SPS2020
K. Säfsten and F. Elgh (Eds.)

doi:10.3233/ATDE200196

A Flexible 4D Printing Service Platform for Smart Manufacturing

Qinglei JI[a,b], Chun ZHAO[a,c], Mo CHEN[a], Xi Vincent WANG[a], Lei FENG[b] and Lihui WANG[a,1]

[a] *Department of Production Engineering, KTH Royal Institute of Technology, 100 44 Stockholm, Sweden*

[b] *Department of Machine Design, KTH Royal Institute of Technology, 100 44 Stockholm, Sweden*

[c] *Computer School, Beijing Information Science and Technology University, Beijing 100101, China*

Abstract. With the extensive application of 3D printing (3DP) in smart manufacturing, 4D printing (4DP), which enhances 3D printed objects with shape morphing ability by using smart materials, has shown significant industrial potential and attracted tremendous attention. One key concern of 4DP is how to effectively and quickly meet different production and application requirements considering the complexity of materials and diversity of stimulus methods. In order to provide a general research platform for 4DP researchers, a flexible 4DP service platform is proposed. Components and modules for building 4DP and test systems are modeled and virtualized to form the different resources. These resources are then integrated virtually or physically to provide some basic functions such as a 3D displacement stage or a visual monitoring system. According to different 4DP requirements, these functions are then encapsulated into services to serve different research. The platform enables a variety of 4DP applications in smart manufacturing environments such as 4D printed magnetic medical robots, test platform for studying the 4DP response, etc. A case study on designing a ferromagnetic 4DP platform based on the service platform is performed to prove the feasibility of the method.

Keywords. 4D printing, Service platform, Smart manufacturing

Introduction

Additive manufacturing, also named as 3D printing (3DP), has been catching attention of both academia and industry for many years [1–4]. In contrast to subtractive manufacturing, 3DP is attractive due to characteristics such as rapid prototyping, higher degree of design options, cost effectiveness, etc [5]. However, wider application of 3DP is hindered because of the lower mechanical properties of the 3D printed prototypes and the higher cost in the case of large-scale mass production. As a result, most of the value of 3DP still just shows in research laboratories.

Recently, 4D printing (4DP), considered as one of the most important branches of 3DP, is receiving more and more attentions. The terminology of 4DP was first named by Skylar Tibbits in a TED talk in 2013 [6, 7]. The fourth dimension of 4DP is time, which

[1] Corresponding Author. lihuiw@kth.se

means that 4D printed objects can change shape or other properties under external stimulus. To realize this function, smart materials, with the ability of varying their properties under certain stimulus, are used in 4DP [8, 9]. 4DP can be activated by water, light, temperature, electric and magnetic fields, etc.

4DP technology has been realized by refitting 3DP platforms [10]. 3DP technologies such as Fused Deposition Modeling (FDM) [11], Direct Ink Writing (DIW) [12–14], Stereolithography (SLA) [15], Digital Light Processing (DLP) [16] and Polyjet (PJ) [17] are most frequently used for 4DP research. For example, in a DIW based magnetic field activated 4DP study [12], the print head of a DIW 3D printer is replaced with a special nozzle capable of generating variational magnetic fields that can stimulate the magnetic particles in the light curing or thermal curing resin passing through the nozzle. The directions of the magnetic particles are programmed with different directions as the print nozzle is moved to different positions. After extruded from the nozzle head, the resin is cured, and the directions of the magnetic particles are fixed. Thus, the printed samples will show a programmed magnetization in different parts of the sample from the macro view.

However, refitting 3DP platforms for 4DP usage is challenging. First, the changing of the print head is restricted by the original distributions of the 3D printer elements such as the driving motor, the building plate, the threaded rods, etc. Secondly, the synergy of the displacement stage motion, the extrusion rate and the stimulus features are challenging. Considering that most of the commercial 3D printers are closed source to the customers, researchers must refit the platform with very limited available interfaces. This causes extra effort in acquiring the motion data which then result in extra cost and delay between the collaboration of different mechanical systems.

Existing open source 3DP platforms are mostly FDM printers. These printers normally originate from a community project called RepRap. The project aims at enabling open source self-replicating 3D printers. Nevertheless, the project was initiated for 3DP. Both hardware and software are designed and optimized for 3DP. It takes much time and effort to revise these 3DP elements for 4DP usage. Furthermore, as previously mentioned, 4DP can be realized with different principles. So, it is very inefficient and wasteful of resources to refit for each printing principle. Existing 4DP solutions normally refit one certain 3D printer for their specific usage, which is not standardized and not appropriate for wider spread of 4DP.

As a result, it is vital to propose a solution that is specifically posed for 4DP and to fit requirements of different 4DP principles. In this paper, we introduce the idea of a Flexible 4D Printing Service Platform (4DP-SP) for Smart Manufacturing based on service platform (SP) concepts. SP can provide an environment in which services are registered, published, and shared [18–20]. The important capability of the SP is to extract the functions of resources and encapsulate one or more functions into services. Then composite services to meet more applications. For example, the cloud manufacturing platform servitization encapsulates manufacturing resources into services and publishes services in the cloud, so that more users can use manufacturing resources remotely [21]. Within the SP, the service has a uniform description method and invocation method. Any related resources can be accessed to the SP through registration. Therefore, the SP can not only enable resource sharing, but also establish unified standards.

The SP in this paper, on the one hand, enables 4DP resources and data to be shared, and establishes a standard experimental environment. Researchers involved 4DP can conduct related research in a standard environment. The 4DP-SP enables quick and effective build-up of a 4DP and testing platform. It also facilitates adding or removing functions for the platform. As the platform is built by integrating independent resources, the communications between the devices of the platform is faster and more stable compared with conventional 3DP platforms which as a result enhances the collaboration between the elements of 4DP. The structure of this paper is organized as follows. The conceptual model of 4DP system is proposed in Section I. The architecture of the 4DP-SP and the features of each layer is presented in Section II. Section III introduces a case study on building up a ferromagnetic 4DP platform based on the 4DP-SP. Section IV concludes the paper.

1. Conceptual model of the 4DP-SP

Conceptual model of the 4DP-SP includes resource, platform and application. Resource includes linear stages, material dispensers, control units that can support 3DP, and some stimulus sources of sound, light, electricity, magnetism and temperature unit. In the 4DP-SP, all kinds of resource models are built for physical resources. Virtual resources are built by driving the model and using the resource data. Through these virtual resources, the SP can monitor and control the physical resources related to 4DP. On this basis, the SP further encapsulates the virtual resources to form a cloud service that can be invoked. Through the flexible composition of cloud services, the SP can support a variety of applications related to 4DP, such as multi-stimulus control, multi-material shape change, self-deployable systems and so on. Furthermore, it is also possible to use the components in the 4DP-SP to build a 4DP testing platform, including the stimulus source for the 4D printed samples and the monitoring unit which can be camera or other sensor sources.

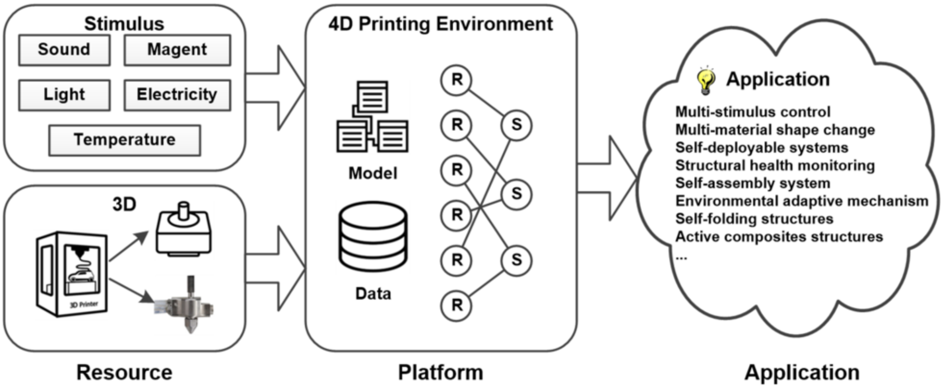


Figure 1. Conceptual model of the 4DP system.

2. Architecture of the 4DP-SP

The SP is the core of the 4DP-SP. The SP encapsulates the resource to services and supports various 4DP applications. As shown in Figure 2, the physical resource layer is at the bottom of the architecture. The physical resource is the basis of the 4DP system and provides various equipment support related to 4DP. Since 4DP is based on 3DP, physical resources include various resources for 3DP, such as linear stage, material dispenser, etc. In addition, many stimulus and sensor resources are added, such as temperature sensors or pressure sensors. Furthermore, stimulus units including temperature, light, sound, pressure, electricity, magnetic units are also supplemented for 4DP functions. On the other hand, physical resources have basic characteristics which are inputs and outputs. Input devices refer to various sensors, such as position, temperature, light, sound, etc. The output device refers to various controllers, such as linear stage, dispenser, temperature controller, electromagnet, etc.

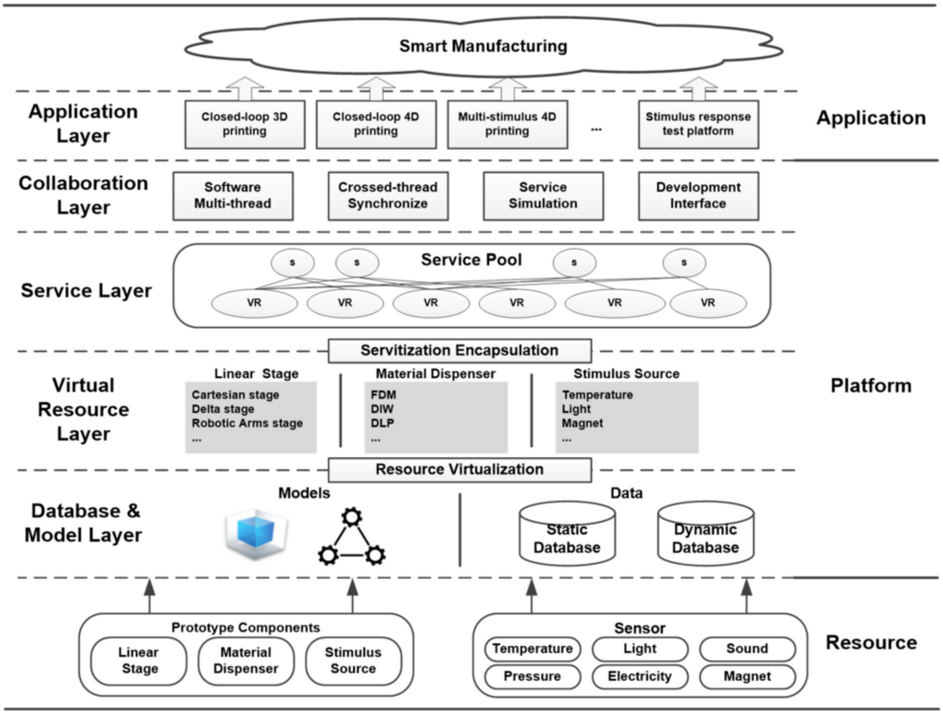


Figure 2. Architecture of the 4DP system.

2.1. Database and model layer

The database and model layer are the lowest layer of the SP, enabling physical resources to access the SP. This layer is divided into two parts including model and database. The modeling for physical resources is to establish a framework and enable physical resources to be digitally described. Through this component model, the SP can meet the output and control of physical resources.

The database is divided into static data and dynamic data. Static data stores static parameter data of resources. Dynamic data stores real-time data of resources. Different from the traditional experimental platform, the platform in this paper collects data in real-time through input resources such as sensors and stores the data with timestamps to different libraries. Dynamic and static data will provide data support throughout the system. In the virtual resource layer, the database provides real-time data to drive the model to form an entity, thereby realizing the monitoring of physical resources. At the service layer, the database provides real-time data to monitor the service states. At the application layer, the database provides data to enable real-time data sampling of the 4DP process. For example, in a closed-loop control 3DP process, the user does not need to consider the data collection scheme as long as the relevant time data is taken from the database.

2.2. Virtual resource layer

The virtual resource layer can run the component model to form an entity and build virtual resources in the cyber environment through real-time data. A complete virtual resource consists of several component models and dynamic data. These models provide the overall framework and some output functions of the virtual resource. Real-time data come from databases of sensors, and the sensor data are stored in the database with timestamps. Virtual resources can enable monitoring and control of physical resources.

As shown in Figure 3, a component model includes a framework and control functions. In the framework, the static description module is used to describe the basic information of the resource, such as the device name, product model, basic parameters and so on. The control type indicates the communication mode of the device and is used to send device control instructions. The dynamic data is mapped to the corresponding database, and the dynamic data can be obtained in real-time during the operation of the virtual resource to realize the device monitoring, and the dynamic data is also used to the feedback of device control. Inputs are the control interface of the virtual resource. The control instructions sent by the platform are analyzed and can perform the functions of the resource.

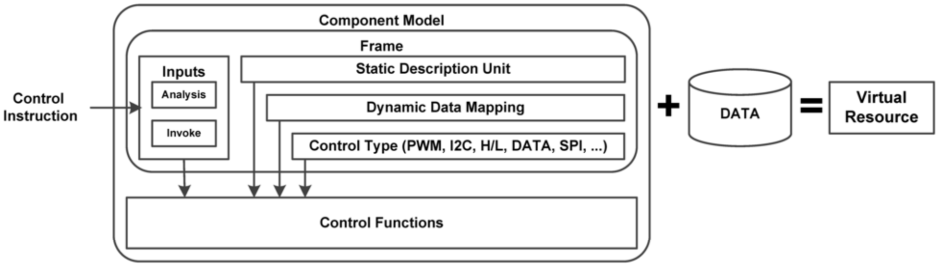


Figure 3. The relationship between models, data and virtual resources.

2.3. Service layer

Virtual resources are encapsulated into services to form services and then stored in the service pool at the service layer, as shown in Figure 4. The service layer provides a servitization method and service center. The service is registered in the service center of the service layer and can be invoked and constructed in the Collaboration layer. The service description method is basically the same as the normal cloud platform. The service interface uses the standard web service method: Simple Object Access Protocol (SOAP) which is based on Extensible Markup Language (XML) messages. The advantage of this method is that it facilitates system integration and can be invoked across platforms. The service management module can parse the XML message and convert the message into a control instruction and send it to the virtual resource. Each service carries a thread, so each service runs independently in a subsystem. During service running, the database provides static and dynamic data support. Clock control is an important characteristic of this platform. In 4DP, both the printing process and the actuation process require high synchronization. In addition, there are a variety of stimulation devices, and each device has a different response time. Therefore, each service needs an internal clock, and the platform needs a global clock to achieve service synchronization.

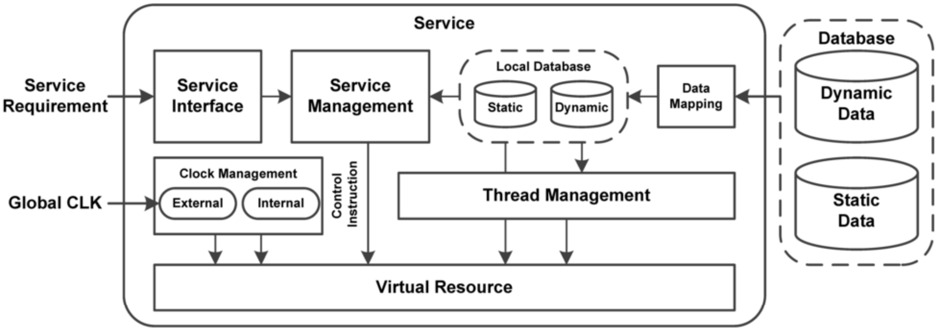


Figure 4. 4DP service.

2.4. Collaboration layer

The Collaboration layer provides many modules to support the running and composition of services, such as running of software multi-thread, synchronization of crossed-thread, service simulation, second development interface, etc.

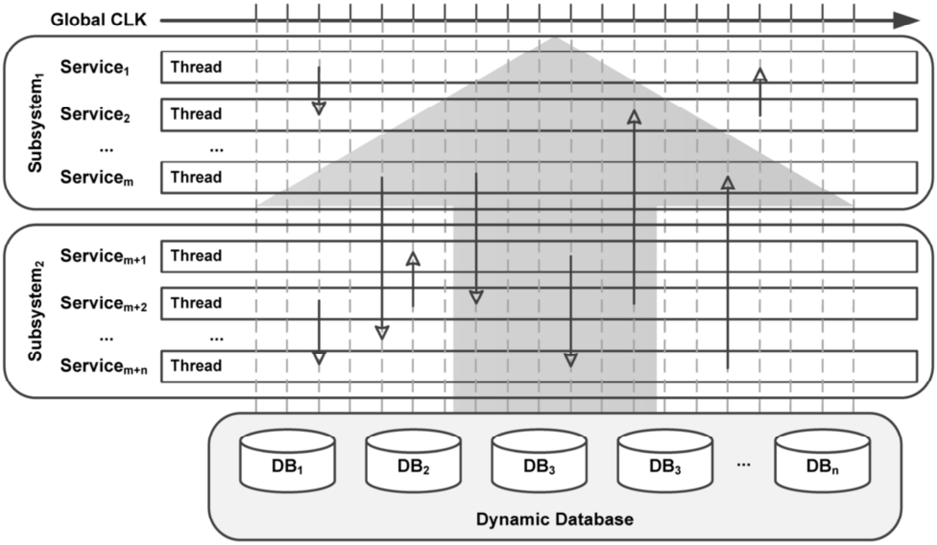


Figure 5. Collaboration of services.

As shown in Figure 5, Each service carries a thread that runs independently and in parallel. In a subsystem, multiple threads are established according to the capabilities of the subsystem, and the subsystem performs different services. Each service runs on its internal clock, so response times are different. Services communicate with each other through messages to get the status of different resources or control other resources. In addition, the platform provides a global clock to provide a basic clock for each service in order to enable different services to synchronize. During the running of multiple services, dynamic data provides background support, and each service can directly obtain data from the database. With the support of the platform, many 4DP applications can be executed, such as closed-loop 3DP, multi-stimulus 4DP, stimulus-response test, etc.

3. Case study

A ferromagnetic 4DP platform that can fabricate objects with programmed magnetic fields is established to prove the feasibility of the 4DP-SP. As shown in Figure 6, a pressure source, a syringe containing the resin mixed with magnetic particles and a magnetic programmer that can generate specific magnetic fields on the syringe tip form the print head unit. The print head is then mounted on a 3D displacement gantry and the extrusion and magnetic programming will operate according to the gantry position under the control of a controller board. During 4DP process, the 3D displacement gantry moves in the space and the pressure source is activated to extrude the composite material. During extrusion, the disordered magnetic particles pass through the magnetic programmer that generates magnetic fields. The movement of the gantry, the extrusion rate and the programming direction are all synchronized with a global clock in the 4DP-SP mentioned previously. With such synchronization, the 4DP process is well controlled and avoids wrong programing at any part of a printed sample. After extrusion, the material is solidified to fix the magnetic particle direction and form the object.

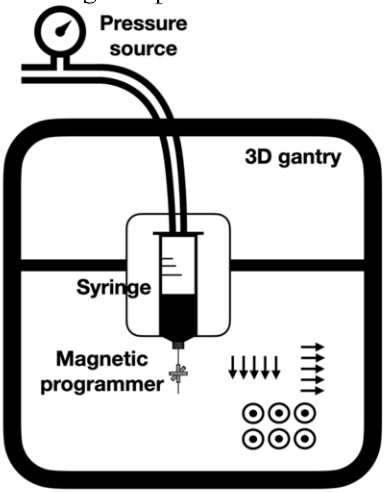


Figure 6. Schematic of ferromagnetic 4DP platform.

3.1. Modeling of devices

The first step to build a platform for this case is to model the various devices involved in this case, including the pressure source, the magnetic programmer, and the 3D displacement gantry. The pressure source is composed of a fixed air pressure supply and a pressure regulator. The pressure regulator takes in the pressure supply and changes its pressure output according to an electric voltage level. The model of the pressure source can be expressed as Equation (1):

$$M_{pressure} = \langle x_{inputvol}, x_{Goal}, y_{ouputpres}, f_{control}, M_d, I_d \rangle \tag{1}$$

Where, $x_{inputvol}$, x_{Goal} and $y_{ouputpres}$ are timed signals. Let $T = [0, \infty)$ be the time domain. $x_{inputvol}: T \rightarrow \mathbb{R}$ is the electric voltage input which controls the air pressure. $x_{Goal}: T \rightarrow \mathbb{R}$ is the goal pressure. $y_{ouputpres}: T \rightarrow \mathbb{R}$ is the real-time output pressure. $f_{control}: \mathbb{R}^4 \rightarrow \mathbb{R}$ is a Proportional Integral Derivative (PID) control function for regulating the pressure $y_{ouputpres}$ to quickly follow the reference input x_{Goal}. $M_d = \{m_1, m_2, \ldots\}$ is the set of component systems which includes sensors and solenoid valve. $I_d = \{i_1, i_2, \ldots\}$ is the set of interface maps which can send data to the database. The error signal is defined as $err(t) = x_{Goal}(t) - y_{ouputpres}(t)$. The discrete-time implementation of the control function $f_{control}$ is defined in Equation (2)[23]:

$$\begin{aligned} x_{inputvol}^{n} &= f_{control}\left(x_{inputvol}^{n-1}, err^n, err^{n-1}, err^{n-2}\right) \\ &= x_{inputvol}^{n-1} + K_p \times (err^n - err^{n-1}) + K_i \times err^n + \\ &\qquad K_d \times (err^n - 2 \times err^{n-1} + err^{n-2}) \end{aligned} \tag{2}$$

During 4DP process, the pressure can change according to the magnetic programmer. So internal interfaces between the magnetic programmer and the pressure source are required. In addition, the magnetic programmer and the 3D displacement gantry, the pressure source and the 3D displacement gantry all need internal interfaces. The advantage of the 4DP-SP is that it includes a shared database that store the data of various resources. For example, the pressure source can acquire the state data of the magnetic programmer directly.

The model of magnetic programmer M_{magnet} defined within $T = [0, \infty)$ is shown in Equation (3) where $x_{inputvol}: T \rightarrow \mathbb{R}$ is the input voltage to control the magnet.

$$M_{magnet} = \langle x_{inputvol}, M_d, I_d \rangle \tag{3}$$

Through varying the input voltage, the output magnetic field intensity of the magnetic programmer will also change. $M_d = \{m_1, m_2, \ldots\}$ is the set of component systems. $I_d = \{i_1, i_2, \ldots\}$ is the set of interface maps which can send data to the database.

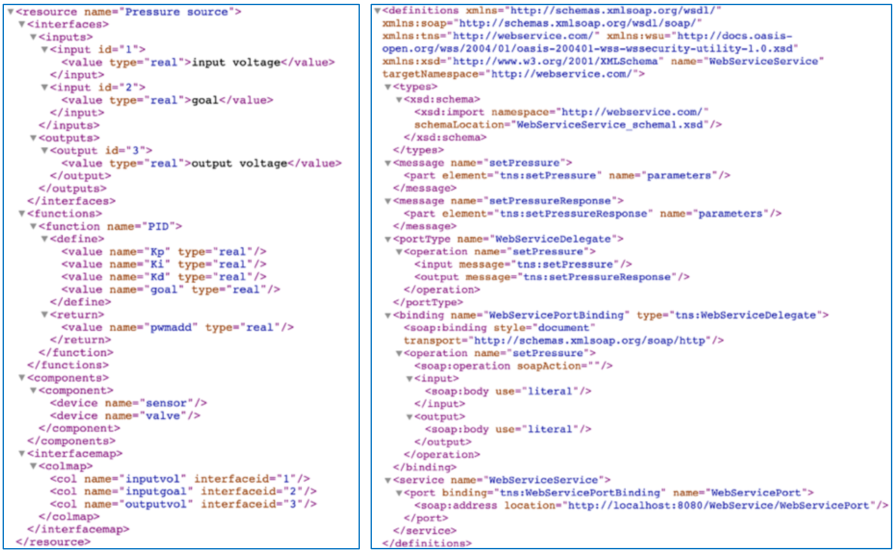
The model of the 3D displacement gantry M_{gantry} is defined in Equation (4):

$$M_{gantry} = \langle x_{input}, y_{input}, z_{input}, P_d, M_d, I_d \rangle \tag{4}$$

$x_{input}, y_{input}, z_{input}$ are the position references sent to the gantry system and are all defined as $T \rightarrow \mathbb{R}$. $M_d = \{m_1, m_2, \ldots\}$ is the set of component systems including 3 stepper motors. $P_d = \{p_1, p_2, \ldots\}$ is the set of position. $I_d = \{i_1, i_2, \ldots\}$ is the set of interface maps which can send data to the database.

3.2. Virtualization of resource

Using the models defined in Section 3.1, 4DP-SP can build and run the system. The entity integrates the data from the physical resource to form a virtual resource in 4DP-SP, monitor and control the physical resource.



```
▼<resource name="Pressure source">
  ▼<interfaces>
    ▼<inputs>
      ▼<input id="1">
          <value type="real">input voltage</value>
        </input>
      ▼<input id="2">
          <value type="real">goal</value>
        </input>
      </inputs>
    ▼<outputs>
      ▼<output id="3">
          <value type="real">output voltage</value>
        </output>
      </outputs>
    </interfaces>
  ▼<functions>
    ▼<function name="PID">
      ▼<define>
          <value name="Kp" type="real"/>
          <value name="Ki" type="real"/>
          <value name="Kd" type="real"/>
          <value name="goal" type="real"/>
        </define>
      ▼<return>
          <value name="pwmadd" type="real"/>
        </return>
      </function>
    </functions>
  ▼<components>
    ▼<component>
        <device name="sensor"/>
        <device name="valve"/>
      </component>
    </components>
  ▼<interfacemap>
    ▼<colmap>
        <col name="inputvol" interfaceid="1"/>
        <col name="inputgoal" interfaceid="2"/>
        <col name="outputvol" interfaceid="3"/>
      </colmap>
    </interfacemap>
</resource>
```

(a) Virtualization of resources

```
▼<definitions xmlns="http://schemas.xmlsoap.org/wsdl/"
 xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
 xmlns:tns="http://webservice.com/" xmlns:wsu="http://docs.oasis-
 open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd"
 xmlns:xsd="http://www.w3.org/2001/XMLSchema" name="WebServiceService"
 targetNamespace="http://webservice.com/">
  ▼<types>
    ▼<xsd:schema>
        <xsd:import namespace="http://webservice.com/"
        schemaLocation="WebServiceService_schema1.xsd"/>
      </xsd:schema>
    </types>
  ▼<message name="setPressure">
      <part element="tns:setPressure" name="parameters"/>
    </message>
  ▼<message name="setPressureResponse">
      <part element="tns:setPressureResponse" name="parameters"/>
    </message>
  ▼<portType name="WebServiceDelegate">
    ▼<operation name="setPressure">
        <input message="tns:setPressure"/>
        <output message="tns:setPressureResponse"/>
      </operation>
    </portType>
  ▼<binding name="WebServicePortBinding" type="tns:WebServiceDelegate">
      <soap:binding style="document"
      transport="http://schemas.xmlsoap.org/soap/http"/>
    ▼<operation name="setPressure">
        <soap:operation soapAction=""/>
      ▼<input>
          <soap:body use="literal"/>
        </input>
      ▼<output>
          <soap:body use="literal"/>
        </output>
      </operation>
    </binding>
  ▼<service name="WebServiceService">
    ▼<port binding="tns:WebServicePortBinding" name="WebServicePort">
        <soap:address location="http://localhost:8080/WebService/WebServicePort"/>
      </port>
    </service>
</definitions>
```

(b) Servitization of virtualized resources

Figure 7. Virtualization and servitization of the resources.

The model-based virtual resource is shown in Figure 7(a). The virtual resource is based on the resource model and shares data, interfaces and functions. In the 4DP-SP, any user can obtain the state of the physical resource through the virtual resource and control the physical resource through the operation of the virtual resource. In addition, the virtual resource provides support for the servitization of function.

3.3. Servitization of virtualized resource

The servitization method of virtual resources is based on the Web Services Description Language (WSDL) standard which extracts the functions of virtual resources, and deploys them to the 4DP-SP.

Figure 7(b) shows the servitization description of virtualized resource. For example, the pressure setting function of virtual resources is encapsulated as a service for the pressure source. The platform user can set the pressure by calling the service of *setPressure* and obtain the pressure return data by calling the service of *setPressureResponse*. Similarly, various functions of virtual resources are encapsulated as services in 4DP-SP. The platform can enable users to compose services and can meet various 4D printing requirements.

4. Conclusion

This paper proposes a 4DP-SP that aims at overcoming the complexity and poor expandability of current 4DP platforms developed from 3DP platforms. Resources that contribute to 4DP are extracted and classified into the resource layer. To monitor and control these resources in further steps, they are then modeled and their data are collected for the 4DP-SP. Through running these models, a virtual resource is established. The virtual resource is a digital twin of the physical resource, which can be easily monitored and controlled by unified internal interface. Different virtual resources are then encapsulated as services. Through invocation and composition of different services, multiple 4DP applications can be realized.

Through the 4DP-SP, researchers can avoid complex refitting process of 3D printers. Different 4DP functions can be realized by simply combining different services provided by the platform. It is also easier to add functions on a basic 4DP-SP so that a 4DP platform with more functions can be realized. Further, through design the 4DP platform in a system level, the collaboration between the different elements are more convenient and robust. The contribution of this study is concluded as follows:

- A new method named 4DP-SP for building up 4DP platforms is introduced which improves the flexibility and extendibility of 4DP platforms.
- Conceptual framework of the 4DP-SP is investigated and created which supports researchers developing such platforms.
- A preliminary design of a ferromagnetic 4DP platform based on 4DP-SP is performed which proves the feasibility of the 4DP-SP.

In the future, real implementation of the 4DP-SP should be performed. A ferromagnetic 4DP platform based on the design in the case study will be built. Tests of the samples fabricated with the platform will be performed and results will be compared with the fabrication process of the traditional 4DP platforms. Cloud-based monitoring and control of 4DP can be investigated. Remote control of the 4DP fabrication process and the 4D morphing process will be studied. Furthermore, through encapsulation of more software and 4DP models, more simulations of 4DP can be executed.

Acknowledgement

This research is financially supported by Swedish Research Council (Grant No. 2017-04550) and KTH XPRES.

References

[1] T. D. Ngo, A. Kashani, G. Imbalzano, K. T. Q. Nguyen, and D. Hui, Additive manufacturing (3D printing): A review of materials, methods, applications and challenges, *Composites Part B: Engineering*, vol. 143, 2018, pp. 172–196.

[2] Q. Ji *et al.*, A Modular Microfluidic Device via Multimaterial 3D Printing for Emulsion Generation, *Scientific Reports*, vol. 8, 2018, pp. 4791.

[3] J. M. Zhang, Q. Ji, Y. Liu, J. Huang, and H. Duan, An integrated micro-millifluidic processing system, *Lab on a Chip*, vol. 18, 2018, pp. 3393–3404.

[4] J. M. Zhang and Q. Ji, Three-Dimensional Printed Devices in Droplet Microfluidics, *micromachines*, vol. 10, 2019, pp. 1–24.

[5] S. H. Huang, P. Liu, A. Mokasdar, and L. Hou, Additive manufacturing and its societal impact: A literature review, *International Journal of Advanced Manufacturing Technology*, vol. 67, 2013, pp. 1191–1203.

[6] J. Choi, O.-C. Kwon, W. Jo, H. J. Lee, and M.-W. Moon, 4D printing technology: A review, *3D Printing and Additive Manufacturing*, vol. 2, 2015, pp. 159–167.

[7] X. Kuang *et al.*, Advances in 4D Printing: Materials and Applications, *Advanced Functional Materials*, vol. 29, 2019, pp. 1805290.

[8] Z. X. Khoo *et al.*, 3D printing of smart materials: A review on recent progresses in 4D printing, *Virtual and Physical Prototyping*, vol. 10, 2015, pp. 103–122.

[9] R. L. Truby and J. A. Lewis, Printing soft matter in three dimensions, *Nature*, vol. 540, 2016, pp.

371.
[10] F. Momeni, X. Liu, and J. Ni, A review of 4D printing, *Materials & Design*, vol. 122, 2017, pp. 42–79.
[11] Q. Zhang, K. Zhang, and G. Hu, Smart three-dimensional lightweight structure triggered from a thin composite sheet via 3D printing technique, *Scientific Reports*, vol. 6, 2016, pp. 22431.
[12] Y. Kim, H. Yuk, R. Zhao, S. A. Chester, and X. Zhao, Printing ferromagnetic domains for untethered fast-transforming soft materials, *Nature*, vol. 558, 2018, pp. 274–279.
[13] Y. Kim, G. A. Parada, S. Liu, and X. Zhao, Ferromagnetic soft continuum robots, *Science Robotics*, vol. 4, 2019, pp. eaax7329.
[14] A. Sydney Gladman, E. A. Matsumoto, R. G. Nuzzo, L. Mahadevan, and J. A. Lewis, Biomimetic 4D printing, *Nature Materials*, vol. 15, 2016, pp. 413–418.
[15] M. Zarek, M. Layani, I. Cooperstein, E. Sachyani, D. Cohn, and S. Magdassi, 3D printing of shape memory polymers for flexible electronic devices, *Advanced Materials*, vol. 28, 2016, pp. 4449–4454.
[16] J. Wu, Z. Zhao, X. Kuang, C. M. Hamel, D. Fang, and H. J. Qi, Reversible shape change structures by grayscale pattern 4D printing, *Multifunctional Materials*, vol. 1, 2018, pp. 15002.
[17] D. Raviv *et al.*, Active printed materials for complex self-evolving deformations, *Scientific Reports*, vol. 4, 2014, pp. 7422.
[18] X. Xu, From cloud computing to cloud manufacturing, *Robotics and Computer-Integrated Manufacturing*, vol. 28, 2012, pp. 75–86.
[19] Y. Liu, L. Wang, and X. V. Wang, Cloud manufacturing: latest advancements and future trends, *Procedia Manufacturing*, vol. 25, 2018, pp. 62–73.
[20] Y. Liu, L. Wang, X. V. Wang, X. Xu, and L. Zhang, Scheduling in cloud manufacturing: state-of-the-art and research challenges, *International Journal of Production Research*, vol. 57, 2019, pp. 4854–4879.
[21] Z. Wansheng *et al.*, A New Model of WEDM-CNC System with Digitizer/Player Architecture, *Procedia CIRP*, vol. 42, 2016, pp. 210–214.
[22] D. Wu, D. W. Rosen, L. Wang, and D. Schaefer, Cloud-based design and manufacturing: A new paradigm in digital manufacturing and design innovation, *Computer-Aided Design*, vol. 59, 2015, pp. 1–14.
[23] C. F. J. Kuo and H. M. Tu, An Entire Strategy for Control of a Calender Roller System. Part III: Intelligent Settling Time-optimal Control, *Textile research journal*, vol. 78, 2008, pp. 81–87.