# A Digital Twin Framework for Performance Monitoring and Anomaly Detection in Fused Deposition Modeling

Efe C. Balta, Dawn M. Tilbury, Kira Barton

***Abstract*— Digital twin (DT) and additive manufacturing (AM) technologies are key enablers for smart manufacturing systems. DTs of AM systems are proposed in recent literature to provide additional analysis and monitoring capabilities to the physical AM processes. This work proposes a DT framework for real-time performance monitoring and anomaly detection in fused deposition modeling (FDM) AM process. The proposed DT framework can accommodate AM process measurement data to model the AM process as a cyber-physical system with continuous and discrete event dynamics, and allow for the development of various applications. A new performance metric is proposed for performance monitoring and a formal specification based anomaly detection method is proposed for AM processes. Implementation of the proposed DT on an off-the-shelf FDM printer and experimental results of anomaly detection and process monitoring are presented at the end.**

## I. Introduction

Additive manufacturing (AM) is an important enabler of smart manufacturing due to its ability to produce customized products with complex geometries [1]. AM has been increasingly utilized in the industry over the past years. Even though AM has seen increased use in industrial applications, key questions about monitoring AM systems, collecting data, and system-level control of multiple AM machines in a manufacturing system (AM Fleets) remain important open questions [2]. Fused deposition modeling (FDM) is an AM process in which a thermoplastic material is extruded through a heated nozzle in a numerically controlled deposition system in a bottom-up layer-by-layer fashion. There has been recent work in modeling FDM processes, developing measurement technologies, and establishing verification techniques to improve the quality and reliability of AM manufactured parts [3]–[5]. Digital twins (DTs) are proposed in recent literature to model the components of AM machines and perform preliminary anomaly detection tasks [6]. Most of the current literature relies on customized sensing and measurement technologies to build DTs of AM systems. There has been little DT work focused on a unified approach to handle different types of data available through the machine, control system, and the design data of an AM process to model the cyber-physical nature of an AM machine for anomaly detection and performance monitoring.

This work presents a DT architecture with appropriate mathematical modeling formalisms and data structures for performance monitoring and anomaly detection of AM processes. Focused examples of the presented approach on FDM technology are presented to illustrate practical use cases of the proposed DT. Experimental results with an off-the-shelf FDM printer illustrates the proposed approach of translating end-product design data into formal specifications to perform anomaly detection and performance monitoring with DTs.

### A. Literature Review and Research Gap

A DT is a software replica of a physical asset or process, which combines modeling information with data analytics to deliver additional monitoring and analysis capabilities to the physical system [7]. In [8] the use of DTs in manufacturing systems to support product design and process optimization is presented. In [9], the importance of DTs in manufacturing systems is motivated and conceptual examples of DTs in smart manufacturing systems are presented. The concept of cyber-physical manufacturing systems and the role of DT based monitoring, simulation, and control is presented in [10]. There has been recent work in developing DTs for smart manufacturing systems for reconfiguration based decision-making [11]. The development in [11] considers the system interactions of multiple machines and does not provide a modeling approach for AM processes. A functional model for AM processes is presented in previous work [2], but details of DT development are not provided. Additionally, a layer-to-layer spatial dynamic model for FDM processes is proposed in previous work [5], but the functional states of the process are not considered and an anomaly detection framework is not provided.

A computational models-based DT for the process dynamics of a directed energy deposition AM process is given in [12]. Although this development has high accuracy with respect to the actual AM process, the computational expense of the provided model make the development infeasible for run-time analysis. In [13], building blocks of a DT for predicting microstructure of the printed parts and estimating the residual stress on parts are presented as a survey. The work in [13] considers many different developments in literature and conclude that a real-time analysis tool is not present in current literature.

In [14], the use of functional models for quality modeling in FDM is proposed. The quality of printed parts are predicted using off-line and in-situ measurements, but their modeling framework requires calibration of model parameters for different AM processes and machines, which may be labor intensive in practice. A DT structure with cloud integration capabilities is proposed, and possible DT

This work was funded in part by NSF 1544678 and NIST Award No.70NANB16H205.

Efe C. Balta, Dawn M. Tilbury, and Kira Barton are with the Department of Mechanical Engineering, University of Michigan, Ann Arbor, MI 48109, USA {baltaefe,tilbury,bartonkl}@umich.edu

applications in FDM printers are given in [15]. The DT in [15] does not model the dynamics of the FDM process or provide details on performance monitoring and anomaly detection. A DT for FDM is given and an anomaly localization scheme is proposed in [6]. The proposed DT in [6] leverages side-channel informations to perform data-driven analysis and localize anomalies in an FDM printed part with promising accuracy, but their development does not leverage a functional model of the FDM machine and requires model training with specially placed sensors on the machine.

In [16], a reference DT architecture for cloud based cyber-physical systems is given and applications on driving assistance systems is presented. Their development focuses on computational and network aspects of DT implementation and does not consider the dynamic models of the underlying physical system for anomaly detection and analysis. In current literature, there does not exist a DT framework for AM processes that can accommodate AM process models with continuous and functional dynamics to leverage AM process data from the physical system in real-time. Therefore, a unified DT architecture for AM processes that can handle multiple data streams to capture the cyber-physical dynamics (continuous and functional) of an AM process and update the dynamic model states in real-time must be devised.

### B. Contributions and Paper Organization

To address the research gap identified in the previous section, three major contributions of this work are the following:

1) Development of a multi-purpose DT for AM systems, with a system model to capture the continuous and functional dynamics of an AM process
2) Identification of a novel performance monitoring metric for energy efficiency of an FDM process and derivation of a formal specification-based anomaly detection approach
3) An experimental study on process monitoring and anomaly detection for an off-the-shelf FDM printer via the proposed DT

The remainder of the paper is structured as follows. Section II presents the preliminaries for an AM process, and illustrates a formal method to specify and monitor specifications on an AM process. Section III presents a DT architecture for AM and discusses possible uses for performance monitoring and anomaly detection. Section IV presents experimental results of anomaly detection and performance monitoring on an off-the-shelf FDM 3D printer. Concluding remarks and future work are given in Section V.

## II. AM Process Preliminaries

This section provides definitions of the AM process, and gives some of the theoretical background on formal specifications for AM process data.

An AM process can be investigated by the constituent temporal and spatial dynamics [5]. Temporal dynamics involve the transient response of the time-dependent process states while spatial dynamics define the space-dependent process states. The dependence on time and space here are defined in a layer-by-layer sense. For example, in the FDM process, geometry of the deposited beads from one layer to another constitute spatial dynamics. Thus, we classify the dynamics of spatial variables from one layer to another as the spatial dynamics and the dynamics of the deposition system within a layer as the temporal dynamics.

### A. Definitions

A *build* is the additively manufactured part in process. A build is processed in a bottom-up layer-by-layer fashion. An *AM process* consists of the build and the AM machine itself.

The *AM-workflow* is the procedure of going from a design geometry to a manufactured part. The steps in the AM-workflow have four levels with the top level as the end-product specifications, followed by the process parameters, a reference list, and the bottom level as the actuator inputs.

### B. Formal Specifications for Process Data

The AM process is a cyber-physical system that has both continuous and functional (discrete-events) dynamics. Thus, a specification for the AM process should account for both discrete events in the system and the evolution of continuous dynamics during the process. In [17], global operational states with continuous and discrete-event dynamics are introduced for process specification and anomaly detection in industrial manufacturing machinery. [17] uses signal templates for finding events in a measured signal, which requires expert knowledge that may not be reusable in different systems.

To define reusable and generalizable formal specifications on the AM process data, the use of a formal specification language is proposed in this work. Signal temporal logic (STL) is a formalism to specify propositions on a signal measured from an underlying system. STL is widely used for describing formal specifications on the output signals (traces) of systems with discrete-event and continuous dynamics, thus it is a suitable choice for formal specifications with an AM process. While the use of STL for formal specifications is not new in literature, the use of STL as a formal method for anomaly detection with AM processes is a novel contribution of this work. An STL formula is formed by the following:

$$\phi \triangleq \quad \top \quad | \quad p \quad | \quad \neg\phi \quad | \quad \phi_i \wedge \phi_j \quad | \quad \phi_i \, \boldsymbol{\mathcal{U}}_{[a,b]} \phi_j$$

where, $\top$ is logical true, $p$ is a predicate, $\neg\phi$ is the logical negation of the proposition $\phi$, $\phi_i \wedge \phi_j$ is the logical conjunction of two propositions, and $\phi_i \, \boldsymbol{\mathcal{U}}_{[a,b]} \phi_j$ is the *until* operator defined as the proposition $\phi_i$ being true at least until the proposition $\phi_j$ is true in the time interval $[t+a, t+b]$, where $t$ is the current time. A signal $\boldsymbol{s}(t)$ at time $t$ is satisfied by a predicate $p$ if $f(\boldsymbol{s}(t)) > 0$ for some function $f$ (i.e. $\boldsymbol{s}(t) \models p \iff f(\boldsymbol{s}(t)) > 0$). Additionally, $\perp = \neg\top$ is the logical false, the *eventually* operator is $\Diamond_{[a,b]}\phi \triangleq \top \, \boldsymbol{\mathcal{U}}_{[a,b]}\phi$, and the *always* operator is $\Box_{[a,b]}\phi \triangleq \neg(\Diamond_{[a,b]}\neg\phi)$. Based on the definition of until, *eventually* operator $\Diamond_{[a,b]}\phi$ describes that the proposition $\phi$ will be true after some time in the given time interval. Similarly, *globally* operator $\Box_{[a,b]}\phi$ describes that the negation of $\phi$ will never be true in the given time interval.

Time intervals of the propositions are used as sliding windows over a given signal, where the satisfaction of a proposition is checked. An extensive analysis and use of STL is given in [18]. Using STL formulas, specifications on the AM process can be expressed in a formal language so that a DT can check these properties. An anomaly detection scheme based on the satisfaction of specifications is formulated in the case study.

A number of formal specifications can be described using STL. Let $\Phi = \{\phi_1, \dots, \phi_{n_s}\}$ denote a finite set of STL formulas defined for a specific AM process. For the propositions in $\Phi$, we investigate if a measured signal $\boldsymbol{s}(t)$ satisfies the propositions. The signal $\boldsymbol{s}(t)$ in this work is defined as data (measurement or state) about the underlying AM process. The satisfaction condition is denoted as the following.

$$\boldsymbol{s}(t) \models \bigwedge_{\forall \phi_j \in \Phi} \phi_j, \tag{1}$$

where we require the signal $\boldsymbol{s}(t)$ to satisfy the conjunction of $n_s$ propositions. If a measured signal $\boldsymbol{s}(t)$ satisfies Eq. (1), we conclude that the underlying AM process satisfies the formal specifications given by the set $\Phi$.

Propositions for an AM process may specify properties of the end-product. Additionally, a proposition may define allowable working conditions of the AM machine or the materials in the process. For example, an AM process proposition $\phi_j$ may define allowable printing temperatures specific to each material in an AM process. To check if the proposition is satisfied, a measurement signal $\boldsymbol{s}(t)$ of the corresponding material temperature may be monitored in addition to other measurement data. An example of such a proposition is given in the case study.

## III. A Digital Twin Architecture for AM

A DT for AM utilizes the parameters from AM-workflow and updates the corresponding process models and parameters in real-time. In this section, an architecture of a DT for AM along with a discussion of how different levels of data are utilized for applications with the DT is presented. Figure 1 illustrates the proposed DT architecture. Three main components of the proposed DT are the *DT Interface*, *DT Functions*, and *DT Core*. The reference list $\boldsymbol{r} = [\boldsymbol{r_1}, \dots, \boldsymbol{r}_{n_f}]$ is the list of reference inputs for the AM plant. An example of $\boldsymbol{r}$ is GCode instructions, used to prescribe actions for the components of a numerically controlled machine, which are commonly used in AM. Each line of a GCode file describes a set of actions for the machine to execute using its actuators. Consequently, $\boldsymbol{r}_j$ denotes a single line of a GCode command executed at time-step $j$.

### A. DT Interface

Real-time data coming from the physical system is managed by the DT Interface. A conceptual AM plant is shown in Fig. 2. An AM plant in this work is considered as a closed-loop controlled AM process in which the only allowable input to the closed-loop AM plant is the reference list $\boldsymbol{r}$ . This consideration is a practical one since most AM machines in

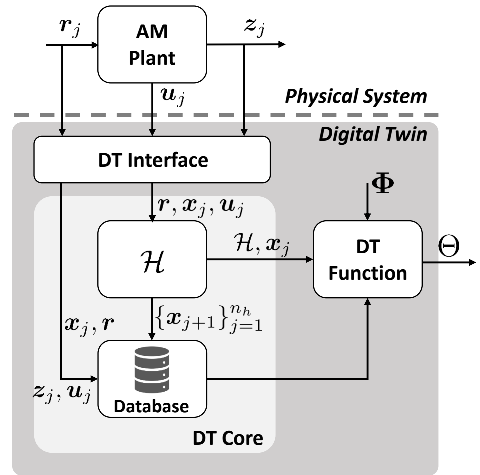


Fig. 1. Structure of the proposed DT

practice have OEM control systems and sensors that are not accessible to the user during the AM process. The reference list input (GCode) is interpreted by an interpreter and a controller input $\hat{\boldsymbol{r}}_j$ is generated for the OEM controller of the AM plant. The OEM controller uses the controller input and the measurements from the AM process coming from the OEM sensors to generate the actuator inputs $\boldsymbol{u}_j$. Based on the actuator inputs, the AM process takes place and a physical part is manufactured. The physical outputs of the AM process is denoted with $\boldsymbol{y}_j$ in Fig. 2. The OEM sensors measure the physical output $\boldsymbol{y}_j$ and send $\boldsymbol{z}_j^{OEM}$ back to the OEM controller. External sensors such as cameras, laser scanners, and temperature readers are often instrumented on an AM machine to measure the physical output $\boldsymbol{y}_j$ [4], [6]. External output measurements are shown with $\boldsymbol{z}_j^{ext}$ in Fig. 2. The output data $\boldsymbol{z}_j = [\boldsymbol{z}_j^{ext}, \boldsymbol{z}_j^{OEM}]^T$ is computed as a vector output of measurements from OEM sensors and external sensors, at the output of the AM Plant. In practice, some of the OEM sensor measurements may be unobservable to the user, in which case $\boldsymbol{z}_j^{OEM}$ represents a partial data stream from the OEM sensor measurements.

The output data $\boldsymbol{z}_j$ can be collected in real-time, during the printing of a single layer (temperature measurement) and in between layers (layer-to-layer metrology). The presented framework can accommodate measurements with different time scales. For consistency of presentation, this work will focus on real-time data collected continuously during the printing of layers. As an illustrative example, FDM machines have heating actuators for the extruders and the heated print bed. A GCode command at time-step $j$ ($\boldsymbol{r}_j$) is interpreted and a reference temperature for an extruder ($\hat{\boldsymbol{r}}_j$) is sent to the OEM controller. The OEM controller has a control algorithm that takes the control input and the measurement $\boldsymbol{z}_j^{OEM}$ from the OEM sensors to compute an appropriate actuator input $\boldsymbol{u}_j$ which, in turn changes the temperature of the extruder ($\boldsymbol{y}_j$). External sensors can be used to measure this temperature change ($\boldsymbol{z}_j^{ext}$).

The output data $\boldsymbol{z}_j$ of an AM Plant is transferred through a predefined transmission protocol ($e.g.$ TCP/IP) in real-

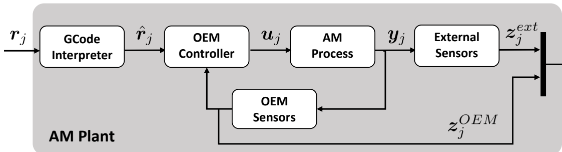


Fig. 2. Block diagram for an AM plant using GCode reference list inputs.

time (at a sampling rate) to the DT Interface. In theory, the reference list $\boldsymbol{r}$, the current-time ($j$) reference to the AM plant $\boldsymbol{r}_j$, and the inputs for the actuators of the AM plant $\boldsymbol{u}_j$ are communicated through the DT Interface. However, in practice, the continuous states of the machine (e.g. position and extruder temperature), controller input $\hat{\boldsymbol{r}}_j$, and the actuator inputs $\boldsymbol{u}_j$ are generally not observable from the AM plant. For this purpose, pre-processing, filtering, state estimation, and event detection on the original data streams $\boldsymbol{z}_j, \boldsymbol{u}_j$, and $\boldsymbol{r}$ (based on the availability on a specific AM plant) are implemented in the DT interface, to generate useful data for the models in the DT Core. An example of the different data streams and their use in a DT is shown in the case study.

Streaming data are sampled at a sampling rate $\tau$ and the data at time-step $j$ are prepared as a state $\boldsymbol{x}_j = [\boldsymbol{q}_j, \boldsymbol{x}_j^r]^T$, where the real-valued continuous states are denoted with $\boldsymbol{x}_j^r \in \mathbb{R}^{n_r}$, and the discrete states denoted with $\boldsymbol{q}_j \in \{\boldsymbol{q}_1, \ldots, \boldsymbol{q}_{n_d}\}$. State estimation and event detection on the data streams from the AM plant is used for evaluating $\boldsymbol{x}_j$. The data streams from the AM plant and the state $\boldsymbol{x}_j$ are shared with a database at the sampling rate $\tau$.

### B. DT Core

The most important purpose of a DT is to provide models of the physical system with real-time updated state information. For this purpose, an AM hybrid automaton (AM-HA) model that captures both the continuous and discrete-event dynamics of the AM process is included in the DT core. A hybrid system model for the continuous and discrete-event dynamics of a micro-AM deposition system is proposed in [19]. Here, we propose a general-purpose hybrid automaton for DT development with various AM-processes. Definition of the AM-HA extends the definition of the functional state model in [2].

**Definition 1.** *[AM-Hybrid Automaton] An AM-HA is a tuple* $\mathcal{H} = (Q, X, U, \Sigma, f, G, R, Init)$ *where:*

- $Q = \{\boldsymbol{q}_1, \ldots, \boldsymbol{q}_{n_d}\}$ *is the set of discrete states*
- $X \subseteq \mathbb{R}^{n_r}$ *is the space of real-valued states*
- $U \subseteq \mathbb{R}^{n_u}$ *is the space of admissible actuator inputs*
- $\Sigma \subseteq Q \times Q$ *is the set of discrete transition events (edges)*
- $f : Q \times X \times U \rightarrow \mathbb{R}^{n_r}$ *is a vector field for the discrete time dynamics of the system ,such that* $\boldsymbol{x}_{j+1} = f(\boldsymbol{q}_i, \boldsymbol{x}_j^r, \boldsymbol{u}_j), \boldsymbol{q}_i \in Q, \boldsymbol{x}_j^r \in X, \boldsymbol{u}_j \in U$
- $G : \Sigma \rightarrow 2^X$ *is a set of guards*
- $L : \Sigma \times X \rightarrow 2^X$ *is a reset map for the continuous states*
- $Init = \{(\tilde{\boldsymbol{q}}, \tilde{\boldsymbol{x}}) | \ \tilde{\boldsymbol{q}} \in Q, \tilde{\boldsymbol{x}} \in X\}$ *is the initial state.*

The hybrid state of $\mathcal{H}$ at time instant $j$ is given as $\boldsymbol{x}_j \in Q \times X$. Streaming data (with the estimated and observed states) from the DT Interface is pushed to the DT Core and the state $\boldsymbol{x}_j$ is updated in each time-step $j$ to track the discrete and continuous states of the system. The hybrid states of $\mathcal{H}$ are assumed to be observable. By updating the hybrid state of the system in real-time, transitions of the model ($e_j \in \Sigma$) are also tracked. If some events are observable through the output $\boldsymbol{z}_j$, the events are used for updating the hybrid state as well. The hybrid system $\mathcal{H}$, is encoded in a predefined format ($e.g.$ XML, JSON). The $\mathcal{H}$ is shared with the DT Function and the up-to-date state $\boldsymbol{x}_j$ is shared every time-step with the DT Function (Fig. 1).

The AM-HA in the DT Core has the capability to predict the future state progression $\{\boldsymbol{x}_{j+1}\}_{j=1}^{n_h}$ for a horizon of length $n_h$ based on the current state, given that $\boldsymbol{r}_j$ and $\boldsymbol{u}_j$ are provided for the prediction horizon. The prediction task for a given horizon is done by evaluating traces of $\mathcal{H}$, with the initial condition as $Init = \boldsymbol{x}_j$ and the actuator inputs $\boldsymbol{u}_j$ derived from the reference list $\boldsymbol{r}$. A detailed analysis on the simulation of the AM-HA $\mathcal{H}$ is subject for future work.

DT Core includes a database to store the evolution of state trajectories $\boldsymbol{x}_j$ as well as the data in the AM-workflow. The reference list $\boldsymbol{r}$ of the AM process is also stored in the database. This way, DT Function may request a real time data stream and the reference data from the DT Core.

### C. DT Function

DT Function makes use of the data from the DT Core to perform various tasks including performance analysis and anomaly detection. DT Function allows various applications to be integrated with the DT, given that proper data types are defined. Based on the available data streams inside the DT, a generic DT Function $g$ has the following data streams available.

- $\boldsymbol{\Phi} = \bigwedge_{\forall \phi_j \in \Phi} \phi_j$: a conjunction of $n_s$ propositions for the AM process
- $\bar{\boldsymbol{x}} = \{\boldsymbol{x}_i\}_{i=t_0}^{t_0+j+n_h}$: sequence of hybrid states starting from initial (initialization of the DT) time-step $t_0$ up to current time-step $j$ and the prediction horizon $n_h$
- $\bar{z}, \bar{\boldsymbol{u}}$: sequences of measured AM process outputs and actuator inputs between time-steps $[t_0, j]$

Utilizing the available input data streams, DT function outputs $\Theta$. Depending on the specific application, the structure of $\Theta$ may differ. Two illustrative DT Functions are discussed in the case study. A DT Function $g$ may use all the available data streams although it is not required to do so. The applications in this work lay the foundation for decision making and in-situ control of FDM process using DTs.

*1) Performance Monitoring:* The AM process is subject to exogenous disturbances, such as material impurities, noise in the environment, and mechanical wear of the AM plant and actuators. To understand the performance of the AM process, certain key performance indicators (KPIs) must be devised. By evaluating KPIs in real-time and between different runs of an AM plant, the performance of the AM plant is analyzed.

A KPI may be evaluated during a layer deposition, per layer, or per process to keep track of historical performance

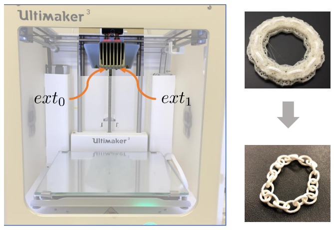


Fig. 3. The Ultimaker 3 used in the case study (left). Printer part using dual extrusion FDM (right top) and the resulting chain geometry with the support material removed (right bottom).

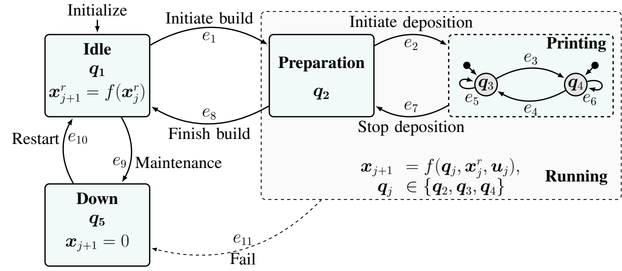


Fig. 4. The AM-HA model for dual extruder printer used in the case study.

of an AM process. A DT Function evaluates the KPIs of the AM process and reports the progression of a certain KPI in the output $\Theta$. An example KPI and its evaluation utilizing DT Function is given in the case study.

*2) Formal Logic-based Anomaly Detection:* Though it is possible to monitor the output of the AM plant $\boldsymbol{z}$, functional dynamics information is not apparent from the data streams in $\boldsymbol{z}$. For this purpose, the traces of the states of AM-HA $\mathcal{H}$ are monitored by the DT Function ($\boldsymbol{s}(t)$ is taken as $\bar{\boldsymbol{x}}$) to check the satisfaction of STL formulas given in $\boldsymbol{\Phi}$. In this work we assume that a subject matter expert specifies the correct STL formulas for the desired behavior of a system. Examples of simple STL specifications, such as allowable working temperature ranges for specific material selections such as polylactic acid (PLA) and (polyvinyl alcohol) PVA for an FDM process, are given in the case study.

A simple STL monitoring scheme is proposed in this work. Based on the maximum window size ($b-a$, for the window $[a,b]$) of a certain property $\phi_i$, the signal-batch size $\beta \in \mathbb{N}$ for the monitoring task is computed off-line. As a signal-batch, a sequence of hybrid states $\{\boldsymbol{x}_j\}_j^{j+\tau\beta}$ starting at initialization time $t_0$ becomes available, the STL properties are checked on each signal-batch, and an anomaly is flagged if any STL specification is violated. The batch processing induces a delay between the estimation and STL monitoring task, thus the batch size should be chosen according to the application type and computational capacity. If $\beta$ is large, satisfaction of global properties may lead to issues due to the noise and disturbances in the measured signal. A global property must be satisfied for the window size in the specification, thus if the signal-batches and the window sizes are too big, the measured signal is prone to faulty violations due to measurement noise. On the other hand, small $\beta$ may cause faulty violations of *eventually* properties. If there are delays in some of the measurement signals, an eventually property with small $\beta$ may result in a premature violation of the property. Note that, different $\phi_i \in \Phi$ may have different $\beta_i$, and specification violation can be checked in parallel. Efficient implementations and computation of batch sizes for monitoring is subject for future work.

## IV. Case Studies in FDM

### A. Setup

An Ultimaker 3 printer is used for the case study, shown in Fig. 3. The printer has two extruders. Left extruder in the picture, $ext_0$, extrudes the structural material (PLA), while the right extruder, $ext_1$, extrudes the support material (PVA). A 3D chain geometry is printed for the case study, which requires the use of both structural and support material during the process (Fig. 3). Process data is collected from the Jedi API of Ultimaker[1]. A real-time data collection pipeline is set up using ADEPT[2] framework of Applied Dynamics International (ADI). A data server is set up to collect data during the print via the API at a fixed sampling rate of $\tau = 200\ ms$. The sampling rate is set based on the rate of data availability through the Ultimaker API. The ADI data collection framework is used for collecting material usage and temperature data about the two extruders shown in Fig. 3.

*1) AM Plant:* The AM-HA for the experimental setup is shown in Fig. 4. The system is initialized as idle ($\boldsymbol{q}_1$), and as real-time data is streamed through the DT Interface, the hybrid state $\boldsymbol{x}_j$ of the hybrid automaton is updated. The super-state ***Printing*** has two sub-states that model the use of two extruders shown in Fig. 3. The model shown in Fig. 4 has eleven transitions $\Sigma = \{e_1, \dots, e_{11}\}$ and five discrete states $Q = \{\boldsymbol{q}_1, \dots, \boldsymbol{q}_5\}$.

For an FDM process, let $M = \{m_1, \dots, m_n\}$ be the set of materials ($m_i$) available for a specific machine. In addition, define $T^i(t) \in \mathbb{R}$ as the temperature reading at the $i^{th}$ extruder, where $t \in \mathbb{R}$ is the time argument. Discrete time readings of the temperature are denoted with subscripts *e.g.* $T^i_j, t \in j\tau, j = 0, 1, \dots$. Then the continuous states for the temperature of two extruders are given as $\boldsymbol{x}^r_j = [T^0_j\ T^1_j]^T$, and the inputs to heating actuators of two extruders are given as $\boldsymbol{u}_j = [u^0_j\ \ u^1_j]^T$. The heating dynamics of the FDM machine are given by $\boldsymbol{x}_{j+1} = f(\boldsymbol{q}_j, \boldsymbol{x}^r_j, \boldsymbol{u}_j)$, where $\boldsymbol{q}_j \in Q$. Since the temperature readings of the two extruders are directly measurable using the data collection framework, the heating dynamics of the FDM machine can be computed, meaning that we can directly evaluate the states $\boldsymbol{x}^r_j$ and the heating input $\boldsymbol{u}_j$ of the system from the output $\boldsymbol{z}_j$. The state and input measurements are used in this case study for KPI and STL monitoring.

There is no heating input in the **Idle** state ($\boldsymbol{q}_1$), thus the dynamics of $\boldsymbol{q}_1$ is the autonomous cooling dynamics with $\boldsymbol{u}_j = 0$. Similarly, the system has no dynamics in the **Down** state ($\boldsymbol{q}_5$).

*2) KPI Monitoring:* A novel KPI to track the energy efficiency of the AM plant is proposed in this work. Define

[1] http://software.ultimaker.com/jedi/api/
[2] https://www.adi.com/products/adept-framework/

$\boldsymbol{\mu}^i[k] = [u^i(t_0), u^i(t_0+\tau), \ldots, u^i(t_f)]^T$ as the vector of actuator input sequence in $t \in [t_0, t_f]$ for the $i^{th}$ extruder at layer $k$. Additionally, define $\ell_i(m_j, k)$ as the total length of material $m_j$ used by the $i^{th}$ extruder at layer $k$. Then the energy efficiency KPI for extruder $i$ at layer $k$ is defined as $\mathcal{E}_i[k] = \left\|\boldsymbol{\mu}^i[k]\right\|_1 / \ell_i(m_j, k)$.

The majority of the energy consumption in an FDM process is due to the heating of materials. The KPI $\mathcal{E}_i[k]$ measures how much energy is consumed for heating versus the amount of actual material extruded for the printing process. Experimental results of this KPI with the case study FDM setup are given below.

*3) STL Monitoring:* The $\beta$ in this case study is chosen to represent the smallest time interval for the activity of either extruder. For the signals shown in Fig. 6, the batch size $\beta$ is chosen as $250\tau = 50sec$. Note that horizon lengths for individual STL properties may differ. The bounds on the STL properties can be set according to different physical phenomena such as a desired viscosity of a material in a certain temperature range, or the maximum temperature that a certain extruder system allows. The STL properties in this case study are defined based on the preferred working ranges of temperatures for the structural and support materials $m_1$ and $m_2$ respectively. Additional STL properties are defined for the performance of the heating actuators on the two extruders as the following.

$$\phi_1 = \square_{[0,\beta]} \left[\boldsymbol{q}_3 \rightarrow |\Delta T^0| \leq \alpha_1(m_1)\right]$$
$$\phi_2 = \square[\neg \boldsymbol{q}_3 \, \boldsymbol{\mathcal{U}}_{[0,\beta]} \boldsymbol{q}_3 \rightarrow \Diamond_{[0,\tau_s^1]} \left(\square_{[0,\tau_s^2]} |\Delta T^0| \leq \alpha_2(m_1)\right)]$$
$$\phi_3 = \square_{[0,\beta]} \left[\boldsymbol{q}_4 \rightarrow |\Delta T^1| \leq \alpha_3(m_2)\right]$$
$$\phi_4 = \square[\neg \boldsymbol{q}_4 \, \boldsymbol{\mathcal{U}}_{[0,\beta]} \boldsymbol{q}_4 \rightarrow \Diamond_{[0,\tau_s^1]} \left(\square_{[0,\tau_s^2]} |\Delta T^1| \leq \alpha_4(m_2)\right)]$$

where $\Delta T^k = T^p(m_i) - T_j^k$ is the temperature error for extruder $k$ at time $j$, $T^p(m_i)$ is the printing temperature for the material $m_i$, $\tau_s^1 = 10$ is the settling time, $\tau_s^2 = 15$ is the steady-state time, $|\cdot|$ is the $L_1$-norm (absolute value), $m_1, m_2$ are the materials used in extruders 0 and 1 respectively, and $\alpha_i(\cdot)$ are material dependent bounds for the satisfactory execution of the FDM process. The printing temperature for the structural material is $T^p(m_1) = 205°$C, and the printing temperature for the support material is $T^p(m_1) = 225°$C.

The property $\phi_1$ reads as; whenever the extruder 0 is active, the $L_1$ norm between the temperature reading and the printing temperature should be always bounded by $\alpha_1(m_1)$. The bound on the structural material (PLA) is set as 10°C, thus $\alpha_1(m_1) = 10°$C. The property $\phi_3$ defines the same bound for the support material as $\alpha_3(m_2) = 10°$C.

The property $\phi_2$ describes that whenever the extruder 0 is switched from inactive to active in the time interval $[0, \beta]$, the temperature of the extruder should eventually reach the bound $\alpha_2(m_1)$ within $\tau_s^1$ seconds and stay within the bound for $\tau_s^2$ seconds. This property is similar to the rise-time and settling time of a dynamic system. The bound $\alpha_2(m_1)$ is defined based on the 2% bound around the printing temperature, thus $\alpha_2(m_1) = 4.1°$C. Similarly, we have $\alpha_4(m_2) = 4.5°$C. The temperature evolution of the FDM is measured during the case study and the satisfaction of the conjunction of all the STL properties are evaluated for the measured signal.

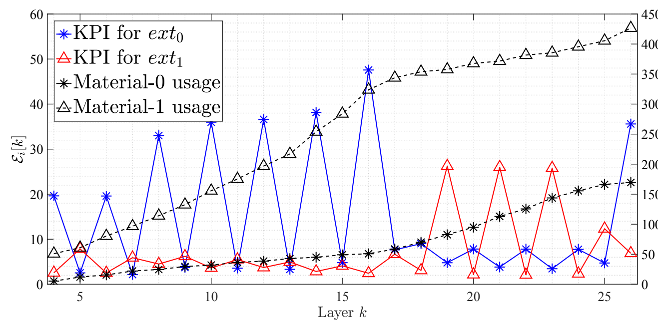


Fig. 5. Star markers show the KPI (on the left) and cumulative material usage (on the right) for the $ext_0$. Triangle markers show the KPI (on the left) and cumulative material usage (on the right) for the $ext_1$.

Note that tracking the STL properties from the time-series measurement of the AM process is not trivial. The proposed DT framework (Fig. 1) with the hybrid model is required to monitor the data with the discrete event and continuous states to detect anomalies. STL properties with the hybrid model define a formal infrastructure between the temporal arguments, discrete states, and continuous states to perform an efficient anomaly detection. Various STL specifications may be devised for specific needs in an anomaly detection application, utilizing the DT framework proposed in this work.

## *B. Results*

*1) Performance monitoring:* Fig. 5 shows the energy efficiency KPI $\mathcal{E}_i[k]$ for the layers $k \in [4, 26]$ in the experiment. The signal $\boldsymbol{\mu}^i[k]$ is calculated using the heater input signal monitored using the ADI data extraction framework. The heater input signal is normalized to be in the range of $[0, 1]$ and divided by the material use of the certain extruder in a layer to compute the KPI $\mathcal{E}_i[k]$ for each layer, for both extruders.

From the analysis of results shown in Fig. 5, it is concluded that the fluctuation of $\mathcal{E}_i[k]$ between layers is largely affected by the changing lengths of material used in each layer and the energy consumed for reheating of materials between material changes. Thus, using the same material for longer in a single layer is more efficient since less re-heating energy is consumed per length of printed material in a layer. The value of the $\mathcal{E}_i[k]$ for both extruders are presented to a user through the output of the DT. By tracking this KPI between different runs of the same AM plant, it is possible to get insight on degradation in the heating system. Degradation will lead into lower efficiency, which results in an increasing trend of $\mathcal{E}_i[k]$ between different runs of the AM plant.

*2) Anomaly detection:* Figure 6 shows the signals $T^i(t), i = 1, 2$ from the data collected during the experiments, and the evolution of discrete states $\boldsymbol{q}_3$ and $\boldsymbol{q}_4$ with respect to time. A pre-processing step is implemented in the DT Interface to detect the times when each extruder is active. The events that result in transition between discrete states, temperature readings, and bounds are shown on the

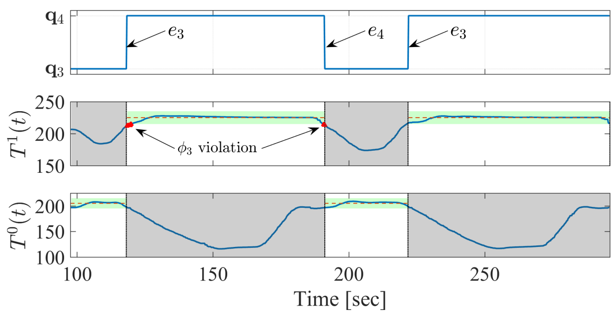


Fig. 6. *Top:* Discrete states and transitions of the hybrid system given in Fig. 4. *Middle/Bottom:* The temperature of $ext_1/ext_0$ versus time with inactive periods grayed-out and printing temperatures are shown with red dashed lines. Temperature bounds are shown with green fills. Violation of $\phi_3$ is shown with the red triangle markers.

top plot of Fig. 6. As shown in Fig. 6, the measured signals are analyzed in the time intervals on which each extruder is active based on the STL specifications. Bounds shown in green are set by $\alpha_2$ and $\alpha_4$, and the printing temperatures are shown with red dashed lines.

The intervals on times $[118.4, 120]$ and $[190.8, 191.2]$ violate the proposition $\phi_3$, as shown in Fig. 6. The violation is caused by the temperature of extruder 1 being outside of the bounds set by $\alpha_3(m_2)$. Note that the bounds set by $\alpha_3(m_2)$ are violated multiple times through the experiment, but since the STL properties define the time intervals in which a property must be satisfied with respect to the states of the hybrid model, an anomaly is accurately detected. The detected anomaly is reported to the user through the output $\Theta$ of the DT. Measurement signals in the case study satisfy the heating dynamics conditions set by propositions $\phi_2$ and $\phi_4$, which are not explicitly illustrated due to space constraints.

## V. Conclusion

A DT architecture for performance monitoring and anomaly detection in AM processes is presented in this work. The proposed DT has a hybrid automaton model in its core, that is able to capture both the functional and the continuous dynamics of an AM process, and a flexible function block that may be utilized for multiple purposes such as anomaly detection and performance monitoring. The use of STL specifications for anomaly detection in AM processes is proposed as a novel contribution. Additionally, a new KPI metric is proposed for performance monitoring in FDM processes. The proposed DT is implemented on an off-the-shelf FDM 3D printer, and experimental results on performance monitoring and anomaly detection are presented. Experimental results show that the proposed DT provides additional monitoring and analysis capabilities to the physical system.

Future work will focus on the development of efficient techniques for real-time data processing and STL monitoring. Leveraging the architecture presented in this work, efficient schemes for system monitoring and control for multiple AM machines will be investigated in future developments.

## Acknowledgment


The authors would like to thank Applied Dynamics International for the real time data collection pipeline and software support. The authors also thank Ilya Kovalenko for his helpful discussions and comments on the work.


## References


[1] W. Gao, Y. Zhang, D. Ramanujan, K. Ramani, Y. Chen, C. B. Williams, C. C. Wang, Y. C. Shin, S. Zhang, and P. D. Zavattieri, "The status, challenges, and future of additive manufacturing in engineering," *Computer-Aided Design*, vol. 69, pp. 65–89, 2015.

[2] E. C. Balta, D. M. Tilbury, and K. Barton, "A centralized framework for system-level control and management of additive manufacturing fleets," in *2018 IEEE 14th International Conference on Automation Science and Engineering (CASE)*. IEEE, 2018, pp. 1071–1078.

[3] L. Santana, J. L. Alves, and A. d. C. S. Netto, "A study of parametric calibration for low cost 3D printing: Seeking improvement in dimensional quality," *Materials & Design*, vol. 135, pp. 159–172, 2017.

[4] P. K. Rao, J. P. Liu, D. Roberson, Z. J. Kong, and C. Williams, "Online real-time quality monitoring in additive manufacturing processes using heterogeneous sensors," *Journal of Manufacturing Science and Engineering*, vol. 137, no. 6, p. 061007, 2015.

[5] E. C. Balta, D. M. Tilbury, and K. Barton, "Control-oriented modeling and layer-to-layer stability for fused deposition modeling: A kernel basis approach," in *2019 American Control Conference (ACC)*. IEEE, 2019.

[6] S. R. Chhetri, S. Faezi, A. Canedo, and M. A. Al Faruque, "Quilt: Quality inference from living digital twins in iot-enabled manufacturing systems," *ACM/IEEE Conference on Internet of Things Design and Implementation*, 2019.

[7] J. Moyne, E. Balta, I. Kovalenko, Y. Qamsane, and K. Barton, "The digital twin in the manufacturing ecosystem of the future," *1st International Workshop on Smart Manufacturing Modeling and Analysis*, 2019.

[8] F. Tao, J. Cheng, Q. Qi, M. Zhang, H. Zhang, and F. Sui, "Digital twin-driven product design, manufacturing and service with big data," *The International Journal of Advanced Manufacturing Technology*, vol. 94, no. 9-12, pp. 3563–3576, 2018.

[9] R. Rosen, G. Von Wichert, G. Lo, and K. D. Bettenhausen, "About the importance of autonomy and digital twins for the future of manufacturing," *IFAC-PapersOnLine*, vol. 48, no. 3, pp. 567–572, 2015.

[10] D. M. Tilbury, "Cyber-physical manufacturing systems," *Annual Review of Control, Robotics, and Autonomous Systems*, vol. 2, 2019.

[11] Y. Qamsane, E. C. Balta, D. M. Tilbury, and K. Barton, "Dynamic rerouting of cyber-physical production systems in response to disruptions based on SDC framework," in *2019 American Control Conference (ACC)*. IEEE, 2019.

[12] G. L. Knapp, T. Mukherjee, J. S. Zuback, H. L. Wei, T. A. Palmer, A. De, and T. DebRoy, "Building blocks for a digital twin of additive manufacturing," *Acta Materialia*, vol. 135, pp. 390–399, 2017.

[13] T. DebRoy, W. Zhang, J. Turner, and S. S. Babu, "Building digital twins of 3D printing machines," *Scripta Materialia*, vol. 135, pp. 119–124, 2017.

[14] H. Sun, P. K. Rao, Z. J. Kong, X. Deng, and R. Jin, "Functional quantitative and qualitative models for quality modeling in a fused deposition modeling process," *IEEE Transactions on Automation Science and Engineering*, vol. 15, no. 1, pp. 393–403, 2018.

[15] M. R. Shahriar, S. N. Al Sunny, X. Liu, M. C. Leu, L. Hu, and N.-T. Nguyen, "MTComm based virtualization and integration of physical machine operations with digital-twins in cyber-physical manufacturing cloud," in *2018 IEEE CSCloud/EdgeCom*. IEEE, 2018, pp. 46–51.

[16] K. M. Alam and A. El Saddik, "C2PS: A digital twin architecture reference model for the cloud-based cyber-physical systems," *IEEE Access*, vol. 5, pp. 2050–2062, 2017.

[17] M. Saez, F. Maturana, K. Barton, and D. Tilbury, "Anomaly detection and productivity analysis for cyber-physical systems in manufacturing," in *2017 13th IEEE Conference on Automation Science and Engineering (CASE)*, Aug 2017, pp. 23–29.

[18] A. Donzé, T. Ferrere, and O. Maler, "Efficient robust monitoring for stl," in *International Conference on Computer Aided Verification*. Springer, 2013, pp. 264–279.

[19] I. Spiegel, I. Kovalenko, D. Hoelzle, P. M. Sammons, and K. L. Barton, "Hybrid modeling and identification of jetting dynamics in electrohydrodynamic jet printing," in *2017 IEEE Conference on Control Technology and Applications (CCTA)*. IEEE, 2017, pp. 695–701.