G. Chrysosouris, "The digital twin implementation for linking the virtual representation of human-based production tasks to their physical counterpart in the factory-floor," *Int. J. Computer Integrated Manufacturing*, vol. 32, no. 1, pp. 1–12, 2019.
- [32] K. Ding, F. Chan, X. Zhang, G. Zhou, and F. Zhang, "Defining a digital twin-based cyber-physical production system for autonomous manufacturing in smart shop floors," *Int. J. Production Research*, vol. 57, no. 20, pp. 6315–6334, 2019.
- [33] R. Soderberg, K. Wormeffjord, J. S. Carlson, and L. Lindkvist, "Toward a digital twin for real-time geometry assurance in individualized production," *CIRP Annals*, vol. 66, no. 1, pp. 137–140, 2017.
- [34] Q. Wang, W. Jiao, P. Wang, and Y. Zhang, "Digital twin for human-robot interactive welding and welder behavior analysis," *IEEE/CAA J. Autom. Sinica*, vol. 8, no. 2, pp. 334–343, 2021.
- [35] Y. Wang, Y. Tian, J. Wang, Y. Cao, S. Li, and B. Tian, "Integrated inspection of QoM, QoP, and QoS for AOI industries in metaverses," *IEEE/CAA J. Autom. Sinica*, vol. 9, no. 12, pp. 2071–2078, 2022.
- [36] G. Liu and S. Daley, "Optimal-tuning PID control for industrial systems," *Control Engineering Practice*, vol. 9, no. 11, pp. 1185–1194, 2001.
- [37] G. Liu, *Nonlinear Identification and Control: A Neural Network Approach*, London, UK: Springer-Verlag Ltd., 2001.
- [38] G. Liu, "Tracking control of multi-agent systems using a networked predictive PID tracking scheme," *IEEE/CAA J. Autom. Sinica*, vol. 10, no. 1, pp. 216–225, 2023.
- [39] G. Liu, J. Yang, and J. Whidborne, *Multiobjective Optimisation and Control*, Baldock, UK: Research Studies Press Ltd., 2003.
- [40] Z. Wang, P. Li, R. Lu, and C. Wang, *Performance Analysis of Multiple Time-Delay Discrete Systems*, Beijing, China: Science and Technology Press, 2022.
- [41] G. Liu, "Coordination of networked nonlinear multi-agents using a high-order fully actuated predictive control strategy," *IEEE/CAA J. Autom. Sinica*, vol. 9, no. 4, pp. 615–623, 2022.



Guo-Ping Liu (Fellow, IEEE) received the B.Eng. in industrial automation and M.Eng. degrees in control engineering from the Central South University of Technology in 1982 and 1985, respectively, and the Ph.D. degree in control systems from the University of Manchester, UK in 1992. He is a Professor with the Southern University of Science and Technology. His research interests include networked multi-agent control, nonlinear identification and intelligent control, multi-objective optimal control, and industrial advanced control applications.

Professor Liu has authored and co-authored 10 books and over 400 journal papers on control systems. He was the General Chair of the 2007 IEEE International Conference on Networking, Sensing and Control, the 2011 International Conference on Intelligent Control and Information Processing, and the 2012 UKACC International Conference on Control. He served as an Editor-in-Chief of the International Journal of Automation and Computing in 2004–2021. He is a Member of the Academy of Europe, a Fellow of IET and a Fellow of CAA.