Aalto University

Kabugo, James Clovis; Jämsä-Jounela, Sirkka Liisa; Schiemann, Robert; Binder, Christian

Industry 4.0 based process data analytics platform

Industry 4.0 based process data analytics platform: A waste-to-energy plant case study

James Clovis Kabugo[a], Sirkka-Liisa Jämsä-Jounela[a,*], Robert Schiemann[b], Christian Binder[b]

[a] *Aalto University School of Chemical Engineering, Kemistintie 1, 02150 Espoo, Finland*
[b] *Outotec GmbH, Gwinnerstrasse 27-33, 60388 Frankfurt, Germany*

ABSTRACT

Industry 4.0 and Industrial Internet of Things (IIoT) technologies are rapidly fueling data and software solutions driven digitalization in many fields notably in industrial automation and manufacturing systems. Among the several benefits offered by these technologies, is the infrastructure for harnessing big-data, machine learning (ML) and cloud computing software tools, for instance in designing advanced data analytics platforms. Although, this is an area of increased interest, the information concerning the implementation of data analytics in the context of Industry 4.0 is scarcely available in scientific literature. Therefore, this work presents a process data analytics platform built around the concept of industry 4.0. The platform utilizes the state-of-the-art IIoT platforms, ML algorithms and big-data software tools. The platform emphasizes the use of ML methods for process data analytics while leveraging big-data processing tools and taking advantage of the currently available industrial grade cloud computing platforms. The industrial applicability of the platform was demonstrated by the development of soft sensors for use in a waste-to-energy (WTE) plant. In the case study, the work studied data-driven soft sensors to predict syngas heating value and hot flue gas temperature. Among the studied data-driven methods, the neural network-based NARX model demonstrated better performance in the prediction of both syngas heating value and flue gas temperature. The modeling results showed that, in cases where process knowledge about the process phenomena at hand is limited, data-driven soft sensors are useful tools for predictive data analytics.

1. Introduction

The global competition in industrial manufacturing, for instance to increase productivity, product quality [1], process safety in addition to economic and environmental sustainability has led to the development of modern process monitoring and data analytics systems. The current developments concerning industrial internet of things (IIoT) technologies, machine learning algorithms and big-data availability provide platforms for the realization of sophisticated process data analytics. There are several cloud computing platforms available for use in industrial internet of things and big-data analytics. Key players include cloud service providers (like Microsoft Azure, Amazon Web Services, IBM, Intel etc.), enterprise solution vendors (notably PTC and Oracle), networking companies (such as Cisco and AT&T) and industrial engineering companies (namely General Electric, Siemens and Bosch), among others. A number of these platforms are available under proprietary licenses with a few others being accessible as open source

projects.

In predictive data analytics in process industry, soft sensors are invaluable tools for providing insights into the state of process operations especially in cases where the direct measurement of key process variables is extremely difficult, nearly impossible or even unreliable [2,3]. For instance, in waste-to-energy (WTE) plants knowing the heating value of solid fuel is a very important aspect for smooth plant operations, which often leads to fewer fluctuations in the power generated and a longer lifetime for the plant equipment. However, due to the heterogeneous nature often associated with source materials of solid fuel, most notably composition, the real-time quantification of the heating value of solid fuel is considerably difficult. Moreover, with stringent environmental regulations, WTE plants are constrained to control particulate and gaseous emissions, such as the NO_x content in the flue gas effluent. Some researchers [4] have reported studies concerning the use of data-driven soft sensors as an alternative tool for the estimation of gaseous emissions in combustion processes. Soft sensors

can be developed by implementing mathematical process models, for instance, those based on mass and energy balances or the application of data-based statistical models as well as the use of more sophisticated machine learning algorithms and hybrid systems taking advantage of both methodologies. Soft sensors offer an affordable alternative to physical measuring devices and real-time prediction of process variables and can be easily utilized in parallel or integrated with the measuring equipment.

In the recovery of energy from biomass, a few researchers have developed soft sensors to predict the heating value of biomass solid fuel. For instance, Kortela et al. [5,6] reported a soft sensor to evaluate the heating value of the biomass material fed to the BioGrate furnace based on the oxygen requirement in the combustion unit process. The same authors also estimated the outlet temperature of flue gas after the combustion reaction using the mass and energy balances around the combustion chamber [6]. Belkhir and Frey [7,8] also presented a soft sensor to estimate the heating value of the wood based biomass solid fuel in a combustion process similar to the earlier work [5]. In their work, the virtual sensor for the solid fuel heating value was derived from the energy balance of the furnace under steady state conditions, where the model can also be used to predict the temperature of flue gas after combustion. However, flue gas temperature can also be derived from the energy balance around the boiler [9].

Data-driven soft sensors are also widely applied in different process industrial operations [10]. Recently, the remarkable performance of machine learning (ML) methods particularly deep learning based algorithms in areas such as pattern recognition, computer vision and robotics, offer other options to realize more sophisticated data-based soft sensors. Such soft sensors are capable of leveraging industrial big-data, leading to enhanced variable prediction. In combustion processes, deep learning based soft sensors have been reported for the prediction of key variables in the form of gaseous components. Such variables are difficult to measure directly during process operations. For instance, the addition of oxygen to the combustion process is a critical variable for efficient fuel combustion. In this respect, a number of researchers have developed deep learning based soft sensors to predict the oxygen content in flue gases exiting the combustion process units [2,11]. Elsewhere, deep learning algorithms have been applied in soft sensors in the estimation of NO_x emission from a coal-fired furnace [12] and in the quantification of heat outflow in a step-grate boiler [13]. Besides data-driven soft sensors based on deep neural networks, other commonly known machine learning algorithms or statistical methods are often employed in modeling process phenomena in biomass combustion processes. For instance, Ögren et al. [14] applied Gaussian process regression and a shallow artificial neural network to develop a vision-based soft sensor for the prediction of syngas composition from the gasification process. In similar works, Pandey et al. [15] and Xiao et al. [16] reportedly used multi-layer feed forward neural networks in the modeling of lower heating value (LHV) of syngas and other gasification products like tars and char, in the gasification process of municipal solid waste. The process phenomena in biomass gasification processes have also been modeled using dynamic neural network models [17], notably the neural network-based nonlinear autoregressive with exogenous input (NN-NARX) model [18].

The concept of industry 4.0 and its benefits for industrial data analytics is highlighted in this paper by developing a unique platform based on the latest IIoT software tools and ML algorithms for industrial use in process monitoring. The platform emphasizes the use of state-of-the-art ML methods, big-data tools and industrial internet cloud computing platforms for the development of robust data-driven soft sensors among other data-driven models. The platform further highlights the use of IIoT technologies in data acquisition and the employment of statistical methods for data pre-processing, soft sensor development, real-time process monitoring and offline data analytics. Moreover, data-driven soft sensors for estimating syngas heating value and flue gas temperature for a WTE industrial case study were developed and their

performance was compared with the corresponding overall steady state energy balance models. This work contributes to the existing knowledge of process data analytics in modern process automation systems by emphasizing the use of readily available open source or proprietary software tools to develop sophisticated data analytics platforms for industrial process applications. Furthermore, the data analytics platform presented here contributes to scientific research aimed at the realization of smart integrated solid waste management systems from waste collection to WTE technologies within the framework of industry 4.0.

2. State-of-the-art platforms for process data analytics

Currently, a number of hardware and software tools for handling process data analytics are available from industrial automation vendors and other computing services vendors. These tools are increasingly being used to develop data analytics platforms for different process and manufacturing industries; one notable example is the Outotec® ACT platform [19]. Therefore, in this Chapter, a brief review of the existing industrial internet of things, machine learning and big-data software platforms are discussed from the perspective of process data analytics. Moreover, with the use of different state-of-the-art software tools, a platform for process data analytics in the modern process automation system is demonstrated.

2.1. Description of the process data analytics platforms

The platforms presented in this work, follow the general data-based process monitoring procedures, which include data acquisition, data pre-processing, model design and model maintenance. The data acquisition layer allows measurements from different local devices in the process to be collected for purposes that may include data inspection or visualization, analysis and storage. In industrial processes, data acquisition can be accomplished through a number of interfaces, for instance, the use of OPC UA, OPC, Modbus and various network protocols like MQTT, CoAP, REST and HTTP. Currently, various internet of things (IoT) platforms can be implemented for data collection from industrial devices, notably through edge computing and IoT cloud getaways. The choice of a data acquisition interface also influences the possibility of real-time process monitoring. Thus, in cases where high-latency behavior in data acquisition dictates, only offline or batch data analysis are appropriate. The most common steps applied in data pre-processing are data visualization, selection of relevant datasets, outlier removal, scaling and data filtering. Other additional data pre-processing measures may include resampling, delay estimation and compensation, data transformations and dimensionality reduction. Model design involves the selection of an appropriate model followed by model training, validation and testing. The steps involved in data pre-processing and model design criteria are implemented using various statistical methods. The choice of the appropriate method in each case depends on the nature of the process phenomenon at hand among other factors. Fig. 1 describes the key components of the proposed data-driven soft sensor development environment, which takes advantage of the state-of-the-art machine learning algorithms, big-data tools and industrial cloud computing platforms. Details about the key components of the platform such as data acquisition, Industrial IoT platforms and ML software tools, among others are discussed below.

2.2. Data acquisition

In data acquisition, measurements from different local devices in the process are collected for purposes that may include data inspection or visualization, analysis or storage, among others. In industrial process applications, data acquisition is achieved via several interfaces for instance the use of OLE Process Control Unified Architecture (OPC UA), OPC, Modbus and through various supported network protocols.
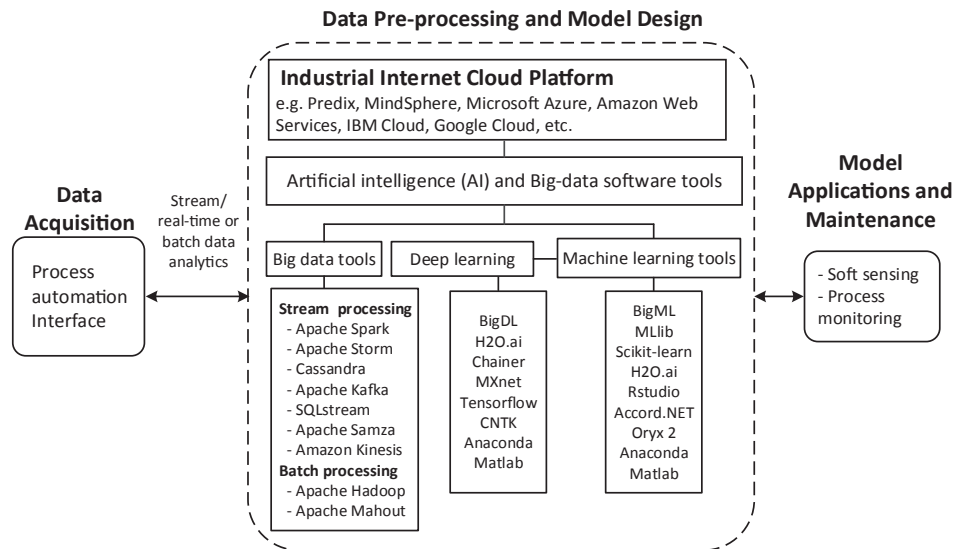
Fig. 1. A schematic representation of a monitoring platform for the development of data-based soft sensors using available machine learning and big-data software tools in a cloud computing environment.

Particular data acquisition environments support real-time and/or batch data collection and data analysis. For example, in case of real-time process monitoring, industrial automation interfaces like OPC UA, OPC, Modbus and MT Connect are applicable. Data from smart devices like sensors can be acquired through network protocols such as MQTT, CoAP, HTTP and several others. In the industrial internet of things (IIoT) architectures, IoT getaways are employed for data acquisition from smart devices, control systems and so forth [20].

2.3. Industrial IoT cloud platforms

The availability and use of low-cost sensors have enabled modern industrial equipment, machines and other devices to generate large amounts of data. However, to leverage such data, it is important for industrial plants to integrate their operations with the digital world. Digitalization makes it easier to acquire, link and manage the raw-data before it can be thoroughly analyzed [21]. Nowadays, several commercial grade industrial IoT platforms suitable for industrial automation are available, notably Predix, MindSphere [21] and Sentience cloud platforms. In addition, there are many multi-purpose IoT cloud platforms, which are available from cloud service providers, for instance, Amazon web services, Microsoft Azure, Google cloud and IBM's Watson IoT. Most of these platforms offer distributed computing, big-data analytics, data and device management tools, machine-to-machine interaction and many other functionalities. The connectivity of devices to IoT cloud platforms can be achieved in different ways, for instance, through "plug and play" [21], use of open communications standards for industrial automation notably the OPC Unified Architecture (OPC UA) and network protocols like MQTT. Cloud computing platforms also provide large and affordable data storage capacity and flexibility for client demands, plus offering highly scalable computing power. Therefore, they are able to accommodate industrial set-up workloads. In addition, industrial grade IoT platforms emphasize secure device connectivity as well as cyber-security. The security aspects are among the key elements of the industrial cloud computing platform [22].

2.4. Edge computing

Edge computing [23] is gaining notable attention for the realization of real-time data analytics on premises. This is expected to lead to more efficient process monitoring than with the use of current plant automation systems. With edge computing, the conventional plant automation architecture can be by-passed and the plant devices are directly connected to the industrial IoT cloud from the edge for instance, by using a 5G [24] communication network. The edge interface offers real-time data processing including data filtering and basic data analytics, in addition to machine-to-machine communication (M2M), all at the plant premises and near the data sources. This means that fast process monitoring can be achieved on premises with the help of soft sensing models and anomaly detection models, among others. However, edge computing is only possible if the process devices in question have built-in capabilities, hardware and software to do so. Nonetheless, currently, several industrial IoT vendors like Siemens and Honeywell are developing devices, which are capable of supporting edge computing. Mobile edge computing (MEC) is among the currently available edge computing technologies within the industrial IoT.

2.5. Machine learning and big-data

There are several machine learning software tools available both commercial and open source software. Popular commercial machine learning software is provided as machine learning as a service (MLaaS). The leading proprietary MLaaS include Amazon machine learning (Amazon ML), Microsoft Azure machine learning, Google machine learning and IBM Watson machine learning. Often, machine learning tools are accessible through respective cloud platforms or under third party cloud application services. They offer highly scalable environments and a variety of machine learning algorithms for data pre-processing, dimensionality reduction, predictive data analytics and plenty of other functionalities [25].

Moreover, most of the commercial machine learning frameworks also provide deep learning libraries and big-data computing software tools like Apache Spark, Hadoop, etc. In addition, there are several machine learning software tools, libraries and frameworks that are freely accessible under open source licenses. Also, these can be implemented through either the cloud environment or on the premises. Examples of open source machine learning software tools are Apache Spark MLlib, Scikit-learn, TensorFlow, H2O.ai, BigML, Accord.NET, Apache SystemML, Apache Mahout, Oryx 2, just to mention but a few. Some of these libraries can be accessed through proprietary machine learning platforms. For instance, both Apache Spark MLlib and H2O.ai are available in Microsoft Azure HDInsight, a big-data analytics cloud service platform [26].

On the other hand, a number of commercial deep learning software

platforms are available from several technology firms. Examples include IBM PowerAI platform, NVIDIA DGX-1 Software Stack, Intel Nervana platform, Skymind intelligence layer and so forth. There are also many available open source deep learning frame works and libraries that offer different deep learning models for big-data analytics. Notable examples are H2O.ai, Tensorflow, Chainer, MXnet, Keras, BigDL, Microsoft Cognitive toolkit (CNTK) and many more.

Most of the industrial automation cloud platforms such as Predix and MindSphere offer application services for data analytics. For instance, MindSphere currently provides built-in application programming interfaces (APIs) for outlier and anomaly detection [27]. Apart from the built-in APIs, in many cases, clients have the opportunity to deploy their own applications into the cloud platform. Therefore, more efficient machine learning algorithms can be implemented in data analysis. Furthermore, general cloud service providers like Microsoft Azure [26] and Amazon Web services provide machine learning and deep learning services and big-data software tools like Apache Spark, Hadoop, Apache Storm and so forth.

2.6. Process automation and industrial IoT technologies

In addition to the existing process monitoring platforms within the process automation system, modern technologies (including edge and IoT cloud computing) can be utilized for process data analytics. The combination of state-of-the-art industrial automation systems coupled with industrial IoT technologies is expected to become the norm for industrial automation according to the proponents of industry 4.0 [1]. As demonstrated in Fig. 2, data acquisition from devices can be done through the plant automation system and via edge computing environments. With the help of IoT cloud gateways, streaming data or historical data is ingested into the industrial IoT cloud platform. Industrial IoT vendors normally provide IoT cloud getaway connections. For instance, Siemens's MindSphere offers MindConnect Nano and MindConnect IoT 2000 getaways, which ensure secure information sharing. On the other hand, as indicated in Fig. 1, there are a number of data streaming frame works like Apache Storm and Apache Spark, which can be employed for real-time stream data analytics. Besides, depending on the IoT platform applied, proprietary stream analytics platforms such as Azure Stream Analytics and Amazon's Kinesis, are also available, as illustrated in Fig. 1. In some instances, streamed data

is collected and stored in cloud storage data centers such as data lakes or SQL databases, among others, to archive or to carry out batch data analytics using batch big-data tools alongside ML algorithms. Industrial IoT cloud platforms like Siemens' MindSphere [21] also provide a number of applications, which may include visualization dashboards for stream jobs, reports, managing workflows back to the process, digital twins and ML data analytics tools as well as offering an interface for custom application development. For instance, in the Azure Stream Analytics query [26], JavaScript user-defined functions can be implemented when executing complex computations.

3. A waste-to-energy plant case study

A waste-to-energy plant was studied here as an example of the application of the above proposed IIoT-based data analytics platform. In this case study, data-based soft sensors for the prediction of syngas heating value and hot flue gas temperature in a biomass WTE process were developed.

Biomass is one of the current sources of renewable energy [28–30]. The other popular renewable energy sources include hydropower, solar, wind and geothermal power. The availability and accessibility of the major renewable energy sources tend to vary according to geographical locations. For instance, cold regions normally experience short periods of sunlight, which makes the feasibility of solar power generation in such areas to be low in comparison with those in warm environments. In addition, the availability of some renewable energy sources like wind and solar is naturally intermittent. Hydropower is considered the most frequently used renewable energy in many countries. China is the world leader in hydropower generation due to its abundant hydropower resources [28]. However, weather changes, like long periods of drought, normally affect the availability of hydropower. In addition, the construction of hydropower dams tends to have negative environmental impacts on the surrounding natural ecosystems. On the other hand, among the advantages of biomass over the other renewable energy sources include its availability: it is readily available in all countries in different types, such as agricultural waste, wood residues and municipal solid waste, among others. Furthermore, biomass energy can be readily stored for later use, for instance, in the form of biofuel. Also biomass energy is considered a reliable renewable energy source compared with either wind or solar. For this reason Tajeddin and Roohi [30] proposed
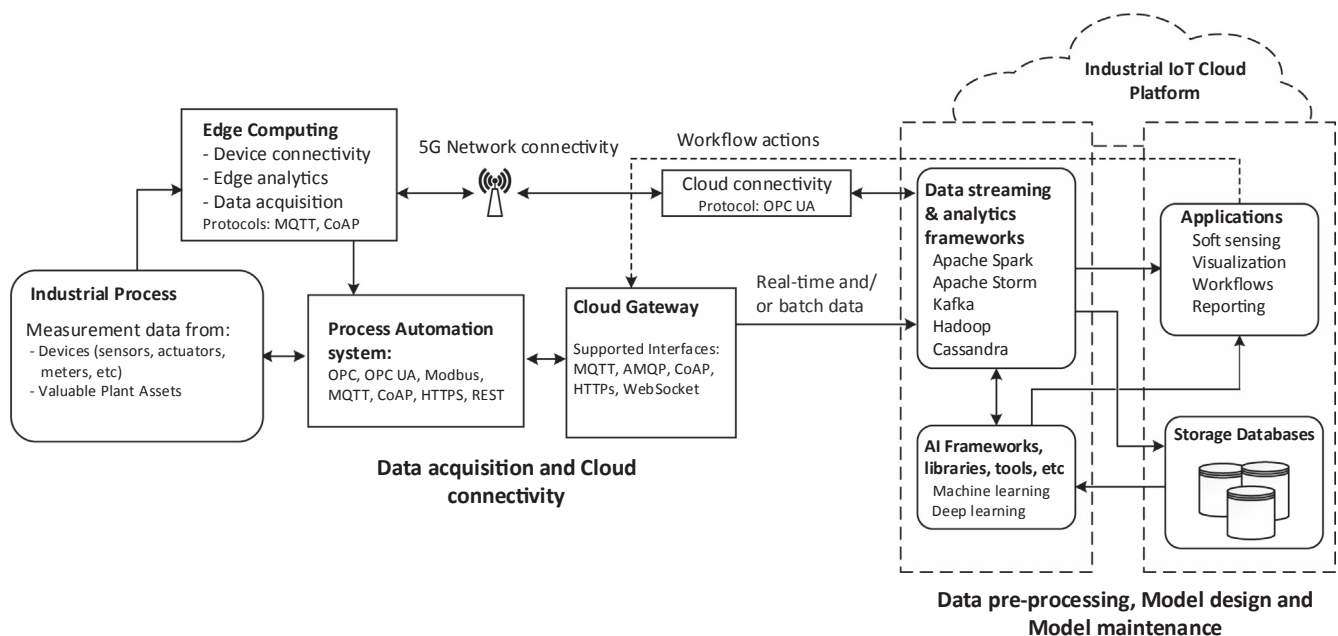

Fig. 2. A detailed process monitoring platform highlighting the use of edge and cloud computing technologies in industrial process automation and data analytics.

a wind-biomass hybrid system to improve the reliability of a wind farm. There are also constraints concerning the utilization of biomass as a renewable energy source. For example, the challenges in waste collection, pollutant emissions like NO_x and SO_2, high investment costs, the risk of deforestation and so on.

With the increasing global population, urbanization and industrialization, waste generation is also rapidly increasing around the world, leading to global health and environmental concerns [31]. Hence, waste management systems, such as WTE technologies, are being increasingly implemented to address the arising problems, including global warming and climate change, through sustainable energy recovery and environmentally friendly waste disposal. In general, WTE technologies fall under biochemical, physicochemical and thermochemical waste treatment methods [32,33]. Nowadays, waste management through the use of thermal treatment processes with energy recovery has become popular in the US, Europe and East Asia and is rising in the other parts of the world. Thermal treatment technologies can be categorized as pyrolysis, gasification and incineration processes [31]. Waste-to-energy plants usually treat different kinds of solid waste materials, which may include municipal solid waste (MSW), industrial sludge, sewage sludge, agricultural waste, wood waste, plastic waste and so forth. The heterogeneity of solid fuel, health and environmental constraints, profitability, among other limitations, make the design and operations of the WTE plant considerably complex. For instance, the complexity of thermal treatment-based WTE plants can be attributed to the application of more efficient pyrolysis, gasification, combustion and flue gas cleaning technologies [34].

Municipal solid waste is one of the major sources of solid fuel for WTE plants especially in countries with large urban centers [32,35]. Currently, incineration processes are widely applied in the thermal treatment of municipal solid waste [36]. In today's municipal solid waste incineration (MSWI), moving grate, fluidized bed and rotary kiln incinerators are employed. Among these, moving grate incinerators are the most commonly used due to their ability to handle higher throughputs even without prior shredding and sorting [31,33]. In addition to heat and energy recovery in combined heat and power plants, WTE plants also employ sophisticated technologies for flue gas cleaning to satisfy emission control policies. In Europe and in countries like Estonia, Denmark, Sweden and Finland, WTE by incineration accounts for over 50% of the municipal solid waste. In East Asia, thermal WTE plants are also becoming popular, particularly in countries like China, Japan and South Korea [31]. For instance, according to Bourtsalas et al. [34], in South Korea, a total of 35 WTE plants contribute about 8% and potentially 0.6% of the country's district heating and electricity demands, respectively.

In the literature, there are not many cases where IIoT, big-data analytics and cloud computing technologies have been implemented in solid waste management systems, especially in WTE processes. However, within this research area, a few researchers have investigated the use of internet of things (IoT) platforms for enabling waste collection in the so-called smart cities [37,38]. For example, Aazam et al. [38] proposed a cloud-based waste management system, enabled by built-in smart sensors in waste bins. With the smart sensors, the level of the waste bins can be monitored and the collected data uploaded into the cloud platform from where it's accessible to the stakeholders for analysis, including waste collection route optimization and planning. Moreover, with smart waste bins, edge computing can be implemented [39]. In a similar work, Anagnostopoulos et al. [37] reviewed IoT technologies for the real-time monitoring of waste collection, transportation and disposal. In addition, they conceptualized a waste management system based on three key components: the physical infrastructure, the IoT platform and the software analytics tools. The physical infrastructure in here refers to things like waste bins, trucks, depots and dumping sites and so on. Whereas the IoT technology is enabled by sensors, actuators, wireless sensor networks (WSNs), Radio Frequency Identification (RFID) tags, Near Field Communications

(NFC), global positioning systems (GPS), cameras and so forth. Software analytics tools, provide algorithms that are applicable in modeling dynamic scheduling and routing problems and a decision support system used by stakeholders among others things. More recently, Popa et al. [40] reported an IoT-based automated waste collection system (AWCS) built using Microsoft Azure IoT cloud. With Azure IoT Hub, measurement data from different devices is connected to the Microsoft Azure cloud. Numerous functionalities provided by Microsoft Azure, such as Azure stream analytics, Azure machine learning, logic apps and data storage can be utilized. For example, Azure logic apps can be used for the anomaly detection of the AWCS and event reporting. The use of IoT platforms in solid waste collection has also been investigated elsewhere, including studies concerning state-of-the-art IoT architectures [41] and algorithms for the optimization of waste collection operations [39].

According to the above-mentioned short review of the most recent research concerning the application of industry 4.0 technologies in waste management, it can be seen that the emphasis has been put on the development of efficient waste collection systems. Using IIoT technologies, from waste collection to WTE plants, still requires more research.

3.1. Process description

In this process, solid waste fuel is fed into the Outotec Advanced Staged Gasifier, where it is combusted to produce heat. The heat generated is used to produce steam for electric power generation. The operations of this particular WTE plant are summarized in Fig. 3. The main process unit is the Outotec Advanced Staged Gasifier, which is divided into two parts, the lower and upper sections. In the lower section, gasification takes place in a bubbling fluidized bed where a sand bed and solid waste particles are fluidized by air. The gasification stage produces syngas, which is directed to the upper section for combustion. Any metallic components and large-sized particulates are removed by discharging part of the bed material through the bottom cone of the fluidized bed. However, after cleaning, the bed material is recycled back to the gasifier. In the upper section of the Outotec Advanced Staged Gasifier, the combustion of syngas is carried out in the presence of air. Reagents for NO_x reduction, such as ammonia are added in the upper section. Such reagents react with NO_x compounds to form harmless nitrogen and water. The flue gas typically exits the gasifier at a temperature of 930 °C. Heat is recovered from the hot flue gas stream using the boiler to produce super-heated steam for electric power generation. After the boiler, flue gas undergoes a series of cleaning stages as well as further heat recovery measures before it is discharged into the environment. At the same time, during flue gas cleaning, subsequent solid particulates in the form of ash are collected for disposal or for further treatment and use.

3.2. Monitored key process phenomena

The study was aimed at developing data-based soft sensors to quantify the heating value of syngas and hot flue gas temperature for a WTE plant. The heating value of syngas generated after the gasification of solid fuel is difficult to quantify directly during process operations in the absence of the required measurements, which are challenging and expensive. A data-driven soft sensor in that case would be very useful to predict the syngas heating value online. The heating value of syngas is among the key performance indicators of this particular WTE process. On the other hand, the combustion chamber outlet temperature affects various downstream process operations, including the turbine power output and the emissions of harmful gaseous compounds. Therefore, a good prediction model for the flue gas temperature at the outlet of the combustion chamber would also be useful in process control strategies, process optimization and process monitoring. For instance, a soft sensor for flue gas temperature would be applicable in fault detection if the
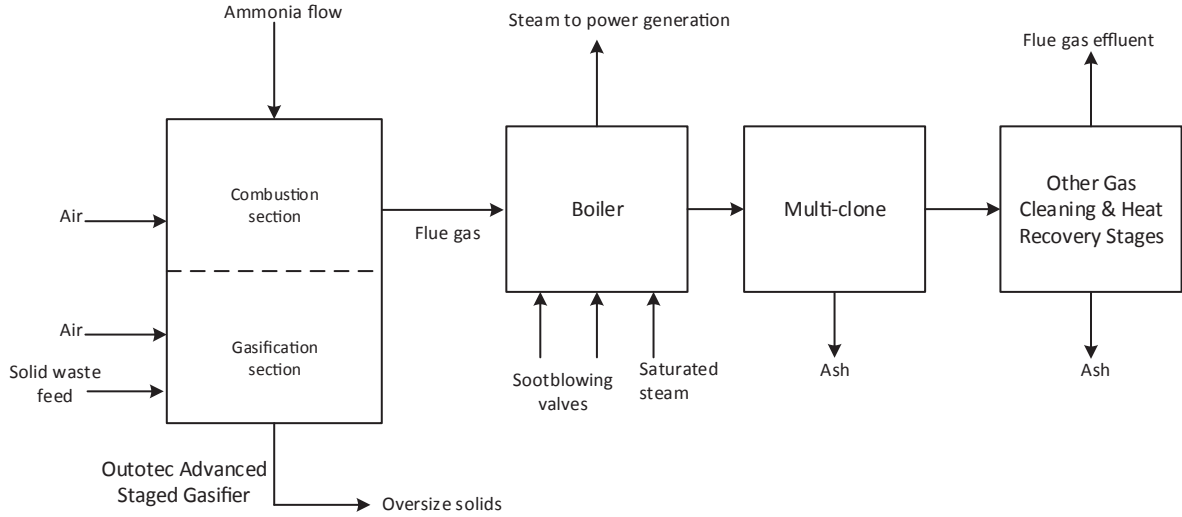
Fig. 3. A simplified process scheme for a WTE plant showing relevant streams and variables studied in this present work.

temperature sensor and soft sensor outputs disagree too much and hence, create alarms.

3.2.1. Syngas heating value

The heating value of syngas can be estimated based on the energy balance around the syngas combustion chamber or it can be determined from the combustion reactions [7,42–44]. Hence, a suitable model can be selected depending on the availability of the required measurements. Here, the overall steady-state energy balance of the combustion section is described by Eq. (1), where h_v (kJ/kg) is the heating value of syngas (enthalpy of combustion reactions), m_s is the syngas mass flow (kg/s), Q_a is the air heat flow (kJ/s), Q_s is the syngas heat flow (kJ/s), Q_f is the flue gas heat flow (kJ/s) and Q_l collectively represents the heat losses (kJ/s).

$$Q_a + Q_s + m_s.h_v = Q_f + Q_l \tag{1}$$

Besides nitrogen, the primary components of syngas normally include CO, CO_2, H_2, CH_4 and H_2O. It is important to note that Eq. (1) was developed as a simple but useful physical model based on the available process knowledge. In practice, syngas heating value will vary depending on the solid fuel flow and the amount of air added to the gasification process.

3.2.2. Flue gas temperature

There is a restriction on the temperature inside the syngas combustion chamber with a minimum required value of 850 °C, required and the gas retention time above that temperature must be at least two seconds, according to the 'EU Waste Burning Directive' [45]. This requirement is important for controlling the gaseous emissions, including volatile organic compounds. According to the process operations, the hot flue gas temperature can be estimated from the energy balance around the gasifier and the boiler. Considering a stationary energy balance around the waste heat boiler, the hot flue gas temperature $T_{f,in}$ can be estimated according to Eqs. (2) and (3).

$$Q_f = m_f (h_{f,in} - h_{f,out}) = m_{st} h_{st,out} - m_w h_{w,in} + Q_l \tag{2}$$

$$T_{f,in} = \frac{m_{st} h_{st,out} - m_w h_{w,in} + Q_l}{m_{f,in} c_{p f,in}} + \frac{h_{f,out}}{c_{p f,in}} \tag{3}$$

where, m is the mass flow (kg/s), h is the stream enthalpy (kJ/kg), Q is the stream heat flow (kJ/s), c_p is the specific heat capacity (kJ/kg/K) and subscripts, f, st, w and l represent flue gas, live steam, boiler feed water and heat losses respectively. However, due to the complex process dynamics surrounding the operation of the boiler, the

determination of flue gas temperature using energy balance models is notably difficult. For example, in this case study, some part of the heat flow used to evaporate the water is not transferred in the waste heat boiler but in the combustion chamber itself. Hence, from a modeling point of view, the energy balance model in Eq. (2) is not a proper representation of reality. Moreover, in this case, the heat losses Q_l were unknown, and thus, Eq. (3) was modified. The heat losses were estimated as a fraction of the heat flow of live steam as follows: $Q_l = (1 - \eta)Q_{st}$, where η is the boiler efficiency. This is a rough estimate, which was assumed in order to test the performance of the steady-state energy balance model against data-driven models. During the testing of the energy balance model, boiler efficiency was assumed to be high, above 90%.

3.3. Data-driven models

Kumar et al. [46] and other previous authors [47,48] extensively reviewed a number of ML algorithms and their use in the development of data-driven models. Either individual ML methods or their hybrids are often employed when solving industrial data analytics problems. The selection of a suitable method for a particular case, often depends on the nature of the data analytics task in addition to other technicalities associated with the data handling and its processing. Siow et al. [49] in their work about internet of things based data analytics platforms, classified the nature of possible data analytics problems into five categories, which include descriptive, diagnostic, discovery, predictive and prescriptive data analytics. The soft sensors considered in the present industrial use case, fall under predictive data analytics. In predictive data analytics, the models assist in understanding the future behavior of the system based on historical data and regression based ML methods [50] are used to develop such predictive models.

Data-driven predictive models find wide applications in different industrial operations for example in soft sensing, process monitoring and quality prediction tasks. From the viewpoint of using ML algorithms to model operations in industrial power plants and other energy-related systems, several research works have been reported in literature. For instance, Tüfekci [50] applied fifteen different machine learning regression methods using WEKA software to predict the full load electrical power output of a particular power plant. In that case, a decision tree based regression method performed better than the rest. Moreover, as a concrete industrial example, Fast and Palmé [51] reported the online use of an artificial neural network (ANN) model to predict turbine power output in a combined heat and power (CHP) plant. The ANN model was found to accurately predict the system

behavior.

There are several ML regression algorithms, which could be used to develop predictive models. These include variants of linear regression methods, support vector regression (SVR), Bayesian networks regression, variants of artificial neural networks (ANNs), decision trees regression, random forest regression, k-nearest neighbor (K-NN), principal component analysis (PCA) based methods, adaptive boosting method and so on [46–48]. Again most of these machine learning algorithms are available through several ML software frameworks and libraries with examples presented in Fig. 1. For instance, Ahmed et al. [52] predicted and forecasted energy consumption for a smart grid system using support vector regression, linear regression and artificial neural network methods with the help of the IBM Cloud machine learning tools. The selection of the appropriate ML algorithm is often based on data characteristics, model complexity and prior knowledge about the system behavior, among other criteria [53]. For instance, if the system exhibits strong non-linearity, then the selection of highly nonlinear models such as ANNs and SVR algorithms would be appropriate. Alternatively, for dynamic systems, either nonlinear or linear, dynamic models would be required and so on.

In this work, the ML methods studied comprised of linear methods, which included multivariable linear regression (MLR), principal component regression (PCR) and partial least squares regression (PLSR) and a nonlinear method, the neural network-based NARX model. The linear methods, which were selected are often used in process industry to develop soft sensors and also applied in process monitoring [53]. On the other hand, the NN-NARX model is suitable for modeling highly nonlinear and dynamic system behaviors. The neural network-based NARX model was also selected because of its capability to handle tasks concerning the forecasting of time series data for dynamic systems. The selected methods were used to predict the heating value of syngas and the hot flue gas temperature and the performance of each method in comparison with the other methods was assessed.

3.3.1. MLR, PCR and PLSR models

A multivariable linear regression (MLR) model can be demonstrated by Eq. (4) [54], where Y is a n-dimensional response vector, X is a design matrix of predictor variables $(x^{(1)}, \cdots, x^{(m)})$ with dimensions of $n \times p$ (and $p = m + 1$), θ is a p-dimensional vector containing estimated parameters and ε is an n-dimensional vector of estimated error. The dimensions n and m correspond to the number of samples and variables respectively. The superscript T in all equations denotes the transposition of the respective matrix.

$$Y = \theta X + \varepsilon, \quad where \quad \theta = (X^T X)^{-1} X^T Y \tag{4}$$

The performance of the regression model is often assessed based on the estimated coefficient of determination, R^2 and root mean squared error (RMSE) according to Eq. (5) and Eq. (6), respectively. In the following equations, y_i is the observed output variable, $\widehat{y_i}$ is the predicted output variable and $\bar{y}_i$ is the mean of the output variable.

$$R^2 = 1 - \frac{\sum_{i=1}^{n} (y_i - \widehat{y_i})^2}{\sum_{i=1}^{n} (y_i - \bar{y}_i)^2} \tag{5}$$

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n} (y_i - \widehat{y_i})^2}{n}} \tag{6}$$

Unlike in the MLR model, which directly uses the input variables to predict the output variable, the principal component regression (PCR) and partial least squares regression models, apply new predictor variables called components to predict the output variable. However, parameter estimation of the PCR and PLSR models is done in the same manner as in the MLR model described in Eq. (4).

In the case of the PCR model, the new predictor variables are obtained after principal component analysis (PCA) on the original dataset of input variables. The PCA method describes the linear relationship

between variables in a dataset and maps them to a set of linearly uncorrelated variables. The derived uncorrelated variables are referred to as principal components or scores and these are obtained without using the output variables. The first principal component represents data with the largest variations, followed by the component with the second largest variations and so forth. The PCA method is briefly illustrated mathematically below, considering a dataset of independent variables represented as a matrix X of dimensions $n \times p$, for which the covariance matrix C is calculated. The covariance matrix can be determined using the singular value decomposition (SVD) method according to Eq. (7),

$$C = \frac{1}{n-1} X^T X = P \Lambda P^T \tag{7}$$

where P is the loading matrix of size $p \times p$ and $\Lambda = diag(\lambda_1, \lambda_2, \cdots, \lambda_p)$ is a diagonal matrix containing the eigenvalues of matrix C presented in the descending order. Usually, just a few principal components are enough to describe a given dataset of independent variables according to Eq. (8) [55].

$$X = T V_m^T + E \tag{8}$$

where T is the X-scores matrix, V_m is comprised of m eigenvectors, which describe most of the variations in the input data and E is the residual matrix. Therefore, only a certain number of principal components are retained for parameter estimation of the PCR model. There are several techniques available to handle this task and among them is the cumulative percentage of variance (CPV) method [56], which was applied in this work.

The partial least square method is also a commonly used statistical method in building predictive models, for instance, in cases where there are many variables or in the presence of strong collinear behavior between variables [57], among others. Moreover, the PLS model can also help address the over-fitting problem, which maybe encountered if the MLR model was applied instead. Like PCA, the PLS method extracts latent factors (or scores) responsible for the observed variation in the dependent variables. The ultimate target is to apply the extracted factors to predict future responses. Unlike in the PCA method, the PLS method applies both the original predictor and response variables to obtain predictor scores and output scores, respectively. In the PLSR method, latent variable matrices, T and U, corresponding to the predictor variables matrix, X and the response variables matrix Y, respectively, are extracted. The extracted factors, T (X-scores) are utilized to predict U latent variables (Y-scores). The obtained Y-scores are then used to make predictions of output variables. The decomposition of X and Y variables to determine T and U latent variables follows Eq. (9) and Eq. (10), respectively.

$$X = T P^T + E \tag{9}$$

$$Y = U Q^T + F \tag{10}$$

where P and Q are orthogonal matrices of regression coefficients, also referred to as loadings, and E and F are error matrices with respect to X and Y variables. The decomposition of X and Y variables is done so that the covariance of T and U is maximized. The variables X and Y are related according to Eq. (11) [57],

$$Y = T B Q^T + F \tag{11}$$

where B is a diagonal matrix composed of the weights of regression. The appropriate number of latent variables to be retained in the PLS model can be estimated using a cross-validation algorithm.

3.3.2. Neural network NARX model

In this work, an artificial neural network-based nonlinear autoregressive model with external input (NN-NARX) was implemented. Equation error models similar to the NN-NARX mode are widely known in the prediction and control applications [58]. Among the advantages

of this method over the previous methods include its applicability in dynamic and non-linear systems as well as in time series forecasting. The NARX model estimates the observed response based on the previous input and output variables as described in Eq. (12).

$$y(t)$$
$$= f[y(t-1), \cdots, y(t-n_a), \cdots, u(t-n_k), \cdots, u(t-n_k-n_b+1)] + e(t)$$
(12)

where $y(t)$ is the output variable at time t, f is the nonlinear function, $u(t)$ is the input variable at time t, n_a is the number of poles, n_b is the number of zeros plus unit, n_k is the delay with respect to the input variable and $e(t)$ is a white noise disturbance. Initially, the non-linear function f is unknown. However, it is approximated in the training phase of the feedforward neural network model. The architecture of the neural network NARX model is comparable to the shallow feedforward multilayer perceptron (MLP) model. In the feedforward neural network model, the output from the jth neuron in the hidden layer is calculated following Eq. (13) [59].

$$z_j(t_k) = f_j\left(\sum_{i=1}^{p} w_{ji}(t_k)x_i(t_k) + b_j\right)$$
(13)

where p is the number of input layer nodes, x_i is a vector of input variables, w_{ji} is the weight associated with the connection of ith input node to the jth hidden node and b_j is a bias term also known as the activation threshold for the jth hidden node. The function f_j is an activation function for the hidden layer nodes and is usually a sigmoid function, such as a logarithmic sigmoid function and a hyperbolic tangent function. The final outputs from the output layer neurons are then computed according to Eq. (14) [59],

$$v_l(t_k) = f_l\left(\sum_{j=1}^{q} w_{lj}(t_k)z_j(t_k)\right)$$
(14)

where q is the number of nodes in the hidden layer, w_{lj} is the weight associated with the connection of the jth hidden node to the lth output node in the output layer and f_l is a linear activation of the lth output node.

The neural network NARX model can be implemented using open-loop or a closed-loop architecture [60]. The open-loop architecture favors online process monitoring, whereas with the closed-loop architecture, future process outputs can be estimated offline, for example, through process simulation [18]. The two architectures of the NARX model, which were employed in this work, are demonstrated in Fig. 4.

After model training using optimization algorithms like back-propagation training methods, the performance of the NN-NARX model is evaluated based on the results of the loss function. For this work the performance of the neural network NARX model was monitored using the mean squared error (MSE) method, which is described in Eq. (15) [61].

$$MSE = \frac{1}{n}\sum_{i=1}^{n} (y_i - \widehat{y_i})^2$$
(15)

where y_i is the measured output variable, $\widehat{y_i}$ is the predicted output variable and n is the number of measurements.

4. Results and discussion

4.1. Experimental data and testing environment

The data used in the modeling of syngas heating value and flue gas temperature was obtained from the Outotec process simulator and it represented a period of three months of plant operation with a one-minute time interval, equivalent to about 126,500 sample data points. For the MLR, PCR and PLSR models, data was split manually into two sets, for model training and testing at 80% and 20%, respectively. In the

case of the neural network model, the data was randomly divided into three parts, for training, validation and testing at 70%, 15% and 15%, respectively. All the work, including data pre-processing and modeling was conducted using MATLAB 2018a software. The neural network toolbox in MATLAB was utilized in the implementation of the NN-NARX model. The results reported represent the performance of the individual data-based model on the respective test data set.

4.1.1. Data pre-processing

Raw data pre-processing is an important step before using the data for model development. This is because raw data such as process data is often associated with noise, disturbances, uncertainties, measurement errors, different orders of magnitude in different signals and so forth, which tend to affect the results obtained after data analysis. Therefore, the use of appropriate data pre-processing methods helps to improve the quality of data thereby enhancing the accuracy and efficiency of data analysis methods. In this work, data pre-processing involved, the removal of outliers, filtering and data scaling by normalization.

In the detection and removal of outliers, the Hampel identifier method was implemented. This method is known to be quite effective in the removal of outliers from datasets [62]. Moreover, with this technique, it is not necessary to know the distribution of the data in advance, which is often important in the selection of a suitable method for outlier removal. In the next step, data filtering was done using a median filter of the seventh order. This was found to be appropriate in terms of minimizing noise in the data and avoiding the loss of important information from it. Due to varying data scales, data standardization was carried out before further data analysis. Data scaling is a common procedure before applying most of the ML algorithms, such as principal component analysis (PCA), partial least squares (PLS) and artificial neural networks, among others. Data standardization prevents data analysis methods from being biased toward the variables that exhibit large variances.

4.1.2. Variable selection for the prediction of syngas heating value and flue gas temperature

The selection of predictor variables was done based on data correlation analysis and process knowledge. First, data correlation analysis was performed on all available process variables around the gasifier. The correlations between the independent and dependent variables were examined from the correlation matrix, and highly correlated independent variables against the target dependent variables were selected for further analysis. Process knowledge was used to eliminate independent variables, which were strongly correlated with the dependent variables but were already known as poor predictor variables according to process operations.

The original independent variables and calculated variables, which exhibited stronger correlation behavior against the output variables, syngas heating value and flue gas temperature, were used in the development of the respective prediction model. For instance, in flue gas temperature prediction, calculated variables derived based on the ideal gas equation, that is, the product of the volumetric flowrate of air and the corresponding air pressure were utilized in the model development. On the other hand, computed variables based on the relationship between volumetric flowrates and respective valve openings were also employed in the modeling of syngas heating value. In this case, the product of the volumetric flow of air and the extent of the valve opening was found to be a good predictor variable after correlation studies. The use of computed variables in data-driven models offers several advantages, which may include capturing some of the non-linear behavior of the system at hand, indicating faults in variable measurements and understanding system disturbances, among others [55]. The original and computed variables used in the data-based prediction models are shown in Table 1.
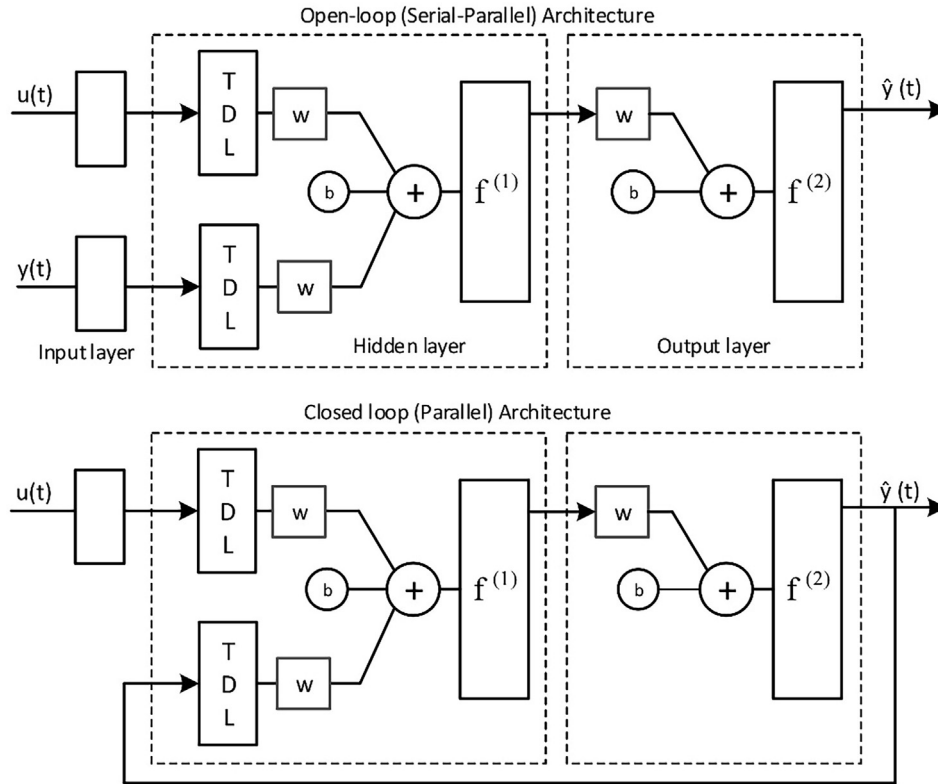
Fig. 4. The architecture of the NARX model applied in this work: TDL denotes tapped delay lines, w is a respective input or output weight, b is the bias term and $f^{(1)}$ and $f^{(2)}$ are activation functions in the hidden and output layer, respectively.

Table 1
List of the investigated variables and their descriptions.

Variable	Description
x_1	Syngas temperature
x_2	Fluidized bed inlet air volumetric flow
x_3	Combustion chamber inlet air volumetric flow
x_4	Fluidized bed air valve opening
x_5	Combustion chamber air valve opening
x_6	Product of the combustion chamber air volumetric flow and valve opening
x_7	Product of the fluidized bed air volumetric flow and valve opening
x_8	Combustion chamber inlet air pressure
x_9	Solid fuel feed valve opening
x_{10}	Product of the combustion chamber air volumetric flow and pressure
x_{11}	Boiler feed water
y_1	Syngas heating value
y_2	Combustion chamber outlet flue gas temperature

Fig. 5. A typical convergence plot observed for the NN-NARX model showing the behavior of MSE during model training, validation and testing with respect to the number of epochs (while using the LM training algorithm).

4.1.3. Training of the NN-NARX model

The NN-NARX model, which comprised of an input layer, one hidden layer and an output layer, was implemented all through the test work. The input and feedback delays were varied, as well as the number of nodes in hidden layers as part of the tuning procedure of the neural network parameters. Moreover, in the training of the model, the performance of different training algorithms, including the Levenberg-Marquardt (LM) algorithm and the Bayesian regularization (BR) algorithm, among others were compared before the selection of the appropriate method. The training of the model was limited to a maximum of 1000 epochs. However, in most cases, model convergence was achieved after about 100–500 epochs, as demonstrated in Fig. 5. After training, validation and testing the model in the open-loop (series-parallel) architecture, a closed–loop architecture was adopted to examine the performance of the trained model in time series forecasting. The training o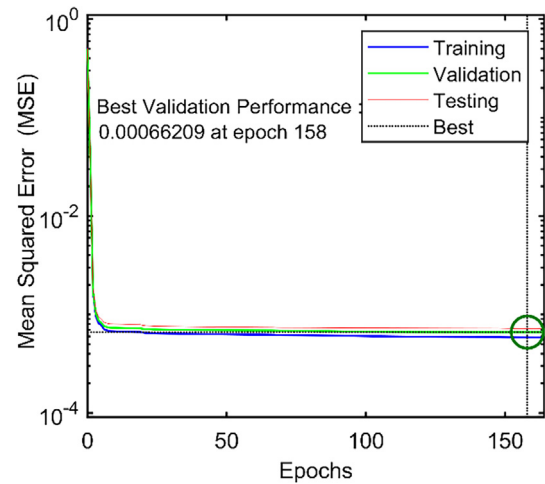f the model in the closed-loop (parallel) architecture was done utilizing a separate dataset, which was originally reserved for this purpose. After the closed–loop NARX model, the forecasted output values were compared with the actual expected output values. The performance of the NARX model was assessed based on the mean squared error (MSE) and monitoring the outputs of the error autocorrelation and the input-error cross-correlation functions in MATLAB.

4.2. Prediction of syngas heating value

In the prediction of syngas heating value, seven input variables (x_1, x_2, x_3, x_4, x_5, x_6, x_7) from Table 1 were retained in the building of the prediction models. These variables showed a significant correlation
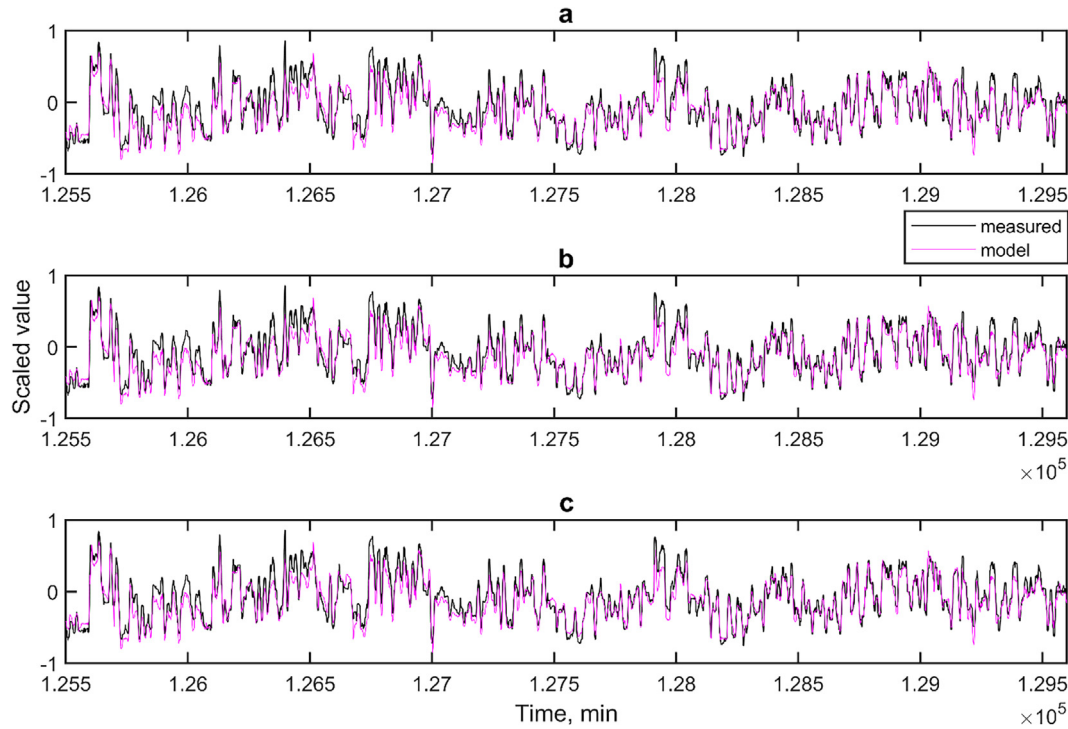
Fig. 6. Modeling of syngas heating value in a WTE plant using data-driven regression methods: (a) with the use of multivariable linear regression, (b) using principal component regression and (c) for partial least squares regression.

with the syngas heating value and were selected as the predictor variables. The performance of the trained MLR, PCR and PLSR models on the test dataset is shown in Fig. 6, which shows that the three models were comparable in the prediction of the syngas heating value. The MLR model showed a correlation coefficient R^2 of 0.868, whereas the PCR and PLSR models both showed a value of 0.870. However, based on the measure of the percent variance explained in the predicted output, the PCR and PLSR models required at least six components out of seven to predict the syngas heating value fairly well. Otherwise, model prediction error increases with the use of a lower number of component variables. Thus, the use of either the PCR or PLSR model was not so effective in this particular case.

First, in the application of the NARX model, the parameters of the neural network were chosen after investigating multiple alternative parameters and evaluation of the model performance. For instance, the number of neurons in the hidden layer, as well as the time delays for the input variables and the feedback output variable were varied while using the LM training algorithm. Based on the results in Table 2, a neural network architecture with 10 neurons and delays of 7 min was adopted. The selection was based on the observed mean squared error values for each scenario, accounting for other performance measures such as the time consumed in the model training, error autocorrelation and the input-error cross-correlation behaviors. For instance, it was

ensured that the results of the error autocorrelation and the input-error cross-correlation functions in MATLAB were within the recommended 95% confidence limits. Otherwise, the model was re-trained until acceptable results were obtained.

The neural network NARX model showed better prediction accuracy of the measured values of output variables compared to all the other methods studied. Fig. 7 shows the performance of an open-loop NARX model in the prediction of the syngas heating value. An average value of mean squared error (MSE) of 0.0021 and a regression correlation coefficient, R^2 of 0.988 were observed after model testing. In this model, the LM training algorithm was implemented. Other training algorithms were also studied under similar conditions, in which time delays of 7 min and one hidden layer of 10 nodes were employed. The performance of the model on the test dataset in each case was monitored. In Table 3, the results of the four training methods were comparable based on the mean squared error values. Only the adaptive learning rate gradient descent method showed a relatively higher prediction error among the five studied methods. In this case, the LM and BR methods both performed slightly better than the rest. However, only the former was selected for further use because the latter required more computation time for training the model, as seen in Table 3.

The neural network NARX model was also tested to forecast the syngas heating value. Since syngas heating value is not continuously

Table 2
Performance results of the neural network NARX model in the determination of the appropriate time delays and number of nodes within the hidden layer (after five parallel experimental runs).

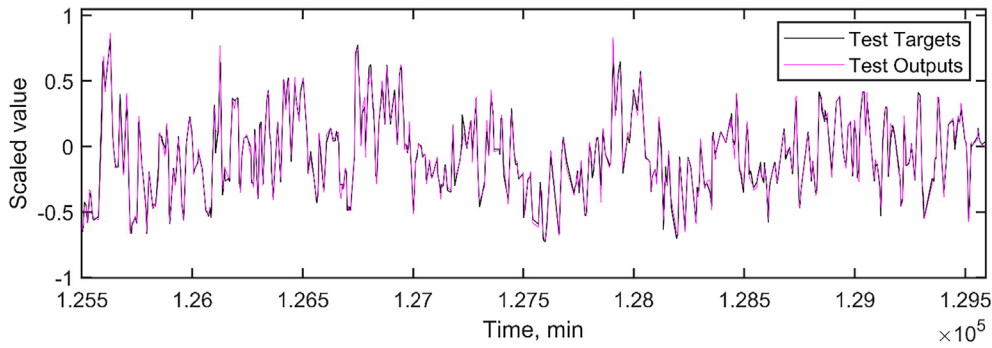| Hidden layer number of nodes	Time delays, min	Mean squared error (MSE)		Regression coefficient, R^2		Average training time, min
		Mean value	Standard deviation	Mean value	Standard deviation	
7	7	0.00209	3.19E-05	0.9879	2.76E-04	1.52
10	7	0.00205	1.67E-05	0.9881	3.07E-04	2.35
14	7	0.00202	1.76E-05	0.9880	1.82E-04	4.23
10	5	0.00207	5.16E-05	0.9878	2.54E-04	2.61
10	10	0.00203	9.57E-06	0.9880	1.40E-04	3.75

Fig. 7. Prediction of syngas heating value for a WTE plant using a neural network NARX model.

Table 3
Evaluation of the performance of different algorithms in the training of the neural network NARX model (after five parallel experimental runs).

| Training algorithm	Mean squared error (MSE)		Regression coefficient, R^2		Average training time, min
	Mean value	Standard deviation	Mean value	Standard deviation	
Levenberg-Marquardt	0.00205	1.67E-05	0.9881	3.07E-04	2.35
Bayesian regularization	0.00204	1.80E-05	0.9881	2.07E-04	11.37
Scaled conjugate gradient	0.00224	3.49E-05	0.9868	3.02E-04	0.49
BFGS quasi newton	0.00221	2.71E-05	0.9872	2.90E-04	0.61
Adaptive learning rate gradient descent	0.00577	6.83E-04	0.9667	3.49E-03	0.09
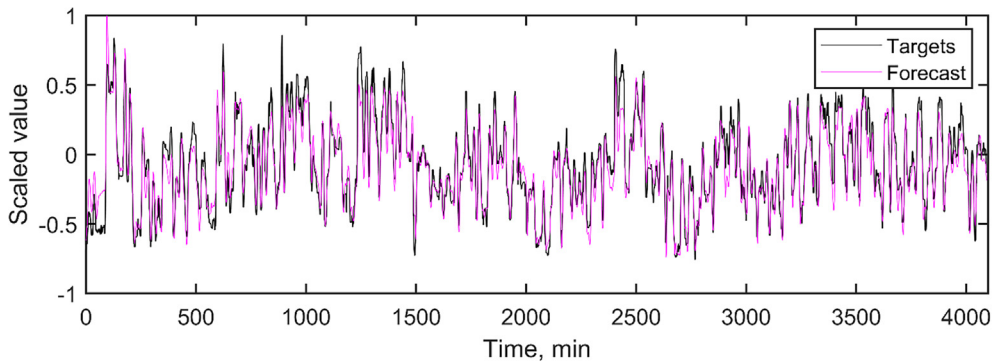
Fig. 8. Forecasting of syngas heating value in a WTE plant using the neural network NARX model.

measured during the process operations, such time series forecasting would be useful in monitoring the performance of the combustion process. Forecasting was carried out by switching the trained open-loop neural network to the closed-loop mode. With the new set of input data for only the predictor variables, the closed-loop network was used to predict the corresponding output values. The predicted output was compared with the target output, which in this case was already known. Fig. 8 shows the results obtained after configuration of the model to forecast syngas heating value. The model performed considerably well in the forecasting of the heating value of syngas for more than two days' operation with an average value of a mean squared error of 0.0120 and a regression coefficient R^2 as high as 0.931.

4.3. Prediction of flue gas temperature

The temperature of flue gas exiting the combustion chamber was modeled using seven predictor variables (x_2, x_3, x_4, x_8, x_9, x_{10}, x_{11}) from Table 1. Similarly to the previous case, the predictor variables were selected based on their significant correlations with the output variable. Fig. 9 demonstrates the performance of the MLR, PCR and PLSR models against the test data. Looking at the results in Fig. 9, again, the three models performed fairly and comparably to each other with regression correlation coefficients R^2 of about 0.89, 0.87 and 0.90 for the MLR,

PLSR and PCR models, respectively. In this case, the PCR and PLSR models were developed using five components out of seven beyond which no significant improvement in the model prediction capacity was noticed. The results showed that any of these models could be used for this particular purpose, even though the PLSR method appeared to perform slightly better than the rest.

The open-loop NARX model performed quite well in the prediction of hot flue gas temperature, as illustrated in Fig. 10. In this case, the observed average values of the mean squared error (MSE) and corresponding correlation coefficient R^2 after model testing were 6.23×10^{-4} and 0.998, respectively. The settings of the neural network model including the network architecture, time delays and training algorithm remained the same as in the previous case.

Again, in order to forecast the flue gas temperature, the trained model from the open-loop architecture was transformed into a closed-loop form and a new dataset of predictor variables was used to forecast the respective output values. The results are shown in Fig. 11, where the forecasted flue gas temperature is compared with the target output values, which were already known prior to forecasting. As illustrated in Fig. 11, the model forecasted the temperature of hot flue gas for the studied future period of about two days fairly well. The average values of the observed mean squared error and correlation coefficient, R^2 were 0.0204 and 0.936, respectively. Although the determined mean squared
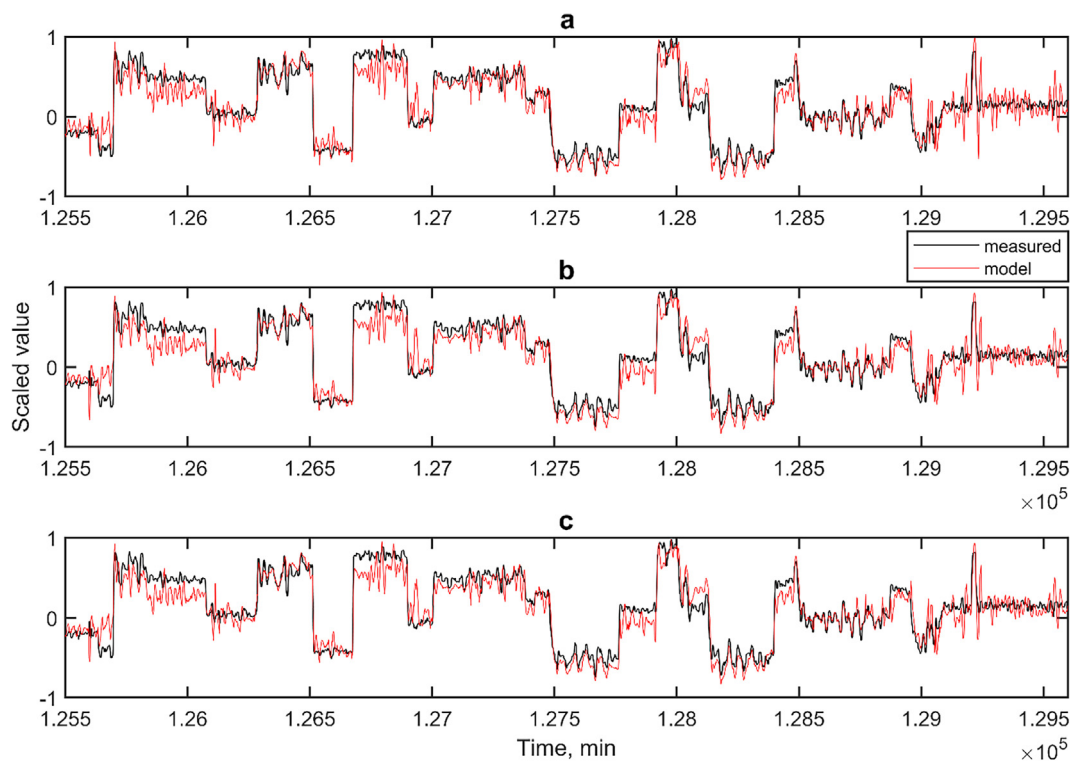
Fig. 9. Modeling of hot flue gas temperature at the exit of the combustion chamber in a WTE plant: (a) multivariable linear regression, (b) principal component regression and (c) partial least squares regression.
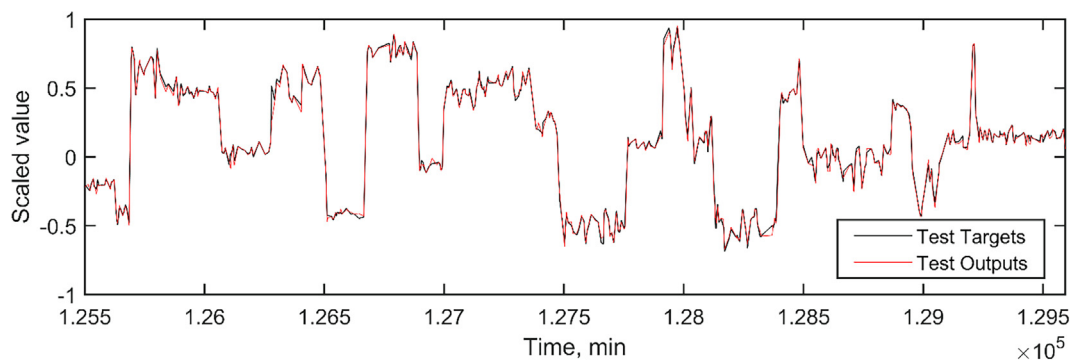

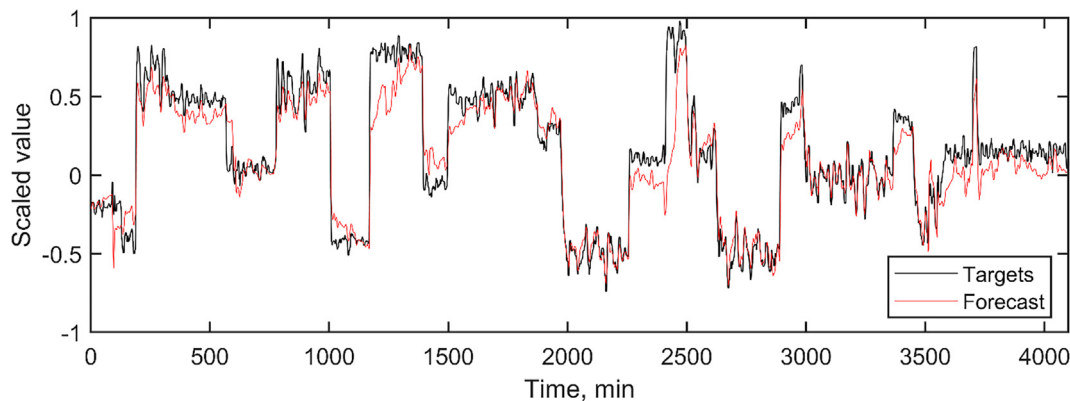**Fig. 10.** Prediction of hot flue gas temperature in a WTE plant using a neural network-NARX model.

Fig. 11. Forecasting of hot flue gas temperature in a WTE plant using a neural network NARX model.
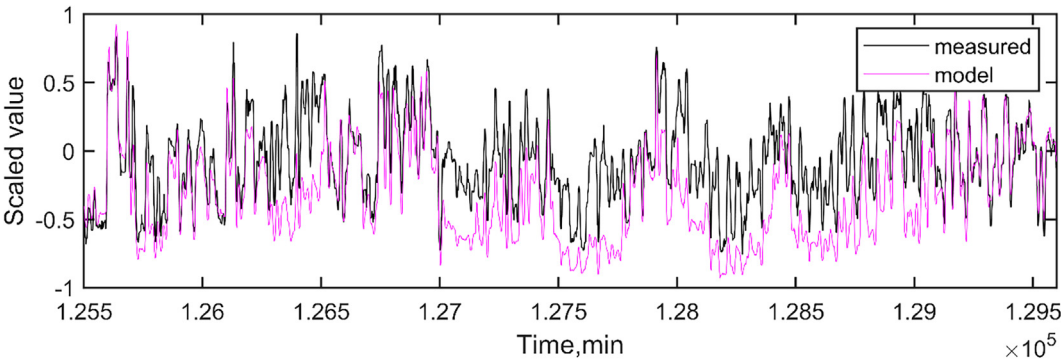
Fig. 12. Modeling of syngas heating value in a WTE plant using a steady state energy balance.
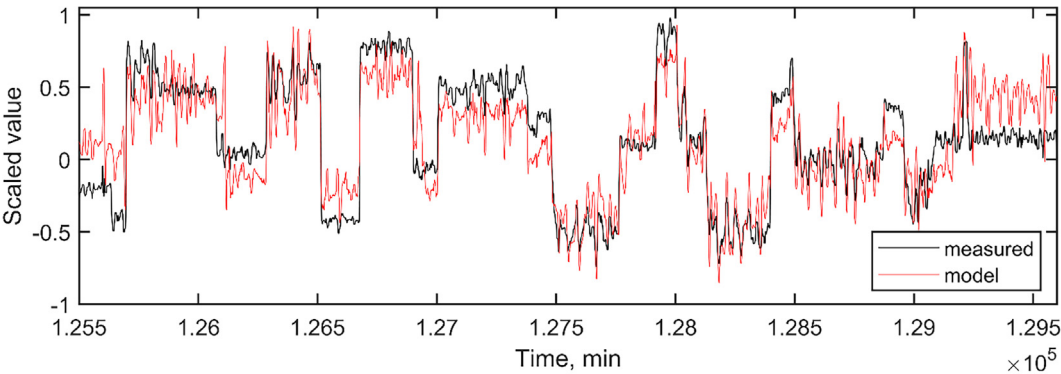

Fig. 13. Prediction of hot flue gas temperature in a WTE plant using a steady state energy balance-multivariable linear regression model.
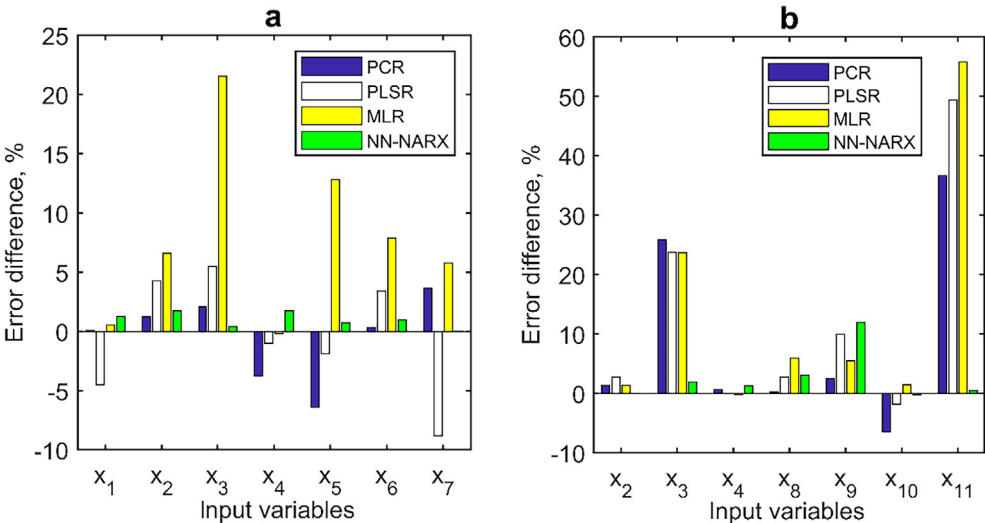

Fig. 14. Sensitivity analysis showing the relative importance of each independent variable in different data-driven models in the prediction of: (a) syngas heating value and (b) hot flue gas temperature.

Table 4
Model performance in the prediction of syngas heating value.

Test case	Root mean squared error (RMSE)				Regression coefficient, R^2			
	MLR	PLSR	PCR	NN-NARX	MLR	PLSR	PCR	NN-NARX
Case 1	0.098	0.098	0.098	0.045	0.872	0.872	0.872	0.988
Case 2	0.098	0.114	0.116	0.046	0.870	0.814	0.806	0.988
Case 3	0.156	0.156	0.156	0.047	0.415	0.415	0.415	0.987

Table 5
Model performance in the prediction of hot flue gas temperature.

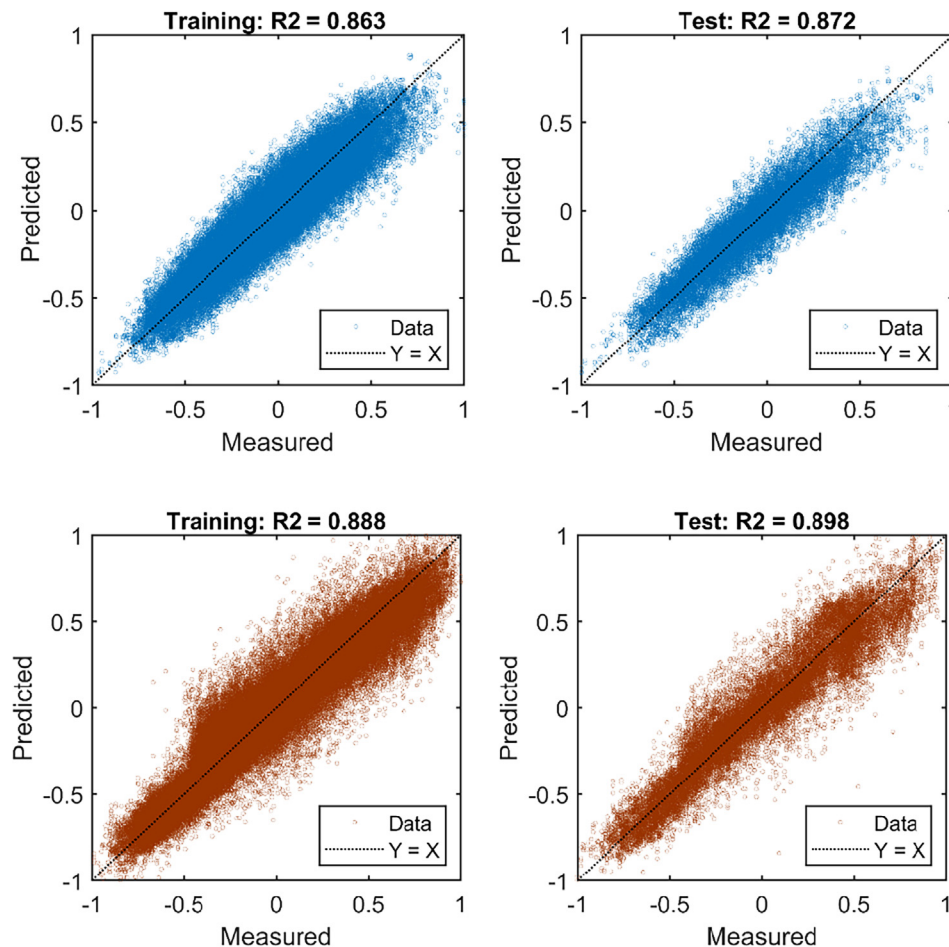| Test case	Root mean squared error (RMSE)				Regression coefficient, R^2			
	MLR	PLSR	PCR	NN-NARX	MLR	PLSR	PCR	NN-NARX
Case 1	0.126	0.131	0.133	0.0249	0.898	0.889	0.884	0.998
Case 2	0.138	0.139	0.157	0.0253	0.869	0.865	0.815	0.998
Case 3	0.182	0.180	0.181	0.0251	0.688	0.686	0.658	0.998

Fig. 15. Performance of the MLR model in Case 1: (above) syngas heating value and (below) hot flue gas temperature.

error increased in the forecasting of flue gas temperature compared to the case without future output prediction, in general the performance of the NARX model was quite good.

4.4. Comparison of steady state energy balances and data-based models

In this section, steady-state energy balances described in Eqs. (1) and (3) were tested and their results compared to the performance of data-driven models in Section 3.3. The heating value of syngas was determined according to Eq. (1). It was assumed that the major components of the syngas mixture flow were N_2, CO, H_2, CO_2, CH_4 and H_2O. Nitrogen was considered the most abundant due to the air addition of air into the gasification process. Therefore, the composition of the syngas mixture was assumed to be 70 wt% N_2, 16 wt% CO, 6.5 wt% H_2, 6.0 wt% CO_2 with CH_4 and H_2O amounts of < 0.1 wt% in each case. These assumptions were based on the literature studies of different biomass-derived syngas and on rough estimates for the average proportion of each individual component in syngas [14]. On the other hand, the flue gas composition was assumed to be mainly CO_2, O_2 and

H_2O. The content of oxygen in the flue gas was known. Thus only the H_2O was assumed to be < 0.1 wt%, and the CO_2 composition was calculated. The enthalpies of the gaseous streams were determined with the help of the Outotec HSC Chemistry software thermodynamic database.

The behavior of the calculated syngas heating value against the simulated process value is presented in Fig. 12, which shows that the energy balance model predicted the heating value of syngas gas fairly well. However, its performance was relatively low as compared to the data-based model results, especially when using the neural network NARX model. Nevertheless, considering the fact that syngas composition was not known, together with the amount of solid fuel fed into the gasifier, it can be said that the model performance was quite reasonable. Moreover, the energy balance model did not account for any heat losses around the combustion chamber due to a lack of enough information. Furthermore, since only the steady-state model was applied here, any dynamic behavior of the process could not be sufficiently described by the model.

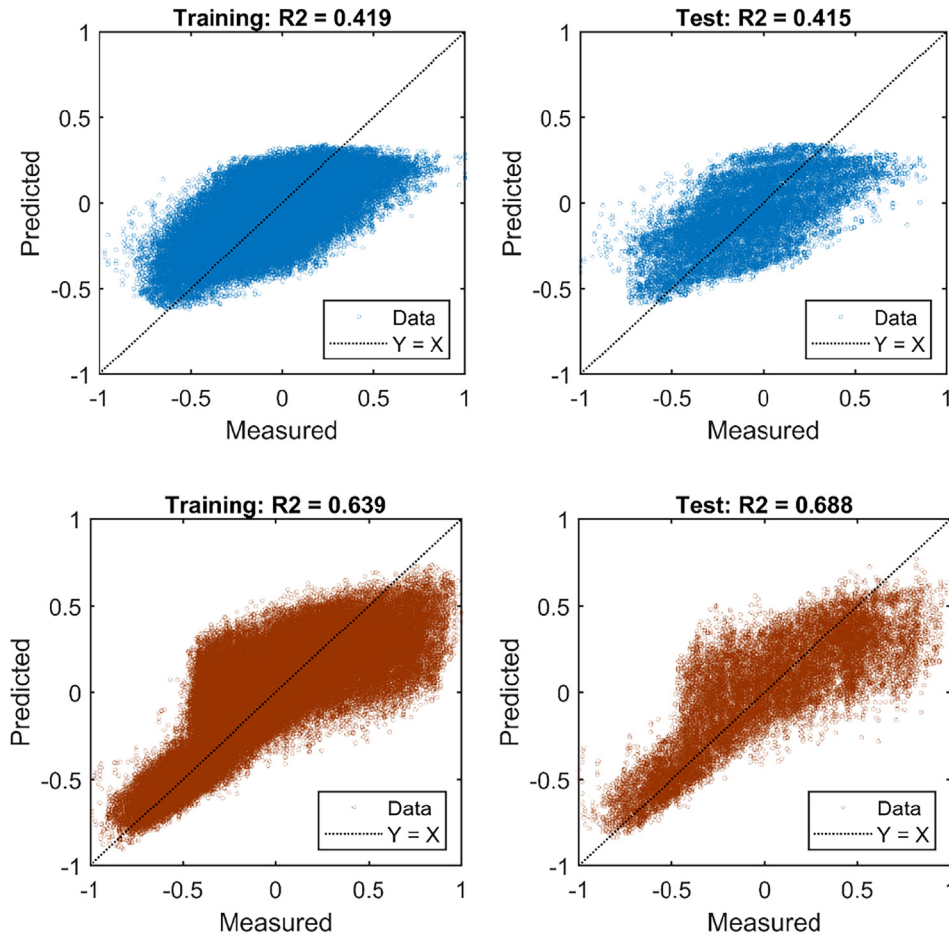The prediction of flue gas temperature using the steady-state model

Fig. 16. Performance of the MLR model in Case 3: (above) syngas heating value and (below) hot flue gas temperature.

described in Eq. (3) could not yield good results due to the severe influence of heat losses and could not be modeled well. Therefore, to improve the performance of the energy balance model, a multivariable linear regression method was applied after rearranging Eq. (3) into two calculated variables, one of which represented the roughly estimated heat losses. Here, the MLR method was implemented to improve heat loss estimation and to account for the uncertainties in the calculated variables. Temperature prediction improved with a regression correlation coefficient R^2 of 0.75 versus 0.54 for the actual energy balance model. However, temperature prediction was still relatively low, as seen in Fig. 13, indicating that the energy balance model applied was insufficient in describing the studied process. Therefore, a more robust process model similar to the one discussed in [44] would be required to describe the process dynamics involved in the operations of the syngas combustion and boiler unit processes. Furthermore, it can be said that, in this case, the data-driven models performed very well in relation to the applied steady-state energy balance model.

4.5. Sensitivity analysis

Sensitivity analysis was performed to establish the most important independent variables with respect to the applied data-driven prediction model. There are several sensitivity analysis methods, which can be employed to study the relative importance of input variables in prediction models, as demonstrated by previous authors [14,18,63,64]. In this particular work, the stepwise technique [64] was utilized in a backward manner. First, all variables in the original dataset were used and the respective prediction error (RMSE) was recorded. The procedure was repeated with only one independent variable eliminated from

the model. This was done with respect to all input variables. To quantify the sensitivity to each variable, the difference in RMSE observed between the model with all input variables and a model corresponding to a particular eliminated independent variable, was calculated. The error difference was expressed as a fraction as described in Eq. (16).

$$\varepsilon_d = \frac{\varepsilon_{n-i} - \varepsilon_n}{\varepsilon_n} \tag{16}$$

where ε_d is the error difference as a fraction, ε_{n-i} represents the RMSE when input variable i is eliminated from the model and ε_n is the RMSE corresponding to the model with all the originally selected n input variables. The open-loop NN-NARX model was re-trained in each case at least five times after the initial run. The results are presented in Fig. 14, where the relative importance of each independent variable is described by the percent error difference calculated from Eq. (16).

In the prediction of syngas heating value (Fig. 14a), variables x_1 and x_4 were observed to be less important in the design of PCR, PLSR and MLR models, and one of these variables can be removed from each model without significant loss in model performance. Again, for these three models, variables x_2 and x_3 showed a significant contribution to model prediction performance in each case. The sensitivity analysis of input variables in the NN-NARX model in syngas modeling generally showed that each variable exhibited small individual contributions within the model. For instance, either variable x_3 or x_7 could be eliminated from the model and still maintain very good model prediction results. Considering the modeling of flue gas temperature (Fig. 14b), variables x_2, x_4 and x_{10} showed low to no importance within the models. Variables x_3 and x_{11} showed strong influence in the PCR, PLSR and MLR models but far less contribution within the NN-NARX
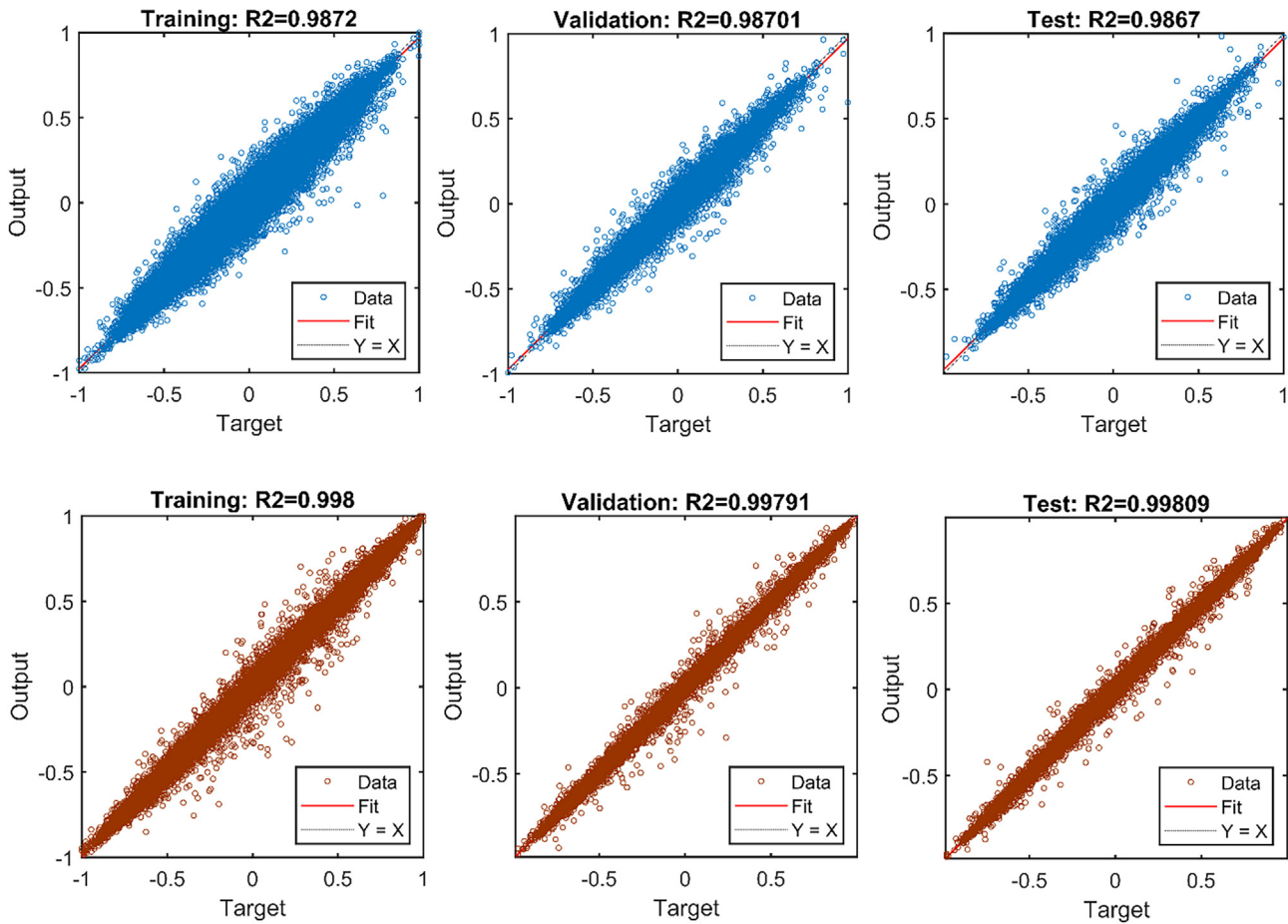
Fig. 17. Performance of the NN-NARX model in Case 3: (above) syngas heating value and (below) hot flue gas temperature.

model. However, in all models, variable x_9 was observed to play a significant role. To summarize, sensitivity analysis studies revealed that some input variables could be eliminated from models due to their insignificant positive contribution to the overall performance of the respective model. Furthermore, it was observed that the dynamic NN-NARX model was far less sensitive to the individual input variables, and in all cases model prediction remained high compared with the other models. Therefore, this implies that fewer input variables could be applied while using the NN-NARX model and still maintain high prediction accuracy. The following section further quantifies this observation.

4.6. Performance evaluation of the selected data-driven methods

According to the sensitivity analysis results in Fig. 14, it was observed that the number of the selected predictor variables could be reduced with respect to the data-driven method. Therefore, in this section the number of input variables has been reduced accordingly and the performance of each prediction model was evaluated. Three scenarios were considered, Case 1, Case 2 and Case 3, respectively. The first scenario represents the original sets of predictor variables for each dependent variable as shown in Fig. 14. The second scenario represents a reduced number of input variables, which included x_2, x_3, x_5, x_6 and x_7 for syngas heating value prediction and x_3, x_8, x_9 and x_{11} in the prediction of flue gas temperature. In the third scenario, variables x_1, x_2, and x_4 were used to predict syngas heating value while variables x_3, x_8 and x_9 were applied in the prediction of flue gas temperature. The performance of each model was evaluated according to the root mean squared error (RMSE) and the regression coefficient R^2. The results after model testing are summarized in Tables 4 and 5, for syngas

heating value and flue gas temperature, respectively. For the NN-NARX model, average values from five experiments after model training are given. In all scenarios and for both output variables, the NN-NARX model consistently showed high prediction accuracy. On the other hand, the MLR, PLSR and PCR models performed poorly when Case 3 was implemented and especially in the prediction of syngas heating value as observed in Table 4.

However, in the second scenario, Case 2, the MLR, PLSR and PCR model, also displayed good prediction performances and the results were comparable to those achieved when using the originally selected variables, in Case 1. Though it was observed that, the effectiveness of using either the PLSR or PCR model depreciated as fewer number of input variables were applied. Nevertheless, as shown in the Tables 4 and 5, the three models, MLR, PLSR and PCR showed comparable results in all cases. To further justify the better performance exhibited by the NN–NARX model when compared with the rest of the studied data-driven methods, detailed results for the MLR and NN-NARX models in Case 1 and Case 3 are presented. Fig. 15 shows the results of the MLR model in Case 1. As previously seen in Sections 4.2 and 4.3, both syngas heating value and flue gas temperature were modeled fairly well. Although the model showed a reasonably good prediction of flue gas temperature, both the training and test datasets still showed a notable model bias. In Case 3, the MLR model poorly predicted the heating value of syngas and also showed significant bias in the estimation of the flue gas temperature according to Fig. 16.

In contrast, as shown in Tables 4 and 5, the performance of the NN-NARX model remained consistently high and similar in all cases. Therefore, only the results obtained in Case 3 are presented here and compared with the MLR model. The results of the MLR model also fairly represents the performance of the PLSR and PCR model in all cases.

Fig. 17 shows the results for the NN–NARX model corresponding to Case 3. Again, it can be seen that the model exhibited high prediction accuracy for both output variables. Even with three predictor variables, the model performed better than the other studied models. Looking at the flue gas temperature prediction, the bias effect, observed in Figs. 15 and 16 was eliminated when the NN-NARX model was applied as demonstrated in Fig. 17. Similarly, the model also predicted the heating value of syngas in Case 3 with high accuracy unlike in the case of the other models as seen in Figs. 16 and 17.

5. Conclusions

In this particular work, software platforms applicable in developing and implementing process data analytics in modern process automation systems were reviewed, and a concept of a process monitoring platform was developed. The platform highlights the use of the state-of-the-art machine learning methods coupled with big-data processing tools and cloud computing technologies in process data analytics. With such an environment, different data-driven models can be realized. The application of the platform was demonstrated by developing data-driven soft sensors, which can be employed to monitor a WTE plant.

Data-driven soft sensors to predict syngas heating value and hot flue gas temperature in a WTE plant were studied. The work compared the performance of the static linear methods with a nonlinear dynamic method. The linear methods included multivariable linear regression, principal component regression and partial least squares regression. All these static methods performed quite well and were comparable to each other in the prediction of both syngas heating value and hot flue gas temperature. In comparison to the above statistical methods, a neural network-based NARX model showed better performance in the prediction of both dependent variables. The neural network-NARX model was able to describe the dynamic behavior of the combustion process quite well. Moreover, the neural network-NARX model also performed well in forecasting the syngas heating value as well as the flue gas temperature. The NN-NARX model was also less sensitive to the selected individual input variables compared with the other tested models. Furthermore, the study also showed that, although energy balance based soft sensors can be utilized for such purposes, developing a reliable physical model based soft sensor would require detailed knowledge about the process phenomena at hand. Hence, in cases where process knowledge is scarcely available, as in this present work, a data-driven soft sensor would be an invaluable alternative tool for predictive analytics and for use in process monitoring.

Declaration of Competing Interest

The authors declare no conflicts of interest in relation to this work.

Acknowledgments

References

[1] Trappey AJ, Trappey CV, Govindarajan UH, Sun JJ, Chuang AC. A review of technology standards and patent portfolios for enabling cyber-physical systems in advanced manufacturing. IEEE Access 2016;4:7356–82.

[2] Lui Y, Fan Y, Chen J. Flame images for oxygen content prediction of combustion systems using DBN. Energy Fuels 2017;31:8776–83.

[3] Shang C, Yang F, Huang D, Lyu W. Data-driven soft sensor development based on deep learning technique. J Process Control 2014;24:223–33.

[4] Liukkonen M, Hälikkä E, Hiltunen T, Hiltunen Y. Dynamic soft sensors for NOx emissions in a circulating fluidized bed boiler. Appl Energy 2012;97:483–90.

[5] Kortela J, Jämsä-Jounela S-L. Fuel-quality soft sensor using the dynamic super-heater model for control strategy improvement of the BioPower 5 CHP plant. Int J

[6] Kortela J, Jämsä-Jounela S-L. Fuel moisture soft-sensor and its validation for the industrial Biopower 5 CHP plant. Appl Energy 2013;105:66–74.

[7] Belkhir F, Frey G. Model-driven soft sensor for predicting biomass calorific value in combustion power plants. In: Industrial electronics and applications (ICIEA), Hefei, China; 2016.

[8] Belkhir F, Frey G. Soft-sensing of key process variables in a biomass combustion plant. In: International renewable energy congress (IREC), Hammamet, Tunisia; 2016.

[9] Korpela T, Suominen O, Majanne Y, Laukkanen V, Lautala P. Robust data reconciliation of combustion variables in multi-fuel fired industrial boilers. Control Eng Pract 2016;55:101–15.

[10] Kadlec P, Gabrys B, Strandt S. Data-driven Soft Sensors in the process industry. Comput Chem Eng 2009;33:795–814.

[11] Weiwu Y, Di T, Yujun L. A data-driven soft sensor modeling method based on deep learning and its application. IEEE Trans Ind Electron 2017;64(5):4237–45.

[12] Li G-Q, Qi X-B, Chan KC, Chen B. Deep bidirectional learning machine for predicting NOx emissions and boiler efficiency from a coal-fired boiler. Energy Fuels 2017;31:11471–80.

[13] Toth P, Garami A, Csordas B. Image-based deep neural network prediction of the heat output of a step-grate biomass boiler. Appl Energy 2017;200:155–69.

[14] Ögren Y, Toth P, Garami A, Sepman A, Wiinikka H. Development of a vision-based soft sensor for estimating equivalence ratio and major species concentration in entrained flow biomass gasification reactors. Appl Energy 2018;226:450–60.

[15] Pandey SD, Das S, Pan I, Leahy JJ, Kwapinski W. Artificial neural network based modelling approach for municipal solid waste gasification in a fluidized bed reactor. Waste Manage 2016;58:202–13.

[16] Xiao G, Ni M-J, Chi Y, Jin B-S, Xiao R, Zhong Z-P, et al. Gasification characteristics of MSW and an ANN prediction model. Waste Manage 2009;29:240–4.

[17] Mikulandrić R, Böhning D, Böhme R, Helsen L, Beckmann M, Loncar D. Dynamic modelling of biomass gasification in a co-current fixed bed gasifier. Energy Convers Manage 2016;125:264–76.

[18] Salah A, Hanel L, Beirow M, Scheffknecht G. Modelling SER biomass gasification using dynamic neural networks. In: 26th European symposium on computer aided process engineering, Portorož, Slovenia, June 12th-15th; 2016.

[19] Ruhanen E, Kosonen M, Kauvosaari S, Henriksson B. Optimisation of paste thickening at the Yara Siilinjärvi plant. In: Paste , Perth, Australia; 2018.

[20] O'Donovan P, Leahy K, Bruton K, O'Sullivan DTJ. An industrial big data pipeline for data-driven analytics maintenance applications in large-scale smart manufacturing facilities. J Big Data 2015;2(25):1–26.

[21] Siemens PLM Software. MindSphere: the cloud-based, open IoT operating system for digital transformation; 2017. [Online]. Available: https://www.plm.automation.siemens.com/media/global/en/Siemens_MindSphere_Whitepaper_tcm27-9395.pdf. [accessed 22 February 2018].

[22] IEC market strategy board. IoT 2020: smart and secure IoT platform. 16th February 2017. [Online]. Available: https://www.iec.ch/whitepaper/pdf/iecWP-loT2020-LR.pdf. [accessed 31st October 2018].

[23] Tseng M, Edmunds T, Canaran L. Introduction to edge computing in IIoT. 18th June 2018. [Online]. Available: https://www.iiconsortium.org/pdf/Introduction_to_Edge_Computing_in_IIoT_2018-06-18.pdf. [accessed 31st October 2018].

[24] Sabella R, Thuelig A, Carrozza MC, Ippolito M. 5G and industrial automation. 18th February 2018. [Online]. Available: https://www.ericsson.com/assets/local/publications/ericsson-technology-review/docs/2018/etr_2018-02_robotics_web-feb18.pdf. [accessed 31st October 2018].

[25] IBM. IBM Watson machine learning. 2nd August 2018. [Online]. Available: https://developer.ibm.com/clouddataservices/docs/ibm-watson-machine-learning/. [accessed 31st October 2018].

[26] Microsoft. Azure IoT technologies and solutions: PaaS and SaaS, 18 5 2018. [Online]. Available: https://docs.microsoft.com/en-us/azure/iot-fundamentals/iot-services-and-technologies. [accessed 12 August 2018].

[27] Siemens. Introduction to MindSphere predictive learning. [Online]. Available: https://documentation.mindsphere.io/resources/html/predictive-learning/en-US/MDSPPredictiveHelp.htm#t=Introduction_to_Predictive_Learning.htm. [accessed 31st October 2018].

[28] Wu Y, Xu C, Zhang T. Evaluation of renewable power sources using a fuzzy MCDM based on cumulative prospect theory: A case in China. Energy 2018;147:1227–39.

[29] Inayata A, Razab M. District cooling system via renewable energy sources: A review. Renew Sustain Energy Rev 2019;107:360–73.

[30] Tajeddina A, Roohi E. Designing a reliable wind farm through hybridization with biomass energy. Appl Therm Eng 2019;154:171–9.

[31] Makarichi L, Jutidamrongphan W, Techato K-A. The evolution of waste-to-energy incineration: A review. Renew Sustain Energy Rev 2018;91:812–21.

[32] Moya D, Aldás C, López G, Kaparaju P. Municipal solid waste as a valuable renewable energy resource: a worldwide opportunity of energy recovery by using waste-to-energy technologies. Energy Procedia 2017;134:286–95.

[33] Malinauskaite J, Jouhara H, Czajczyńska D, Stanchev P, Katsou E, Rostkowski P, et al. Municipal solid waste management and waste-to-energy in the context of a circular economy and energy recycling in Europe. Energy 2017;141:2013–44.

[34] Bourtsalas AC, Seo Y, Alam MT, Seo Y-C. The status of waste management and waste to energy for district heating in South Korea. Waste Manage 2019;85:304–16.

[35] Wu Y, Xu C, Li L, Wang Y, Chen K. A risk assessment framework of PPP waste-to-energy incineration projects in China under 2-dimension linguistic environment. J Cleaner Prod 2018;183:602–17.

[36] Lombardi L, Carnevale EA. Evaluation of the environmental sustainability of different waste-to-energy plant configurations. Waste Manage 2018;73:232–46.

[37] Anagnostopoulos T, Zaslavsky A, Kolomvatsos K, Medvedev A, Amirian P, Morley J,

et al. Challenges and opportunities of waste management in IoT-enabled smart cities: a survey. IEEE Trans Sustain Comput 2017;2(3):275–89.

[38] Aazam M, St-Hilaire M, Lung C-H, Lambadaris I. Cloud-based smart waste management for smart cities. In: IEEE 21st international workshop on computer aided modelling and design of communication links and networks (CAMAD), Toronto, ON, Canada, 23-25 Oct. 2016.

[39] Bányai T, Tamás P, Illés B, Stankeviciute Z. Optimization of MunicipalWaste collection routing: impact of industry 4.0 technologies on environmental awareness and sustainability. Int J Environ Res Public Health 2019;16(4):1–26.

[40] Popa CL, Carutasu G, Cotet CE, Carutasu LN, Dobrescu T. Smart city platform development for an automated waste collection system. Sustainability 2017;9(11):1–15.

[41] Pardini K, Rodrigues JJPC, Kozlov SA, Kumar N, Furtado V. IoT-based solid waste management solutions: a survey. J Sens Actuator Networks 2019;8(1):1–25.

[42] Van Kessel L, Arendsen A, Brem G. On-line determination of the calorific value of solid fuels. Fuel 2004;83:59–71.

[43] Atnaw SM, Sulaiman SA, Yusup S. Influence of fuel moisture content and reactor temperature on the calorific value of syngas resulted from gasification of oil palm fronds. The Sci World J 2014;2014:1–9.

[44] Boriouchkine A, Jämsä-Jounela S-L. Simplification of a mechanistic model of biomass combustion for on-line computations. Energies 2016;9(9):735.

[45] Bardi S, Astolfi A. Modeling and control of a waste-to-energy plant: waste-bed temperature regulation. IEEE Control Syst Mag 2010;30(6):27–37.

[46] Kumar P, Amgoth T, Annavarapu C. Machine learning algorithms for wireless sensor networks: A survey. Information Fusion 2019;49:1–25.

[47] Amasyali K, El-Gohary NM. A review of data-driven building energy consumption prediction studies. Renew Sustain Energy Rev 2018;81:1192–205.

[48] Kadlec P, Grbić R, Gabrys B. Review of adaptation mechanisms for data-driven soft sensors. Comput Chem Eng 2011;35:1–24.

[49] Siow E, Tiropanis T, Hall W. Analytics for the internet of things: a survey. ACM Comput Surv 2018;1(1):1–35.

[50] Tüfekci P. Prediction of full load electrical power output of a base load operated combined cycle power plant using machine learning methods. Electr Power Energy Syst 2014;60:126–40.

[51] Fast M, Palme T. Application of artificial neural networks to the condition monitoring and diagnosis of a combined heat and power plant. Energy 2010;35:1114–20.

[52] Ahmed A, Arab K, Bouida Z, Ibnkahla M. Data communication and analytics for smart grid systems. In: 2018 IEEE international conference on communications (ICC), Kansas City, MO, USA, 20-24 May 2018.

[53] Ge Z, Song Z, Ding S, Huang B. Data mining and analytics in the process industry: the role of machine learning. IEEE Access 2017;5:20590–616.

[54] Brown SH. Multiple linear regression analysis: a matrix approach with matlab. Alabama J Math 2009:1–3.

[55] Kämpjärvi P, Sourander M, Komulainen T, Vatanski N, Nikus M, Jämsä-Jounela S-L. Fault detection and isolation of an on-line analyzer for an ethylene cracking process. Control Eng Pract 2008;6:1–13.

[56] Rato T, Reis M, Schmitt E, Hubert M, Ketelaere BD. A systematic comparison of PCA-based statistical process monitoring methods for high-dimensional, time-dependent processes. AIChE J 2016;26(5):1478–93.

[57] Godoy JL, Vega JR, Marchetti JL. Relationships between PCA and PLS-regression. Chemom Intell Lab Syst 2014;130:182–91.

[58] Diversi R, Guidorzi R, Soverini U. Identification of ARX and ARARX models in the presence of input and output noises. Eur J Control 2010;3:242–55.

[59] Dhussa AK, Sambi S, Kumar S, Kumar S, Kumar S. Nonlinear Autoregressive Exogenous modeling of a large anaerobic digester producing biogas from cattle waste. Bioresour Technol 2014;170:342–9.

[60] Boussaada Z, Curea O, Remaci A, Camblong H, Bellaaj NM. A nonlinear auto-regressive exogenous (NARX) neural network model for the prediction of the daily direct solar radiation. Energies 2018;620(11):2–21.

[61] Chan RW, Yuen JK, Lee EW, Arashpour M. Application of nonlinear-autoregressive-exogenous model to predict the hysteretic behaviour of passive control systems. Eng Struct 2015;85:1–10.

[62] Bastos Alvaro Furlani, Lao Keng-Weng, Todeschini Grazia, Santoso Surya. Novel moving average filter for detecting RMS voltage step changes in triggerless PQ data. IEEE Trans Power Delivery 2018;33(6):2920–9. https://doi.org/10.1109/TPWRD.2018.2831183.

[63] Mottahedi A, Sereshki F, Ataei M. Development of overbreak prediction models in drill and blast tunneling using soft computing methods. Eng Comput 2018;34:45–58.

[64] Cao M, Alkayem NF, Pan L, Novák D. Advanced methods in neural networks-based sensitivity analysis with their applications in civil engineering. In: Artificial neural networks: models and applications, Rijeka, Croatia, IntechOpen; 2016. p. 335–53.