Paper ID APEN-MIT-2022_7183
Applied Energy Symposium: MIT A+B
July 5-8, 2022 • Cambridge, USA

Development of a Digital Twin Platform for Industrial Energy Systems

Paul Schwarzmayr
Institute of Energy Systems and Thermodynamics
TU Wien
Vienna, Austria
https://orcid.org/0000-0003-4735-3447

Felix Birkelbach
Institute of Energy Systems and Thermodynamics
TU Wien
Vienna, Austria
https://orcid.org/0000-0003-4928-6209

Lukas Kasper
Institute of Energy Systems and Thermodynamics
TU Wien
Vienna, Austria
https://orcid.org/0000-0001-9474-3021

René Hofmann
Institute of Energy Systems and Thermodynamics
TU Wien
Vienna, Austria
https://orcid.org/0000-0001-6580-4913

Abstract—The reduction of waste heat in energy intensive industrial processes in combination with digital technologies will play a key role for the development and decarbonization of modern industrial energy systems. In the last few years, a significant share of the CO_2 related to energy was emitted by the industry sector. Since industrial processes often are batch processes, waste heat recovery in these processes requires thermal energy storage systems for closing the temporal gap between energy supply and demand. The ongoing digitalization in the field of industrial energy systems enables modern applications like digital twins to increase the efficiency of energy intensive processes. This paper presents the implementation of a five-dimensional digital twin platform for a packed bed thermal energy storage test rig. The five-dimensional digital twin platform allows the development of services and applications in interdisciplinary teams and facilitates their interaction on a standardized platform. By that the digital twin helps to make modern industrial energy systems more efficient.

Keywords—thermal energy storage, digital twin, packed bed, energy efficiency, waste heat recovery, decarbonization, industrial energy systems

I. INTRODUCTION

To curb global warming and to reach the goals defined in the Paris Agreement global CO_2 emissions need to be reduced by more than 75% until the year 2050 [1]. A considerable share (more than 40%) of the CO_2 emissions from energy in 2019 were emitted by the industry sector [2]. Besides the transition to renewable energy systems, a reduction of primary energy consumption will be necessary. An important lever to reduce primary energy consumption is the management and recovery of waste heat. This can be very challenging because most industrial processes are batch processes where the availability of waste heat and the demand of process heat do not occur at the same time. Thermal energy storage technologies can increase flexibility and efficiency of industrial processes by matching heat supply with demand. The rapid development of digital technologies and the ubiquitous availability of data enables a multitude of new applications to increase utility, reliability, productivity, and sustainability of industrial energy systems. The concept of a digital twin (DT) is gaining in importance [3] and is considered a key technology to link specialized services such as predictive maintenance, fault detection and operational optimization in one uniform platform. A DT can provide a platform for services from different domains to interact with each other to make the most of their potential.

A. Industrial use case – the Linz-Donawitz-process

In this paper a steel producing process – the so-called *Linz-Donawitz* (LD) process – at *voestalpine Stahl Donawitz GmbH* (VASD) in Leoben, Austria, will be taken as use case for the demonstration and evaluation of the examined technologies. As most industrial processes, the LD-process is a batch process with a cycle time of 30-60 minutes. During the first half of a cycle the LD-process produces large amounts of crucible gas. This hot crucible gas, which carries a considerable amount of abrasive and adhesive metal dust, exits the crucible with a temperature of about 1400 degrees Celsius. In a radiation heat exchanger, the crucible gas is cooled to around 700 degrees Celsius producing steam for other processes and electricity generation. For the exhaust gas purification, the process gas must have a maximum temperature of 150 degrees Celsius. Due to the high amount of metal dust in the crucible gas convective heat exchangers would erode quickly. Therefore, the further cooling is realized by injecting water in an evaporation cooler which results in a substantial amount of high temperature heat being wasted. During the second half of a cycle the crucible is emptied, and no exhaust gas is produced. To make the waste heat, which is produced in the first half of a cycle, reusable in other processes the evaporation cooler needs to be replaced with a thermal energy storage. The only type of thermal energy storage found to be suitable to store high temperature heat from LD-crucible gas is a packed bed thermal energy storage (PBTES). A PBTES is a sensible heat thermal energy storage which normally uses rocks as storage material. For a more detailed description of PBTES systems the authors refer to Esence et al. [4]. The main advantages of the PBTES for this specific use case are its robustness against the metal dust in the process gas and the high power rates. In this type of storage, the heat is directly transferred from the process gas to the storage material and vice versa. Hence there are no heat exchanger tubes that could be damaged. The high power rates are reached because of the large heat exchange surface of the packed bed.

In Fig. 1 the integration of the PBTES into the LD-process at VASD is depicted. Instead of the evaporation cooler, which cools the process gas to about 150 degrees Celsius, a vertical PBTES is installed. Like the evaporation cooler the PBTES is capable of ensuring a maximum temperature of the exhaust gas and additionally enables the recovery of waste heat. During the first half of the LD-process' cycle the process gas, which exits the radiation heat exchanger with temperatures around 700 degrees Celsius, flows through the packed bed from top to bottom and charges the PBTES. The storage can be discharged during the second half of a cycle with fresh air. This cold, fresh air flows through the PBTES from bottom to top and is heated up by the hot storage material. The hot gas which exits the PBTES during discharging can be used for preheating in other processes or for steam production.

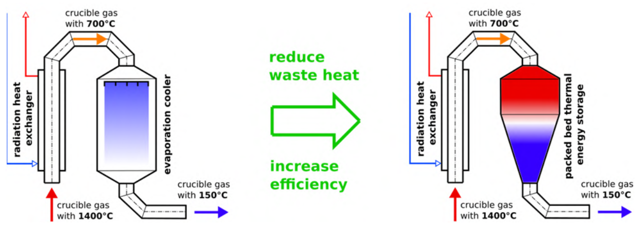

Fig. 1. Use case at voestalpine Stahl Donawitz GmbH

Despite all these advantages, integrating a thermal energy storage into an existing process involves various challenges. An economically optimal and safe operation of the PBTES and a proper storage management are key. Due to the very harsh environment the PBTES will be operated in, the behavior and some properties of the storage will significantly change over time. Problems such as damaged sensors or a clogging of the packed bed will occur frequently and the adhesive metal dust in the process gas will lead to changing properties of the PBTES. To track these changes and detect or even predict errors we implemented a digital twin with specialized services for a PBTES test rig.

II. Material and Methods

A. *Packed bed thermal energy storage test rig*

For experimental investigations a lab-scale test rig of a PBTES has been erected in the laboratory at TU Wien. As can be seen in Fig. 2, the test rig is a vertical PBTES, and its storage tank consists of a conical steel vessel. For the storage material LD-slag with a grain size of 16 – 32 millimeters was chosen. The LD-slag, which is a by-product of the LD-process, consists of irregular shaped and porous rocks. These properties lead to high power rates, homogeneous and even perfusion of the packed bed and a high temperature stratification in the storage material. The test rig is connected to an air supply unit (ASU) which provides hot air with temperatures up to 400 degrees Celsius and a mass flow of max. 400 kilograms per second. Additionally, the test rig is equipped with several control valves that allow to simulate clogging of the packed bed and leaks in the air pipes. The most important facts and properties of the test rig and the storage material are summarized in TABLE I.

TABLE I: Facts and properties of the lab-scale test rig

Diameter of the conical storage vessel	200 mm (bottom) / 500 mm (middle) / 700 mm (top)
Height of the storage vessel	2050 mm
Volume of the storage vessel	0.405 m³
Mass of the storage material	900 kg
Type of storage material	LD-slag (mostly calcium oxide)
Grain size of storage material	16 – 32 mm

To prepare the PBTES for the deployment as the physical entity in a digital twin, it is equipped with 49 PT100 temperature sensors inside the packed bed and 8 PT100 temperature sensors on the storage tanks surface. The exact position of these sensors inside the packed bed is shown in Fig. 2. This large number of sensors allows for a detailed measurement of the temperature field – especially the location and shape of the thermocline – inside the PBTES.

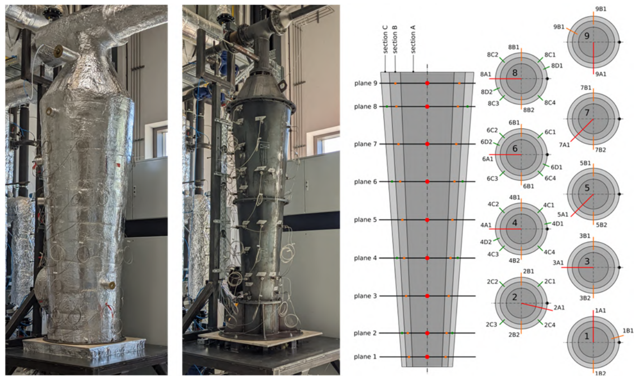

Fig. 2. Packed bed regenerator test rig: insulated (left), uninsulated (middle), positions of temperature sensors (right)

B. *Digital twin platform for industrial energy systems*

The 5D digital twin platform proposed by Kasper et al. [5] builds the foundation for the modeling of the digital twin for the PBTES test rig. Kasper et al. designed their platform to be implemented for industrial energy systems and divided it into five dimensions (compare Fig. 3), following the 5D DT concept by Tao et al. [3].

The only dimension located in the physical space is the physical entity. The physical entity can consist of one single aggregate or even a whole energy system. Programmable logic controllers and a supervisory control and data acquisition (SCADA) system form the interface of the physical entity into the virtual space and the rest of the digital twin.

The counterpart of the physical entity in the virtual space is the virtual entity. The virtual entity should be able to represent the behavior and properties of the physical entity as accurately and timely as possible. It can generally consist of various types of models integrated and connected to the other dimensions of the digital twin via a model management layer.

The central dimension of the 5D DT framework is the connection dimension. This dimension represents all the connections between the other four dimensions and is responsible for managing the communication and information flow inside the DT.

The data dimension is the centralized storage of the data gathered from the physical entity and the virtual entity of the

DT. The data dimension not only contains runtime data and simulation results in the form of relational databases, but also information about plant equipment, instrumentation and topology and other semantic information.

The fifth dimension of the DT is the service dimension. Its purpose is to tightly integrate services that can make use of the data, information and models stored in the other four dimensions. In addition to all the application specific services which can be integrated into the service dimension the most important part of this dimension is a service orchestrator. The service orchestrator is responsible for the management and definition of workflows inside the digital twin, and it plays a key role in realizing the potential of digital twin services.

III. IMPLEMENTATION

The 5D DT platform applied to the evaluation case presented in Section II, the PBTES test rig, the 5D DT platform is visualized in Fig. 3.

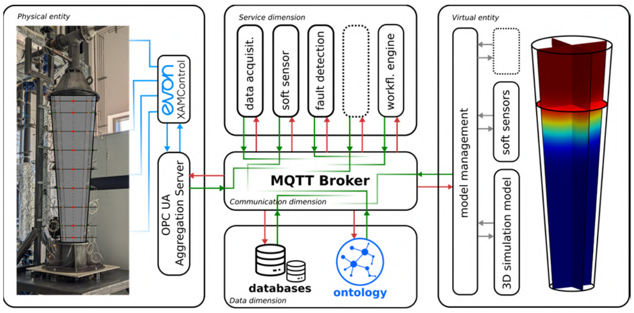

Fig. 3. 5D digital twin framework implemented for the PBTES test rig. Concept after [5]

A. *Physical entity*

The physical entity consists of the PBTES test rig which is equipped with 49 temperature sensors (see in Section II) to be able to monitor the temperature distribution in the packed bed. The SCADA system, which is a part of the physical entity, consists of XAMControl [1] and an open platform communications unified architecture (OPCUA) aggregation server. XAMControl is a software for process automation and in the presented digital twin, it is used for data acquisition and process control. XAMControl collects real time data from all the sensors of the PBTES test rig via OPCUA from the hardware PLCs, preprocesses them and hosts them to the OPC UA aggregation server again via OPC UA. The OPC UA aggregation server, which enables 100% OPC UA compatible communication and data exchange, acts as an interface between the physical and the virtual space of the DT by integrating data from the physical entity to the other four dimensions of the DT.

B. *Virtual entity*

The virtual entity, which is the counterpart of the physical entity in the virtual space, contains a three-dimensional thermal model of the PBTES test rig. This model utilizes the finite volume method and is based on the heat equation and other energy balances for modeling the losses and the energy in- and outputs during charging and discharging of the storage. To increase the flexibility of this model it is designed as a hybrid model and implemented in MATLAB®[2]. This means, several parameters can be adapted and fitted via data-driven methods. Some examples of such adaptable parameters are the heat capacity of the storage material, the heat transfer coefficient between the packed bed and the process gas and the parameters that characterize the heat loss to the surrounding. In addition to the three-dimensional thermal model the virtual entity also includes soft sensors. These soft sensors are purely data-driven models which are modeled and trained on-demand based on runtime data and domain knowledge provided by the ontology. More details about soft sensors and ontologies are given later in this Section.

C. *Connection dimension*

The connection dimension, which can be seen as a central communication hub, is realized by a message queuing telemetry transport (MQTT) broker. We chose the open-source MQTT broker Eclipse Mosquitto [6] for our implementation. MQTT is a topic-based publish-/subscribe network protocol that enables the transportation of messages and data between devices, or in our case the dimensions and services of the DT. Every MQTT message typically consists of a topic and a payload. The topic includes information about the message type (e.g., service name and request or response) and a unique identifier. Every dimension and service of the DT is connected to the MQTT broker as a client and can publish and/or subscribe to different topics. The MQTT broker is responsible for receiving, filtering, and sending messages to the subscribed clients. To allow subscribed clients the proper processing of received messages, every publishing client must follow a certain naming convention of the message payload. The payload typically contains data that the services need as input for their calculations. In the proposed DT framework, every service has a corresponding request topic to which it is subscribed and a corresponding response topic to which it publishes. The mapping of the topics to each other is done by a workflow engine, which will be described in more detail later in this Section. TABLE II shows the structure of MQTT message topic and payloads using the soft sensor service as an example.

TABLE II: Request and response MQTT messages to and from the soft sensor service

MQTT message to start the soft sensor service	
message-topic	5dit/fbr/services/softsensor/request/#
payload	“damagedsensor”: *FBR-TE-4A1*, “faulttime”: *2022-07-05T12:34:56*
MQTT message from soft sensor service to continue the workflow	
message-topic	5dit/fbr/services/softsensor/response/#
payload	empty

Every service-related topic starts with *5dit/fbr/service/*, continues with the service name and ends with either */request/unique_identifier* if it is a request topic or */response/unique_identifier* if it is a response topic. The payload is predefined for each individual topic and can consist of multiple attribute-value pairs and arrays.

[1] https://en.evon-automation.com/

[2] https://de.mathworks.com/

Note: Since every topic starts with *5dit/fbr/services/* and ends with */unique_identifier* we will just call them by *servicename/request* or *servicename/response* if they are mentioned in the text.

D. Data dimension

The "brain" of the DT and centralized storage location for runtime data and simulation results is called the data dimension. In the current status of the project, the data dimension consists of a PostgreSQL [7] relational database and two ontologies which can be accessed via a federated query endpoint called FedX [8]. FedX federates multiple SPARQL protocol and RDF query language (SPARQL) endpoints, which are the access points to information stored in different ontologies, under a single virtual endpoint. This allows users and services to access every information stored in any ontology inside the DT via one single endpoint. The relational database is used for the storage of runtime data from the physical entity and some important metrics such as the power rate or the state of charge (SOC), which are calculated from the runtime data. The ontologies are responsible for knowledge representation and for improving data accessibility inside the DT. To add context information and semantics to the runtime data in the relational database an existing ontology, the sensor, observation, sample, and actuator (SOSA) ontology [9], is reused and extended. To map the data in the relational database to the SOSA ontology the Ontop Framework [10] is utilized. Ontop represents the content of relational databases as virtual knowledge graphs without moving them to another database. Data from the relational database can therefore be accessed via ontology-based data access (OBDA). Based on the mappings in the ontology, Ontop translates SPARQL queries to structured query language (SQL) queries which are executed by the relational database. To add additional information about the location of sensors inside the packed bed relative to each other the SOSA ontology was extended specifically for the present use case. To do so, the storage volume is divided into nine horizontal planes and three radial sections (see Fig. 2). Every sensor is assigned to the plane and section it is located. This way context information such as the neighbors of a specific sensor can be extracted from the ontology. To provide and store information about sensor failures, a RDF4J [11] triple store which makes use of the existing owl-time ontology [12], is utilized. This ontology allows the documentation of error events as well as their start- and end time. If required by another user or service inside the DT this information can again be accessed by querying the federated query endpoint. Fig. 4 shows the structure of all the implemented ontologies combined in one graphic. The blue branch represents all the concepts which could be reused from the SOSA ontology. The extension of the SOSA ontology for the present use case can be seen in the green branch and the orange branch shows the structures and concepts which we reused from the owl-time ontology.

To access the context information which is available in the ontologies, users and services need to send a SPARQL query to the endpoint of the federated query endpoint.

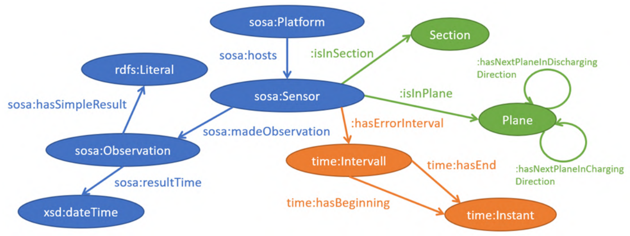

Fig. 4. Structure of the ontologies

An exemplary SPARQL query which aims to get information about which sensors are the nearest sensors (i.e., the neighbors in charging/discharging direction) to sensor FBR-TE-4A1 is presented in Fig. 5.

```
SELECT ?neighborSensors
WHERE {:FBR-TE4A1  :isInSection ?section;
                   :isInPlane ?plane.
       ?plane :hasNextPlaneInDischargingFlowDirection |
              :hasNextPlaneInChargingFlowDirection ?neighborPlanes
       ?neighborSensors :isInSection ?section;
                        :isInPlane ?neighborPlanes.}
```

Fig. 5. Exemplary SPARQL query to access context information (neighbor sensors)

E. Service dimension

As presented in Section II the service dimension is composed of several micro services and a service orchestrator. For the service orchestration a workflow engine called Zeebe [13] was chosen. Zeebe is an open-source workflow and decision engine that enables the execution of business process model and notation (BPMN) workflows. Via BPMN, workflows and business processes can be modeled in a graphical way and directly be loaded into a workflow engine. The main benefit of BPMN workflows for the development of a digital twin is that they provide a perfect interface for the cooperation of experts from different domains.

In addition to the BPMN workflow engine, four services are implemented at the moment and relevant for the use case considered in this work:

- The data acquisition service, which is responsible for data acquisition from the physical entity, preprocessing, and storage of the sensor data as well as context information in the data dimension.
- The fault detection service, which is currently under development and responsible for the detection of deviation in the sensor data from the expected behavior.
- The soft sensor service, which can react to messages from the fault detection service and automatically train data driven models to compensate for damaged sensors, called soft sensors. Soft sensors are trained based on runtime data as well as context information from the data dimension and can be used to reproduce measurement data that was corrupted by damaged sensors.
- The sensor replacement service, which is an interface for the machine operator to document and integrate information about a replaced or repaired sensor into the ontology of the DT.

IV. EVALUATION

To highlight the benefits more clearly and to evaluate the proposed DT framework the interaction and communication inside the DT will be presented based on an exemplary scenario. For the prediction of the storage behavior and for a proper storage management, the SOC is an important characteristic number. Let us consider a scenario where a temperature sensor inside the PBTES gets damaged and delivers a constant value of zero degrees Celsius. Under normal conditions this failure might go undetected for a while and lead to a significant error in the SOC calculation. This error will propagate to other services and evaluations and make an economic operation of the storage impossible. In this Section we will present how the digital twin framework proposed in Section III can contribute to tackle and overcome these issues. Fig. 6 shows all the involved workflows notated as BPMN workflows and their communication via the ontology. The top box in Fig. 6 represents the ontology, which is not part of BPMN, but is included regardless to make its interaction with the BPMN workflows clearer. The solid arrows define the workflows, and the dashed arrows are visualizing the information flow between service tasks and the ontology.

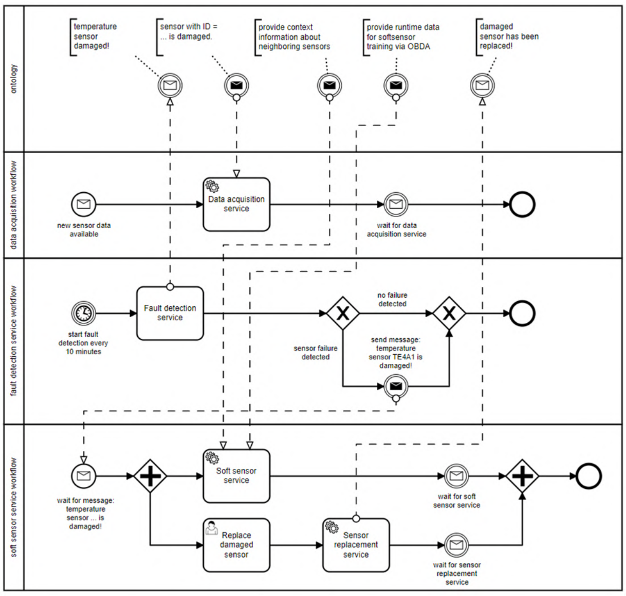

Fig. 6. Communication of data acquisition and soft sensor service via the ontology

The second box in Fig. 6 contains the data acquisition workflow. This workflow is started, when a MQTT message is published on the topic *temperaturesensorvalues*. In the current implementation this is done by the OPC UA aggregation server which is part of the physical entity. Every minute the OPC UA aggregation server fetches the life values of every temperature sensor in the physical entity via OPC UA and publishes them to the before mentioned MQTT topic as attribute-value pairs in the message payload. In the next step of this workflow the workflow engine publishes an MQTT message to the topic *dataacquisition/request* which starts the data acquisition service. The data acquisition service takes an array of temperature values from the message payload as input, calculates some characteristic metrics based on them and stores the results in the DTs database. Before these calculations are performed the data acquisition service sends a SPARQL query to the ontologies' federated query endpoint to check if there is record of any damaged sensor. If there is a damaged sensor, the federated query endpoint responds with an MQTT message which includes the ID of the damaged sensor. With this information, the data acquisition service fetches a soft sensor for the damaged sensor from the virtual entity to reproduce the true sensor value. When the data acquisition service is done with its calculations, it publishes an MQTT message to the topic *dataacquisition/response*, which continues and finishes this workflow. In the third box in Fig. 6 a prototypical fault detection service is shown. In contrast to the data acquisition workflow, this workflow is started every 10 minutes by the workflow engine by publishing an MQTT message to the topic *faultdetection/request*. When the fault detection service is started it requests runtime data from the last 10 minutes from the ontologies and checks if there are any deviations. Because the fault detection service is currently in the development phase, this step will not be described in more detail in this work. If no deviation could be detected this workflow is terminated with no further action. But if the fault detection service detects a damaged sensor, it documents this in the ontology by a SPARQL update query and at the same time publishes an MQTT message to the topic *temperaturesensordamaged*. In the payload of this message the fault detection service includes the ID of the damaged sensor and the time at which the error occurred. As it is depicted in Fig. 6 this MQTT message automatically starts the soft sensor workflow. After the massage-start-event this workflow splits up into two branches and triggers the soft sensor service in the first branch and a user task in the second branch. The soft sensor service starts by querying the ontologies for context information about which sensors are the neighbors of the damaged sensor. After receiving the response from the ontologies, the soft sensor service requests runtime data from the damaged sensor and its neighbors via OBDA which it uses to train a data-driven regression model. This model, which we call a soft sensor, is stored in the virtual entity, and can be used by other services to reproduce true measurements from the damaged sensor. When the soft sensor service is finished with all its tasks it publishes an MQTT message to the topic *softsensor/response* that finishes the first branch of this workflow. In the second branch the user task will result in a pop-up on the machine operators screen with the request to replace the damaged sensor at the next planned maintenance. It continues with the sensor replacement service when the machine operation has confirmed the replacement of the damaged sensor. The purpose of the sensor replacement service is, to record this information in the ontology by closing the error interval of the before damaged and now replaced sensor. When done, the sensor replacement service publishes an MQTT message to the topic *sensorreplacement/response* and thereby finishes the second branch which terminates the whole workflow. The evaluation results of the discussed scenario are shown in Fig. 7. The PBTES was operated according to a predefined schedule (eight charging/discharging cycles) for two days. On the x-axis of each plot the time since the start of the experiment is shown. The graph shows the trajectory of temperature sensor FBR-TE 4A1, which is the central sensor in plane 4 (compare Fig. 2). In the middle plot the SOC calculated by the data acquisition service is displayed. The red dashed line shows the true value of the temperature sensor FBR-TE-4A1 which overlaps with the measured value (blue dotted) for the first 770 minutes of the experiment. Consequently, the error in the state of charge calculation, which is plotted in the bottom plot, is zero during this time interval. After 770 minutes have elapsed, the measured value of FBR-TE-4A1 is manually overwritten with a constant value of 0 degrees Celsius in XAMControl to simulate a sensor fault. Under normal conditions (without the DT) this sensor error would lead to a significant error in the SOC calculation. This error is visualized in Fig. 7 via the trajectory of the blue dotted lines in the middle and bottom plot. With the proposed DT framework, this error can be immediately detected by the fault detection service which triggers further workflows as described earlier in this Section. As can be obtained from Fig. 7, the automatically trained soft sensor reproduces the true value of FBR-TE-4A1 with high accuracy. Thereby the SOC error (related to SOC = 1) of before nearly 15 percent can be reduced to a maximum of three percent.

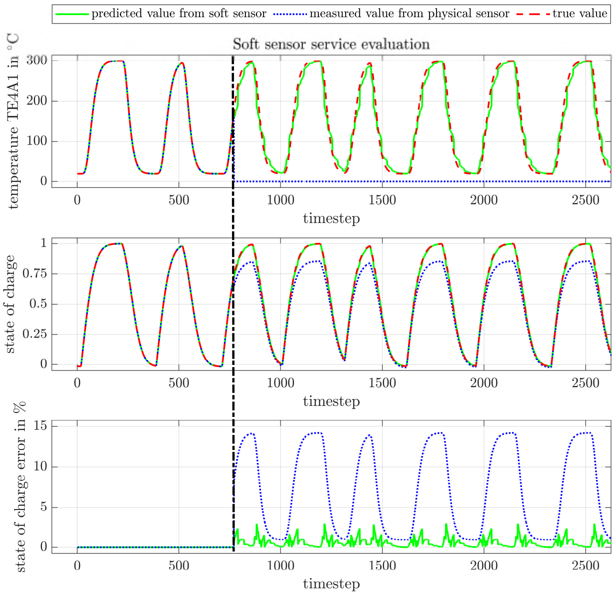

Fig. 7. Evaluation of the proposed framework with soft sensor service and improved SOC estimation

V. Conclusion and Outlook

In this paper we summarize the development of a digital twin for a lab-scale test rig of a packed bed thermal energy storage. For the implementation of the digital twin, a five-dimensional framework was used. For the presented use case – a digital twin for a thermal energy storage – this approach proved to be very successful. The separation of the digital twin into five dimensions facilitates the cooperation in interdisciplinary teams. The encapsulation of domain-specific applications as micro-services allows experts of different fields to develop their services and models independently and deploy them together in one standardized platform. Industrial energy systems often operate under very harsh and constantly changing conditions which makes their economic and efficient operation difficult. The digital twin implemented in this work is able to handle and overcome these problems by combining innovative technologies and expertise from several domains. On the digital twin platform various services can be linked together to make the operation of industrial energy systems more reliable. We showed that the deployment of three simple services and their connection via the connection dimension and the workflow engine already show noticeable advantages.

In the discussed evaluation case, where a soft sensor service is used to compensate a damaged sensor, the digital twin achieves to significantly reduce the impact of a damaged sensor on other services and the physical entity's efficiency. In addition to that, the implementation of a 5D digital twin for industrial energy systems provides many new opportunities for the future. High-fidelity adaptable simulation models that constantly represent the behavior of the physical entity are a perfect basis for operational optimization and the simulation of possible scenarios in the future. Classification models for fault detection could be trained on failures simulated by the virtual entity and deployed to detect these failures in the physical entity via transfer learning. The centralized storage of information in the data dimension provides a data- and knowledge base for machine learning algorithms which could predict failures in the future by recognizing patterns in historic data. The five-dimensional digital twin framework is designed to be implemented on other industrial energy systems and to facilitate the development of complex services of different domains. Due to the scalability of all the five dimensions the digital twin can be extended with new services, models, and workflows at any time without affecting the existing parts of the system.

Acknowledgment

The authors acknowledge funding support of this work through the research project *5DIndustrialTwin* as part of the Austrian Climate and Energy Fund's initiative Energieforschung (e!MISSION) 6th call (KLIEN/FFG project number 881140).

References

[1] IRENA, "World energy transition outlook 2022: 1.5°C pathway", International renewable energy agency, Abu Dhabi, 2022.

[2] IEA, "Greenhouse gas emissions from energy: overview", International energy agency, Paris, 2021.

[3] F. Tao, H. Zhang, A. Liu, A.Y.C. Nee, "Digital Twin in Industry: State-of-the-Art", IEEE Trans. on Ind. Inf., vol. 15, pp. 2405-2415, 2019.

[4] T. Esence, A. Bruch, S. Molina, B. Stutz, J.F. Fourmiguè, "A review on experience feedback and numerical modeling of packed-bed thermal energy storage systems", Solar Energy, vol. 153, pp. 628-654, September 2017.

[5] L. Kasper, F. Birkelbach, P. Schwarzmayr, G. Steindl, D. Ramsauer, R. Hofmann, "Towards a practical Digital Twin platform tailored to the Requirements of Industrial Energy Systems", Appl. Sci., vol. 12, pp. 6981, 2022.

[6] Eclipse Foundation. Eclipse Mosquitto: An open source MQTT broker. https://mosquitto.org/ (accessed June 20, 2022)

[7] The PostgreSQL Global Development Group. PostgreSQL: The World's Most Advanced Open Source Relational Database. https://www.postgresql.org/ (accessed June 20, 2022)

[8] Eclipse Foundation. Federation with FedX. https://rdf4j.org/documentation/programming/federation/ (accessed June 20, 2022)

[9] W3C Recommendation. Semantic Sensor Network Ontology. https://www.w3.org/TR/vocab-ssn/ (accessed June 20, 2022)

[10] University of Bozen-Bolzano. Ontop: A Virtual Knowledge Graph System. https://ontop-vkg.org/ (accessed June20, 2022)

[11] Eclipse Foundation. Eclipse RDF4J. https://rdf4j.org/ (accessed June 20, 2022)

[12] W3C Recommendation. Time Ontology in OWL. https://www.w3.org/TR/owl-time/ (accessed June 20, 2022)

[13] Camunda. Zeebe: Cloud Native Workflow and Decision Engine. https://camunda.com/platform/zeebe/ (accessed June 20, 2022)