Computers and Electronics in Agriculture 203 (2022) 107475

Contents lists available at ScienceDirect

Computers and Electronics in Agriculture

journal homepage: www.elsevier.com/locate/compag

Next Generation Task Controller for agricultural Machinery using OPC Unified architecture

Matti Siponen [a], Ilkka Seilonen [a], Samuel Brodie [b], Timo Oksanen [a,b,*]

[a] Department of Electrical Engineering and Automation, Aalto University, P.O. Box 15500 (Maarintie 8), 02150 Espoo, Finland
[b] Chair of Agrimechatronics, Technical University of Munich, 85354 Freising, Germany

ARTICLE INFO

Keywords:
ISO 11783
ISOBUS
OPC UA
Standardization

ABSTRACT

ISO 11783 is a communication protocol for tractors and implements. It specifies a network of control functions connected with a CAN bus. The task controller is a control function that is used to control implements based on planned tasks. Implements expose their structure to the task controller with Device Descriptor Object Pool XML files. The guidelines for modelling implements leave room for interpretation, which can lead to compatibility issues. The agricultural industry has decided that the physical layer of the future high-speed ISOBUS will use Ethernet, but the higher OSI layers are undecided. One Ethernet-compatible middleware to be considered is OPC UA, a platform-independent communication standard for systems and devices. It uses object-oriented information modelling techniques and allows users to define new information models based on the OPC UA base information model and its standardized extensions. In this article, an OPC UA information model was designed for data exchange between the task controller and implements, and its suitability for use in the high-speed Next-Generation ISO 11783 Task Controller was evaluated. Modelling rules for the information model were designed with a focus on modelling all types of implements that apply a product or products. OPC UA Implement Server and OPC UA Task Controller applications were developed to test the information model. The evaluation of these OPC UA applications showed that the OPC UA model (OPC UA for the Next Generation ISO 11783 Task Controller information model) presented is suitable for communication between the task controller and implements.

1. Introduction

Precision farming is the use of information technologies to gather and analyze data to make decisions related to crop production (National Research Council, 1997); defined by Zhang et al. (2002) as “the spatial and temporal variability of soil and crop factors within a field” it is an important topic in agriculture and as such is widely researched.

One focus of precision agriculture is on sensing and imaging (Mulla, 2013; Weiss et al., 2020). The current state of the art includes thermal remote imaging (Khanal et al., 2017; Stark et al., 2014), optical imaging (Kuska & Mahlein, 2018), and radar (El Hajj et al., 2017).

In general, the above methods are used to produce management zones which describe the properties of a field in spatial terms, normally in relation to its potential crop yield or nutrient content (Farid et al., 2016; Khosla et al., 2008; Piotrowska-Długosz et al., 2011). Tummers et al. (2019) describe fertilization management and data acquisition as two of the key features present in modern Farm Management Information Systems.

These management zones can be used to create prescription maps, maps that determine how product should be applied to the field spatially. This is achieved with variable-rate application (sometimes referred to as site-specific application), a technology where a product’s application rate is varied depending on either online data from sensors or GNSS position and a predetermined prescription map (Grisso et al., 2011).

To enable agricultural machinery made by different manufacturers to utilize these technologies when connected together, there has to be an international agreement on how to exchange data related to site-specific application and section control. The ISO 11783 standard series defines a communication protocol and communication network for tractors and implements (ISO, 2017). The Task Controller (TC) is a component of ISO 11783 networks used for controlling implements based on planned tasks received from a Farm Management Information System, or FMIS (ISO, 2015a).

* Corresponding author.
E-mail address: timo.oksanen@aalto.fi (T. Oksanen).

https://doi.org/10.1016/j.compag.2022.107475
Received 20 May 2022; Received in revised form 25 October 2022; Accepted 30 October 2022
Available online 5 November 2022

The ISOBUS Task Controller enables easy treatment of management zones by defining TC-GEO functionality to plan for product application that varies spatially. Automatic section control is a technology where boom sections consisting of nozzles or rows are automatically turned off when entering an area that has already been treated to avoid applying overlap treatment (Larson et al., 2015).

Implements expose their structure to the TC with Device Descriptor Object Pool (DDOP) XML files that are transferred from implements to the TC via CAN bus (ISO 11898) using the CAN 2.0B extended frame message format. Although there are guidelines on how implements should be modelled with the DDOP format, they allow different modelling approaches to be used, which in turn leads to similar implements sometimes having vastly different DDOPs. This means that it's possible that the TC won't be able to control an implement properly, if at all, because it fails to process the DDOP. Furthermore, the authors posit that the DDOP format provides insufficient means for modelling booms explicitly, and additionally that it does not provide any means for modelling many-to-many relationships between bins (defined in ISO 11783 as the tank of a sprayer or the bin of a seeder) and booms. For instance, under the current ISO DDOP design model it is not possible to model an implement that has two booms which share the same bin.

The first objective of this article is to identify how the DDOP format could be enhanced to enable new modelling features that would be beneficial to future devices, and how this enhanced DDOP format could be used to guarantee that similar implements are also modelled similarly. An Ethernet-based network shall replace the use of CAN in the future high-speed network (Smart & Brill, 2019), however, the higher-level protocols have not been decided (Piirainen, 2014). Smart & Brill (2019, 2022) state that "The current HSI requirements do not require deterministic behaviour in the communications. However, in researching the potential impact if or when a deterministic link needs to be developed (other than that used by PT07), such as Time Sensitive Networking (TSN), the basic HSI architecture could evolve to support this, or other potential deterministic protocols. In this scenario, the known HSI needs are delivered best-effort, and since the 1 Gb/s network is sufficiently over-provisioned, the real-time requirements can still be met. This transition would require a change in the network switch hardware, and hardware/software at the end-points that require the deterministic behaviour. Due to the projected cost and complexity, and in the absence of requirements for this capability, this is presently out of scope for HSI.".

Oksanen et al. (2015) studied the use of OPC UA for remote access of machinery process data such as GNSS position or process data extracted from ISO 11783-compatible implements. In the 2015 work OPC UA was not used for tractor-implement in-vehicle communication and control, rather it was used to broadcast existing information from the ISO 11783 network (based on CAN-bus) to a remote client system. To further study the viability of using OPC UA in ISO 11783 networks, the second objective of this article is to design and evaluate an OPC UA information model for data exchange between the TC and implements using the aforementioned enhanced DDOP format. This information model shall be called "OPC UA for the Next Generation ISO 11783 Task Controller" (OPC UA TC) and it shall be evaluated with OPC UA applications acting as an implement and a TC as illustrated in Fig. 1.

This article addresses the following questions:

a) What new modelling features could be added to enhance the current DDOP format?
b) How could this enhanced DDOP format be implemented in the OPC UA TC information model?
c) How should OPC UA TC be used to guarantee that implements that are alike are modelled in the same way?
d) Is OPC UA TC suitable as a higher-layer protocol for communication between a TC and an implement?

The necessity of information modelling in this way is that it enables inter-manufacturer, plug-and-play connection between a TC and an implement, as is possible today, but with much faster data rates (by using Ethernet). Additionally, this opens up the potential for improving upon the application layer by replacing the underlying CAN messages of today with UPC UP messages. The need for standardized information modelling stems from the fact that for a TC to accurately control an implement, it must have information about certain characteristics such as geometry, actuation delays, controllable sections et cetera.

2. The ISO 11783 standard series

2.1. Tractor-implement network

ISO 11783 is a communication protocol for agricultural and forestry

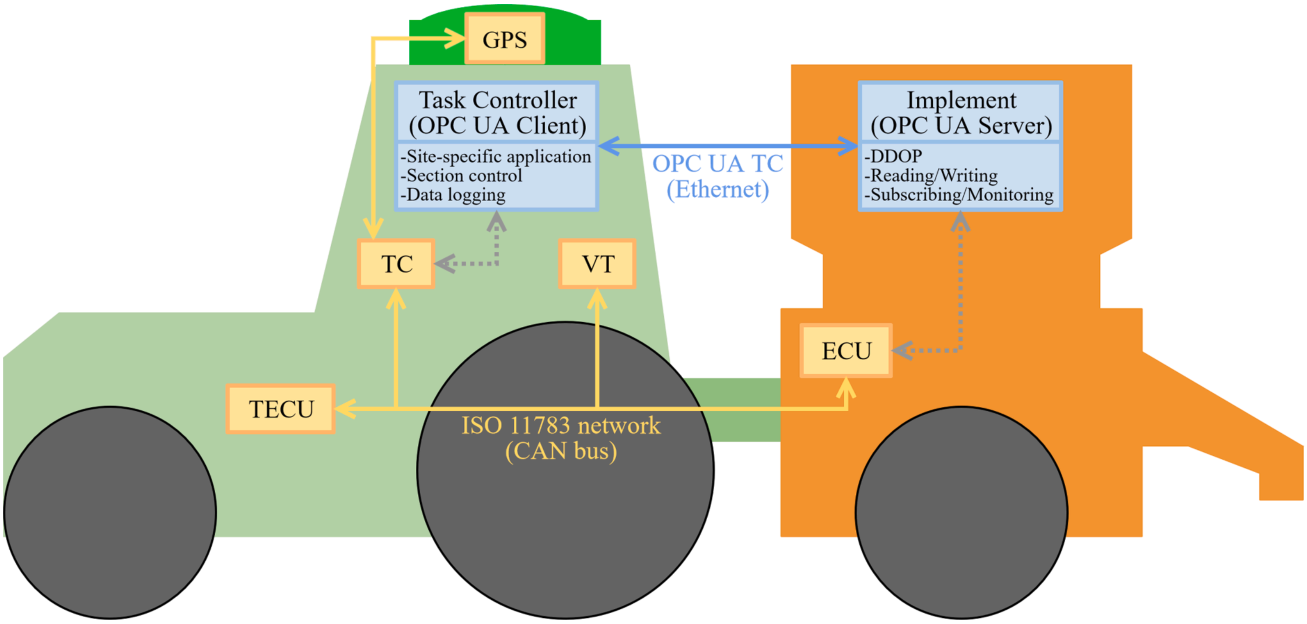

Fig. 1. An OPC UA Client acting as the TC and an OPC UA Server acting as an implement communicating using OPC UA TC. The green shape on the left is the tractor and the orange on the right is an implement.

tractors and machinery (ISO, 2017). The standard series specifies a network for control and communication between tractors and implements (ISO, 2011a). The entities on an ISO 11783 network are electronic control units (ECU) connected with a single linear, 250 kbit/s, twisted, non-shielded, quad-cable (ISO, 2019a) which communicate with one another using the CAN 2.0B extended frame format with 29-bit identifiers and up to 8 bytes of data per frame (ISO, 2018a). The functionalities performed by an ECU are called control functions (CF) and they include spray rate control, section on/off control, product loss monitoring and so on. A collection of CFs that work together to perform as a group of CFs is called a working set with one of the CFs acting as the working set master. The working set master communicates with the common user interface, the Virtual Terminal (ISO, 2018b) and also with the Task Controller (ISO, 2015a). The basic process data communication with the tractor is based on separate messaging defined in ISO11783-7 Implement Messages Application Layer (ISO, 2015b). Each CF in an ISO 11783 network must have a unique 64-bit NAME that both identifies and describes it (ISO, 2019b). The bits that compose a NAME can be split into ten separate bit fields that each describe a different aspect of the CF such as manufacturer, device class, function and so on. NAMEs are sufficient for uniquely identifying CFs on a global level.

2.2. Compliance

ISO 11783 leaves some room for interpretation, which could potentially lead to various incompatible implementations of the standard. To avoid this, the Agricultural Industry Electronics Foundation (AEF) was founded in 2008 by implement and tractor manufacturers with the aim of promoting the use of ISOBUS, their implementation of ISO 11783, in order to enhance compatibility between tractors and implements made by different manufacturers. The AEF has defined guidelines on how ISO 11783 should be interpreted to guarantee compatibility between ISOBUS products (AEF, 2012d). To market a product as ISOBUS compatible, the product must pass one or more of the AEF's conformance tests. The products that have been certified by the AEF are given the AEF Certified Label and added to AEF ISOBUS database.

2.3. Task controller

The purpose of the Task Controller (TC) is to automate some aspects of farmwork such as controlling the (variable) product application rate, turning sections on/off, and recording data. On a technical level the TC is a CF that manages a connection between itself and the implements, and communicates with them using process data messages (ISO, 2015a). The TC receives tasks from an FMIS in the form of an XML file. In addition to commands related to site-specific application, the TC can also support section control by keeping a map of treated areas and commanding boom sections on and off to avoid applying a product multiple times to a treated area. The results of an executed task are saved in XML format and sent to the FMIS.

Compatible implements for planned tasks are identified by comparing their NAME to the full or partial NAME preselected at the FMIS. The NAME provides sufficient means for describing single operation implements with its industry group, device class, and function fields, however, no identifiers have been defined for identifying multi-operation implements using these fields. In practice, the FMIS has to specify the manufacturer and the model of a multi-operation implement in a planned task, which means that the task will not be compatible with similar multi-operation implements made by different manufacturers. Therefore, when a task is planned in an FMIS, the use of NAME (industry group, device class, and function fields) is not adequate for identifying all compatible implements and hinders the TC's ability to utilize implements that would be suitable for the planned task.

The DDOP is a standardized way for device manufacturers to describe their devices (ISO, 2015a). It is imported/exported to the TC in XML format and transferred in a binary format if it is transmitted on the bus (from the implement to the TC). A DDOP file shall contain all information necessary to plan tasks using the device. A DDOP consists of exactly-one device (DVC) object, at least one device element (DET) object and any number of device process data variable (DPD), device property (DPT) and data value presentation (DVP) objects. The DVC object acts as the root object of the DDOP and provides general attributes of the modelled device such as its NAME. The DET objects have different device element types, such as connector, bin, section and so on, and form a tree hierarchy of the modelled device's components and subsystems. The DPD and DPT objects expose process data variables and properties related to a DET object that are referenced using device object reference (DOR) objects. DVP objects allow defining unit conversions for values of DPD and DPT objects that are presented with 32-bit signed integers. An implement's DDOP is transferred via CAN bus after a handshaking process where the implement queries if the current version of the DDOP has already been transferred to the TC as well as if the TC has enough memory to receive it.

2.4. Data dictionary

The ISOBUS data dictionary contains a record of DDEs (data dictionary entities) that are used to describe DPD and DPT objects present in DDOPs (ISO, 2011b; VDMA, 2020). Each DDE has a unique data dictionary identifier (DDI) number, definition of the data, unit, bit resolution and ranges. DDEs may also contain comments and attachments. There are currently over 500 DDEs defined such as "Setpoint Mass Per Area Application Rate", "Actual Seeding Depth", "Total Area" and so on. DDIs are referenced by DPD and DPT objects to define the data they represent and by planned tasks to define which DPD objects are to be monitored and/or controlled by the TC. The DPD objects to be controlled are typically setpoints of application rates, work states, working depths and so on. DVP objects can be used to convert the unit of a DPD or a DPT object specified by a DDI to custom units such as converting millimeters to meters, inches or feet.

DDE number 179 "Actual Cultural Practice" is used by function and bin DET objects to specify which current cultural practice is being performed. This enumerated list includes: fertilizing (#1), sowing and planting (#2), crop protection (#3), harvesting (#8), and so on. Cultural practice identifiers are used by multi-operation implements to allow the TC to control the correct subsystem of the implement when both subsystems have the same DPDs. An FMIS may reference a cultural practice and optionally an operation technique which further describes the cultural practice, e.g., operation techniques for fertilizing would be liquid fertilizing, organic fertilizing and gaseous fertilizing.

The AEF has defined guidelines for implementing three separate levels of TC functionality: "basic" for logging task totals, "geo" for site-specific application, and "section control" for turning sections on and off based on treated areas (AEF, 2012a; AEF, 2012b; AEF, 2012c). These three TC functionalities and a fourth functionality for data loggers were added to the second edition of ISO 11783–10 published in 2015 (ISO, 2015a). ISO also defines recommended DDIs for each device class and TC functionality, as well as providing example DDOPs.

However, the guidelines in the ISO standard leave a lot of room for interpretation, which allows similar implements to be modelled differently. In the end, whether or not a TC made by one manufacturer and an ISOBUS-certified implement made by another manufacturer are fully compatible can't be known for certain until they are connected to one another and tested. Testing all possible combinations in advance isn't feasible, which means that it falls upon the end users to do the tests. The compatibility issues between the TC and implements could be avoided by defining and enforcing stricter modelling rules that would leave no room for interpretation. However, this could make all currently available devices incompatible with future devices conforming to the new modelling rules. On the other hand, replacing the CAN bus and CAN 2.0B extended frame format with another communication protocol would also make current devices incompatible with future devices, which would make such a transition a logical point for defining stricter modelling rules and adding new modelling features.

3. Requirements for data exchange in tractor-implement system

3.1. Key properties of ISO 11783

At the time the current network for tractor-implement systems was developed, in the late 1980 s, it was considered that a 250 kbit/s network would fulfill all requirements far into the future (Oksanen & Auernhammer, 2021). The network was based on the latest technology at that time, CAN bus, which was a good choice for the standardization group as the technology is still valid and mainstream in vehicle networks. The fundamentals of the ISO 11783 network have been based on a maximum CAN backbone length of 40 m and data rate of 250 kbit/s (Smart & Brill, 2019).

3.2. Use cases

The agricultural industry has prepared for the future demands of a next-generation communication system by studying use cases which need a fast network. These include: “more precise command and control for prescription”, “more precise data logging of as-applied and yield information” and “higher featured and more responsive display of information” (Smart & Brill, 2019). For instance, a TC would be able to control the seed-rate of each row of a planter and manage target seed spacing far more consistently (Smart & Brill, 2019).The specific use cases related to Task Controller are primary drivers for a faster network (Smart & Brill, 2019). It has been identified that command and control timing accuracy should be “well below 10 ms” (Smart & Brill, 2019); while the current rate with CAN bus is limited to 100 ms.

The other use cases of a faster network are related to digital camera systems, in-field vehicle-to-vehicle communication, plus HMI and electric drives in the plug-and-play system (Smart & Brill, 2019). These use cases are beyond the scope of this article.

3.3. Physical layer

In this article, the requirements of the agricultural industry are considered when assessing technological viability such as modularity, hot plug-and-play and intercompatibility between different machine manufacturers (Oksanen & Auernhammer, 2021).

Earlier studies have compared the various communication technologies available to the agricultural industry. These include CAN Flexible Data Rate (CAN-FD), FlexRay, wireless radio technologies and Ethernet. Out of these options, Ethernet was found to be the clear best option, with automotive cabling, one pair Ethernet. The preferred option out of many Ethernet technologies is IEEE 1000BASE-T1 (Smart & Brill, 2019).

As an industry requirement, the architecture should be similar to the current ISO 11783 network. This would allow for parallel networks during the transition, having the existing ISO 11783 and the new faster backbone cables next to each other in the wire harness. Easy hot plug-and-play of implements and tractor is a mandatory requirement and the system cannot be pre-programmed due to the plug-and-play nature. (Smart & Brill, 2019).

The key difference to a CAN-based network is related to physical topology: Ethernet requires switches between physical links.

4. OPC Unified architecture

OPC UA is a platform-independent communication standard for industrial systems and devices (OPC Foundation, 2017a). The main purpose of OPC UA is the modelling and communicating of data (Mahnke et al., 2009). OPC UA allows the modelling of information with object-oriented techniques, such as type hierarchies and inheritance (OPC Foundation, 2017c; OPC Foundation, 2019). OPC UA supports communicating data between OPC UA applications by defining mappings between the security model, service sets, data structures and compatible network protocols (OPC Foundation, 2017f.).

In an OPC UA Client-Server communication model, *Client* and *Server* applications form a *Session* to exchange *Service* request and response *Messages* via OPC UA Communication Stack (OPC Foundation, 2017a). OPC UA defines ten *Service Sets* that provide Services for discovering Servers, managing *Nodes*, reading and writing *Attributes*, calling *Methods* and so on (OPC Foundation, 2017g). OPC UA also supports the OPC UA PubSub communication model (OPC Foundation, 2018a), though this article is focused on the OPC UA Client-Server model for the following reasons:

1. The current ISO 11783 Task Controller is defined in such a way that the TC server and TC client maintain a one-to-one connection. The OPC UA *Sessions* in the Client-Server model are largely the same. To replicate the setpoint rate-actual rate design of ISO 11783 with PubSub, each tractor and implement would need to act as both publisher and subscriber simultaneously.
2. Broadcast messaging (as enabled by PubSub) is already a feature of ISO 11783 but broadcast messages do not form a part of the TC operation (with the exception of initialization, which can be handled differently in OPC UA).
3. PubSub was not mature at the time this work began. Although the protocol was defined, there were no suitable SDKs/implementations.

The information exposed by an OPC UA Server is called its *AddressSpace* and it consists of Nodes connected by *References* (OPC Foundation 2017a). A Node consists of Attributes that identify and name the Node and *References* that connect the Nodes to each other. Nodes are used for both defining types and creating instances of the defined types (OPC Foundation 2017a). Types are defined with *ObjectType*, *VariableType*, *DataType* and *ReferenceType* Nodes, which are collectively called *TypeDefinitionNodes*, while instances are created with *Object*, *Variable* and *Method* Nodes. Subsets of the AddressSpace can be defined with *View* Nodes. An OPC UA information model defines a standardized configuration of Nodes and References for Servers using the information model. The base OPC UA information model is used as a basis for defining new information models (OPC Foundation 2017b). Standardized information models have been defined for *Data Access* (OPC Foundation 2017d) *Alarms & Conditions* (OPC Foundation 2017i.), *Programs* (OPC Foundation 2017e), *Historical Access* (OPC Foundation 2018c), *Aggregates* (OPC Foundation 2017j.) and *Devices* (OPC Foundation 2013). In addition to the base and standardized information models, OPC UA also allows users to define their own information models that best suit their needs. These information models are called companion specifications and they can be created internally by the OPC Foundation, jointly with the OPC Foundation and another organization or externally without the OPC Foundation’s involvement. Joint companion specifications include IEC61131-3 Client *Function Blocks* for OPC UA (OPC Foundation and PLCopen, 2016), OPC UA for AutomationML (OPC Foundation and AutomationML, 2016), OPC UA for Robotics (OPC Foundation and VDMA, 2019), OPC UA for Commercial Kitchen Equipment (OPC Foundation and HKI, 2019) and many more. The use of OPC UA in various applications has been studied thoroughly in recent years. OPC UA for Smart Grids has been demonstrated by Lehnhoff et al. by mapping the Common Information Model (CIM) and IEC 61850 to OPC UA (Lehnhoff et al., 2012). Kožár and Kadera have developed IEC 61,499 function blocks representing OPC UA Clients for data exchange with remote OPC UA Servers and an automated process for creating the AddressSpace for an OPC UA Server based on the structure of an IEC 61,499 application (Kožár and Kadera, 2016). OPC UA was applied to cyber-physical production systems of Industry 4.0 by Schleipen et al. in four different application scenarios, which were quality defect tracking, visualization of process information, management information board and orchestration of cyber-physical production systems (Schleipen et al., 2016). Pauker et al. have presented a systematic approach to designing OPC UA information models in the manufacturing domain with model-driven architecture and unified modelling language (Pauker et al., 2016). The feasibility of using OPC UA in field devices has been analyzed by Veichtlbauer et al. and the authors found advantages such as seamless integration of field devices and semantic interoperability in multi-vendor automation systems but also disadvantages such as poor resource utilization and communication latencies, which the authors suggest will become less significant as embedded devices become more powerful (Veichtlbauer et al., 2017). Kim and Sung have evaluated the use of the PLCopen OPC UA companion specification with an experimental robot system where OPC UA was used to exchange data between an OPC UA Client-Server hybrid application acting as a motion controller and an OPC UA Server application acting as a supervisory machine (Kim and Sung, 2017). Lee et al. have defined a bi-directional transformation between OPC UA and UML and demonstrated their approach with power grid, building automation and smart device domain use cases (Lee et al., 2017). An OPC UA information model has been designed by Wally et al. for modelling variability information in automated production systems that can be used in parallel with other information models (Wally et al., 2018). Ye and Hong have developed a four-layer architecture for a manufacturing system referencing the Reference Architecture Model for Industry 4.0 (RAMI 4.0) model using AutomationML and OPC UA (Ye and Hong, 2018). Cavalieri et al. have provided a solution for integration of OPC UA and Open Connectivity Foundation (OCF) Resource Model with OCF OPC UA Information Model that provides ObjectTypes for mapping the OCF Resource Model to the OPC UA AddressSpace (Cavalieri et al., 2019). Tantik and Anderl have validated their concept of applying representational state transfer to OPC UA with a use-case involving an OPC UA Client that measures and sets the position of a pattern on a gripper of a robot arm represented by an OPC UA Server utilizing the OPC UA for Robotics companion specification to model the robot (Tantik and Anderl, 2019). The problem of the convergence of information technology and operation technology has been addressed by Tian and Hu with their OPC UA information model for time-sensitive networking (Tian and Hu, 2019). Beňo et al. have demonstrated transferring data from an OPC UA Server to the cloud with a Raspberry Pi 3 acting as an IoT edge device and discussed the benefits of using this architecture including modular deployment and safe data storage (Beňo et al., 2019). The scalability of OPC UA has been demonstrated by Pribiš et al. by implementing an OPC UA Server on an ARM 32-bit microcontroller connected via Ethernet switch to another similar microcontroller running a UaExpert OPC UA Client (Pribiš et al., 2019).

The transfer of data on ISO 11783 communication networks with OPC UA was studied by Piirainen in their article (Piirainen, 2014). An information model for representing DDOPs with OPC UA Objects and Variables was designed and tested. This information model was based on the device model defined in the OPC UA for Devices companion specification and it defined ObjectTypes and VariableTypes for representing device descriptor objects. The OPC UA TC information model designed for this article is inspired by the information model designed by Piirainen.

The performance of OPC UA in an agricultural (tractor-implement) configuration would depend on several factors. One factor is how the industry decides to standardize the physical layer, which as discussed, will very likely be IEEE 1000BASE-T1 (Smart & Brill, 2019).

Profanter et al. (2019) studied the performance of several Industry 4.0 protocols and concluded that “OPC UA has its strength in the semantic modeling of information. […] The performance comparison of the protocol implementations shows that open62541 for OPC UA and eProsima FastRTPS for DDS deliver high performance, whereas the MQTT and ROS implementations show a significant slowdown in the RTT of packages sent to the server”.

There is a range of literature which establishes that the specific implementation (i.e. SDK/library) of the OPC UA protocol affects the performance of the system. Haskamp et al. (2017) performed benchmarking of twenty-one different OPC UA implementations and found that only three satisfied all of the evaluation criteria defined. Morato et al. (2021) tested four different OPC UA implementations. The results showed that the *task execution time* (latency) and CPU usage are affected by the implementation, with the best performing being the open source implementation *open62541* which has a mean task execution time of 312.83 μs when using 100 Mbit/s Ethernet, and significantly degraded performance when using WiFi. Zunino et al. (2021) found similar discrepancies between different implementations when testing PubSub, while Cavalieri and Chiacchio (2013) evaluate the performance impact of various OPC UA features such as publish interval and security which are both shown to impact performance. This paper focuses on information modelling as the specific hardware/software/security requirements of the agricultural industry are yet to be adequately defined.

5. Requirements of OPC UA for the next Generation ISO 11783 Task Controller

OPC UA TC is required to expose an implement’s DDOP as Object and Variable Nodes in an OPC UA Server’s AddressSpace. In addition to exposing an implement’s DDOP, OPC UA TC should provide a means for controlling the implement by allowing Clients to either: write values to settable DPD objects represented by Variable Nodes, or, call functions represented by Method Nodes to perform more complex control operations. A standardized information model for modelling devices has been defined in the OPC UA for Devices companion specification and OPC UA TC shall use this device model as its base.

In addition to implementing the data modelling requirements for the DDOP format defined by ISO 11783–10, new requirements to enhance OPC UA TC with new modelling features have been identified for this paper. These new modelling features are:

a) grouping process data variables and properties
b) modelling many-to-many relationships of device elements
c) creation of new device element type for modelling booms
d) creation of new cultural practice identifiers

To unify how implements of a common type are modelled with OPC UA TC, stricter modelling rules must be defined for different device element types. These modelling rules include defining the allowed relationships between DET objects of the same or different device element types and defining required process data variables and properties for each device element type.

OPC UA TC is evaluated with an OPC UA Server acting as a simulated implement and an OPC UA Client acting as a simplified TC in simulated scenarios relating to the control of product-applying implements. The TC has tasks involving data logging, site-specific application and section control.

5.1. Grouping process data variables and properties

In the current the ISO 11783 standard both DPD and DPT objects reference DDIs to identify the definition, unit, value range and other metadata of the process data variable or property (ISO, 2015a). This approach requires the TC to process all the DPD and DPT objects one by one with different handlers based on their DDIs. It would be beneficial if DPD and DPT objects could be organized into groups of related DPD and DPT objects e.g., DPD objects representing “Setpoint Mass Per Area Application Rate” and “Actual Mass Per Area Application Rate” could be grouped together. Therefore, a new device descriptor object was designed for grouping DPD and DPT objects. To make these groups more efficient to process, objects organizing DPD and DPT objects should be able to identify their contents based on an identifier instead of presenting a list of DDIs. This identifier is required to be applicable to all currently defined DDEs.

5.2. Modelling many-to-many relationships of device elements

As defined currently, the parent–child relationships of DET objects in a DDOP form a tree structure. Modelling many-to-many relationships is not possible when using parent–child relationships. However, the authors find that modelling many-to-many relationships would be beneficial in future use cases, such as modelling more complicated relationships of booms and bins that are not expressible with only parent–child or sibling relationships. Therefore, a new type of device descriptor object has been designed for modelling these types of relationships.

5.3. New device element type for modelling booms

The first edition of ISO 11783–10 was released in 2009 and featured site-specific application and data logging, but it did not include the concept of section control. Section control is included in the second edition of ISO 11783–10 released in 2015, which also includes guidance on how booms should be modeled. The standard suggests (but does not explicitly require) that booms be modelled as DET objects of either device or function type depending on the modelled device (ISO, 2015a). While these modelling approaches are backward compatible with older DDOPs, it also means that a function DET object could represent either a boom or something entirely different. For this article, it was decided that a new boom device element type is needed for modeling booms accurately. Thus, a new device element type for booms was defined that would allow booms to be modeled more explicitly in the DDOP and modelling rules for this device element type were defined in OPC UA TC.

5.4. New cultural practice identifiers

Currently, eighteen cultural practice identifiers are defined in the ISOBUS data dictionary. They are used by the TC to find the correct subsystem to control when controlling multi-operation implements, but cultural practice identifiers could also be used to match a planned task received from the FMIS with a compatible implement. Because cultural practices by themselves are abstract, defining concrete cultural practices (such as granular fertilizing, liquid crop protection, disk tillage and so on) would allow the operations of an implement to be identified more specifically. Therefore, new cultural practice identifiers shall be defined and the use of cultural practices in matching process between a planned task and an implement shall be tested. Additionally, instead of providing just one cultural practice, implement subsystems could provide a list of cultural practices they are capable of performing, e.g., a bin applying liquid could specify liquid fertilizing, liquid crop protection and irrigation as its compatible cultural practices. OPC UA allows the value rank of a Variable Node to be defined as scalar or one-dimensional array, which is sufficient for representing lists of values.

6. Design of OPC UA for the next Generation ISO 11783 Task Controller

6.1. Modeling principles

In this article, the aforementioned requirements of a high speed replacement of the current ISO 11783 network are fulfilled with a two-step approach: in the first step the current DDOP modeling principle (ISO, 2015a) is extended to cover the requirements for modeling the boom and many-to-many relationships. In the second step this extended-DDOP is mapped to OPC UA modeling. All figures in this section use the notation defined by the OPC Foundation (2017c, Annex C).

6.2. Overview, definitions, modeling of variables and objects

OPC UA TC consists of ObjectTypes and VariableTypes that correspond to device descriptor objects from the ISO 11783 DDOP format as listed in Table 1. The grouping of DPD and DPT objects is done with data container (DTC) objects and the modelling of many-to-many relationships between DET objects is done with resource connector (RSC) objects. Defining a new VariableType for DPT objects was not necessary because the base OPC UA Information model provides one — *PropertyType* —which fulfils the requirements of the DPT object without requiring any further modification.

6.3. DVC objects

DVC objects are represented with instances of *DVCobjectType* as illustrated in Fig. 2.

6.4. DET objects

DET objects are represented with instances of *DETobjectType* as illustrated in Fig. 3.

6.5. DTC objects

A DTC object groups related DPD and DPT objects and identifies its contents with a three-part identifier which consists of: Definition, Unit and Structure Configuration. Definition defines the meaning of the grouped objects, e.g., application rate or work state. Unit defines the SI unit of the grouped objects, e.g., volume per area (m^3/m^2) or time (s). Structure Configuration defines the Roles of grouped DPD and DPT objects in presenting the data, e.g., a configuration consisting of actual value, setpoint value or default value and another configuration consisting of readable state and setpoint state. The use of DTC objects makes processing DDOPs more efficient by allowing the TC to ignore DTC objects with irrelevant contents. all DDEs of ISOBUS data dictionary are able to be remapped to Definitions, Units and Roles. A selection of these mappings are presented in Table 2.

DTC objects are represented with instances of *DTCvariableType*, which is illustrated in Fig. 4 along with an instance of DTCvariableType that demonstrates how DPD and DPT objects are grouped with DTC objects.

Table 1
Device descriptor objects and their corresponding ObjectTypes and VariableTypes both as currently standardized and from the proposed extension to the DDOP model.

Device descriptor object in currently standardized DDOP	Device descriptor object in proposed extended-DDOP	OPC UA TypeDefinitionNode	OPC UA DI Supertype
Device (DVC)		DVCobjectType	DeviceType
Device Element (DET)		DETobjectType	TopologyElementType
Device Process Data (DPD)		DPDvariableType	–
Device Property (DPT)		PropertyType	–
	Data Container (DTC)	DTCvariableType	–
	Resource Connector (RSC)	RSCobjectType	TopologyElementType

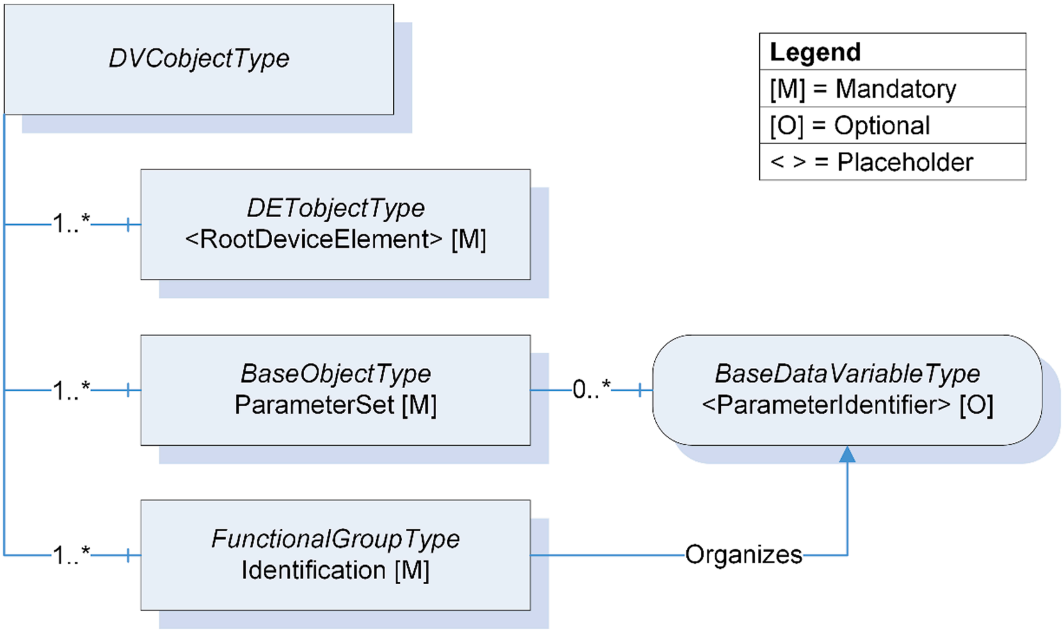

Fig. 2. DVCobjectType defined using the notation of the OPC Foundation.

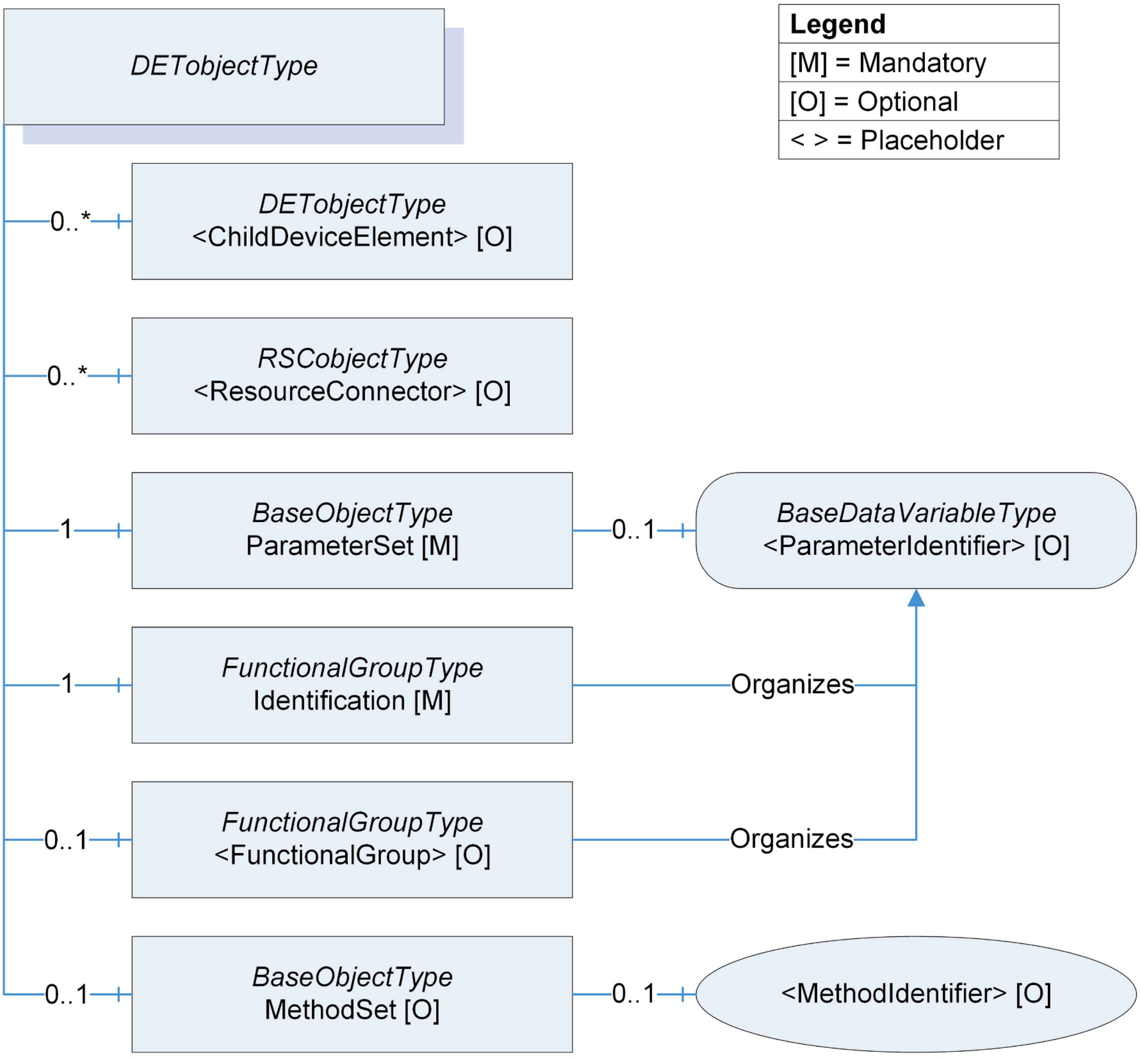

Fig. 3. DETobjectType defined using the notation of OPC UA (OPC Foundation, 2017c, Annex C).

Table 2
Mappings from DDEs to Definition, Role and Unit.

DDI	DDE Name	Definition	Role	Unit
1	Setpoint Volume Per Area Application Rate as [mm^3/m^2]	0: Application Rate	SetpointValue	16: m^3/m^2
2	Actual Volume Per Area Application Rate as [mm^3/m^2]	0: Application Rate	ActualValue	16: m^3/m^2
3	Default Volume Per Area Application Rate as [mm^3/m^2]	0: Application Rate	DefaultValue	16: m^3/m^2
4	Minimum Volume Per Area Application Rate as [mm^3/m^2]	0: Application Rate	MinimumValue	16: m^3/m^2
5	Maximum Volume Per Area Application Rate as [mm^3/m^2]	0: Application Rate	MaximumValue	16: m^3/m^2
6	Setpoint Mass Per Area Application Rate	0: Application Rate	SetpointValue	7: kg/ m^2
7	Actual Mass Per Area Application Rate	0: Application Rate	ActualValue	7: kg/m^2
51	Setpoint Tillage Depth	1: Tillage Depth	SetpointValue	10: m
116	Total Area	13: Area	Total	11: m^2
134	Device Element Offset X	27: Device Element Offset	OffsetX	10: m
141	Actual Work State	29: Work State	ReadableState	0: Undefined
158	Prescription Control State	37: Prescription Control State	SettableState	0: Undefined
161	Actual Condensed Work State (1–16)	29: Work State	ReadableState	0: Undefined
276	Lifetime Fuel Consumption	33: Fuel Consumption	LifetimeTotal	13: m^3
320	Last loaded Weight	95: Loaded Content	LastValue	4: kg

6.6. DPD objects

DPD objects are represented with instances of *DPDvariableType*, which is illustrated in Fig. 5. DPT objects are represented by an instance of PropertyType, which is defined in the OPC UA specification. Neither DPD or DPT objects continue to use DDIs to define the data they model.

6.7. RSC objects

An RSC object connects a boom DET object to a bin DET object. This allows modelling complex relationships between booms and bins that were impossible to model in a tree hierarchy. Additionally, RSC objects allow modelling implements that allow application rates from a bin to booms to be controlled separately for each boom by placing a DTC object that groups DPD and DPT objects related to application rate control under the RSC objects that connect the booms to the bin. While modelling an implement like this is possible by placing the DPD and DPT objects related to application rate control under a function DET object acting as boom, this modelling approach is usable only when no boom is connected to more than one bin, while the use of RSC objects places no limitations on how many bins a boom may be connected to nor how many booms may be connected to a bin.

RSC objects are represented with instances of *RSCobjectType*, which is illustrated in Fig. 6.

6.8. Modelling rules for DDOP in OPC UA TC

Where the DDOP contains DVC, DET, and RSC objects (in OPC UA TC), the attributes of such objects are modelled as Variables belonging to the *ParameterSet* Objects of instances of DVCobjectType, DETobjectType and RSCobjectType respectively. DTC objects belonging to DET and RSC objects are also placed into these ParameterSet Objects.

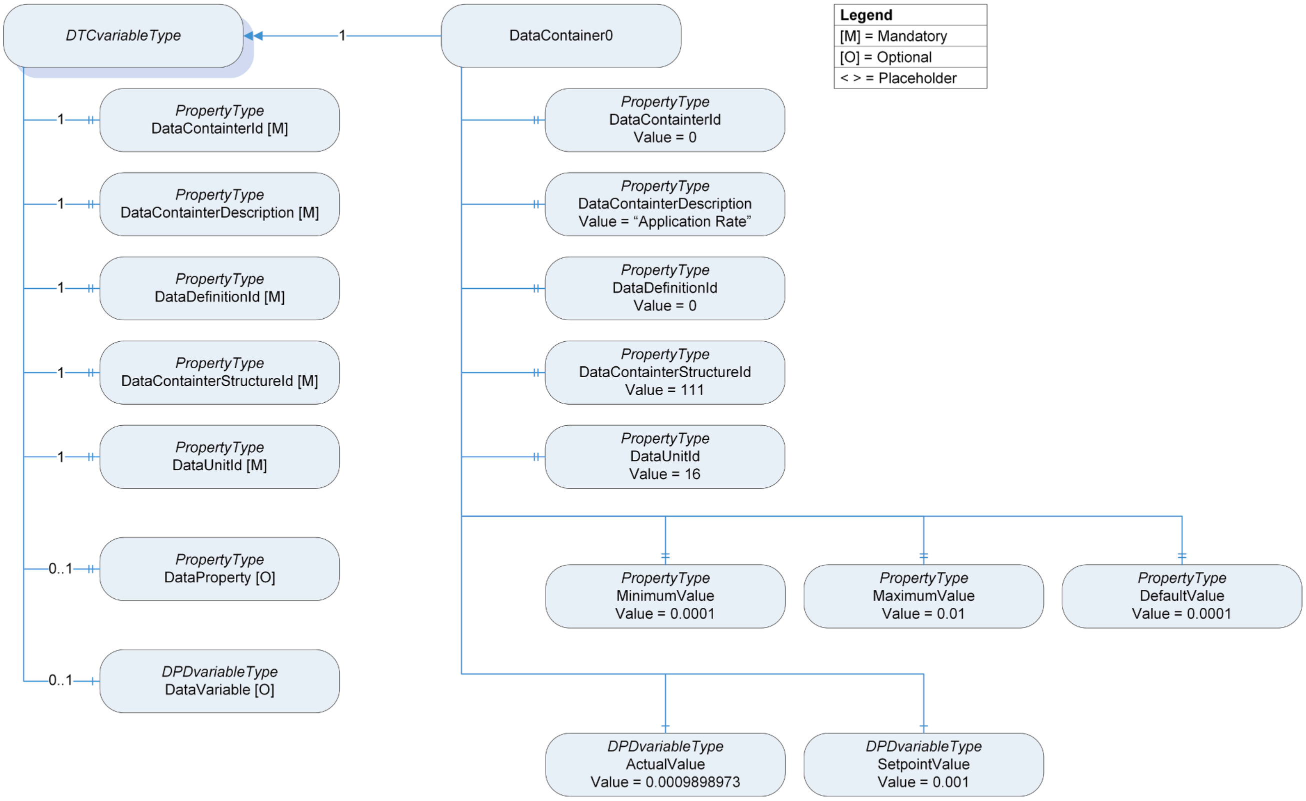

Fig. 4. DTCvariableType and its instance that groups DPD and DPT objects related to application rate. The Definition of the DTC object is defined by *DataDefinitionId*, the Unit of is defined by *DataUnitId* and the Roles of the grouped DPD and DPT objects are specified by *DataContainerStructureId*. Defined using the notation of OPC UA (OPC Foundation, 2017c, Annex C).

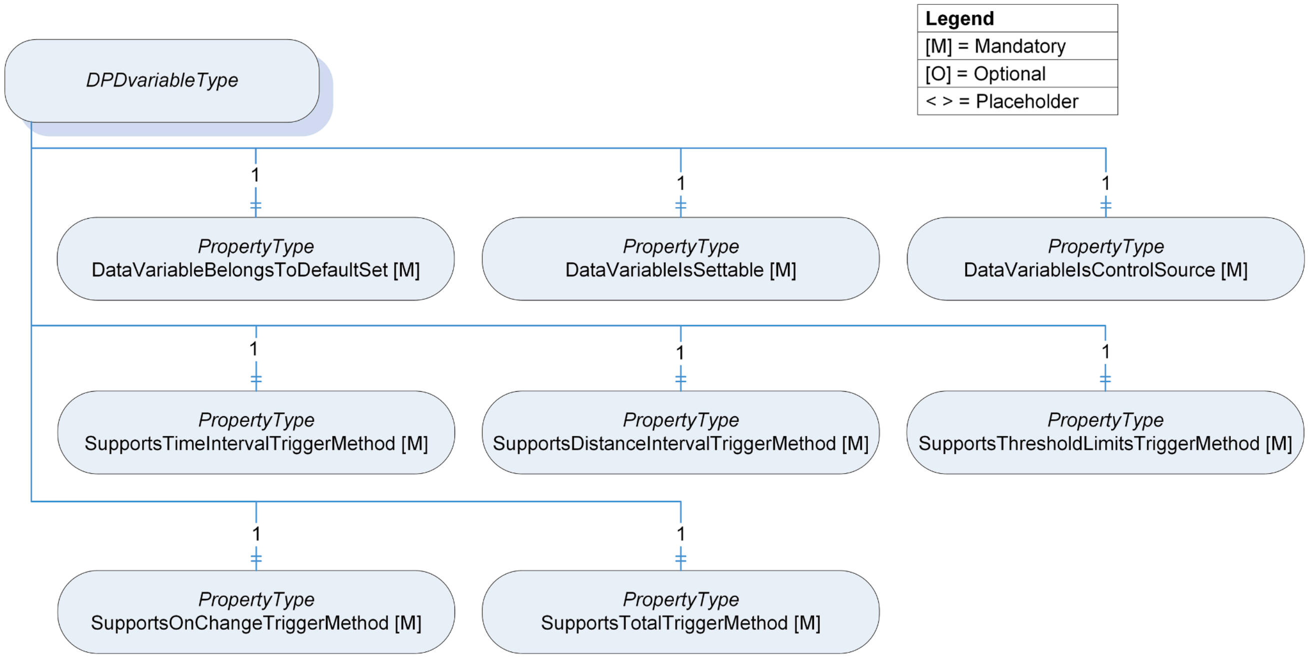
Legend
[M] = Mandatory
[O] = Optional
< > = Placeholder
DPDvariableType
1
PropertyType
DataVariableBelongsToDefaultSet [M]
PropertyType
DataVariableIsSettable [M]
PropertyType
DataVariableIsControlSource [M]
PropertyType
SupportsTimeIntervalTriggerMethod [M]
PropertyType
SupportsDistanceIntervalTriggerMethod [M]
PropertyType
SupportsThresholdLimitsTriggerMethod [M]
PropertyType
SupportsOnChangeTriggerMethod [M]
PropertyType
SupportsTotalTriggerMethod [M]

Fig. 5. DPDvariableType defined using the notation of OPC UA (OPC Foundation, 2017c, Annex C).

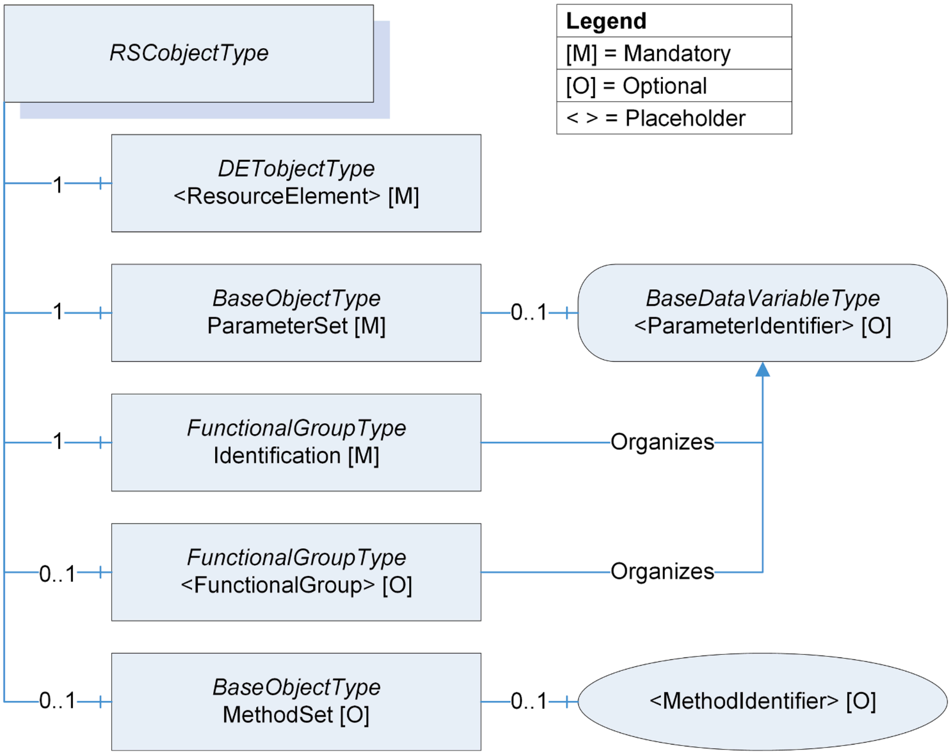

Fig. 6. RSCobjectType defined using the notation of OPC UA (OPC Foundation, 2017c, Annex C).

In addition to device element types defined in ISO 11783–10, the value of the *DeviceElementType* attribute of a DET object may be set to 8 to signify that the DET object is a boom DET object. The modelling rules for this and other device element types are defined in the next subsection.

In DDOP format, the hierarchy of DET objects is expressed with their DeviceElementObjectId and *ParentObjectId* attributes and DET objects were connected to DPD and DPT objects with DOR objects. In OPC UA TC, these relationships are modelled with *HasComponent* References between Nodes representing these device descriptor objects. OPC UA TC uses SI units and it is assumed that the TC can handle necessary unit conversions. Therefore, DVP objects are no longer needed for providing unit-conversion formulae. The names and descriptions of Nodes can be defined in multiple languages with the *LocalizedText* data type, which means that the TC would no longer need to configure the language of an implement.

In addition to exposing process data variables and properties with DTC objects, DET and RSC objects may also provide Methods that allow controlling these variables with predetermined functions. The use of Methods allows the TC to issue more complex control commands than simply writing to values of settable process data variables. For example, a Method for controlling all sections of a boom with a Boolean input array with length equal to the number of sections in the boom would allow the TC to control boom sections either one-by-one or all at once. Methods could also be used to set all total values of a DET object to zero or to set the setpoint application rate to its default, maximum or minimum value.

6.9. Modelling rules for OPC UA for the next Generation ISO 11783 Task Controller

Modelling rules for using OPC UA TC to model implements were defined with focus on implements which apply products, such as seed or fertilizer. These modelling rules define the allowed relationships between different device descriptor objects and between different types of DET objects and required DTC objects for each type of DET object. The modelling rules have been chosen such that implements modelled with these modelling rules will be compatible with the TC-BAS, TC-GEO and TC-SC functionalities defined in ISO 11783–10.

The allowed relationships between different device descriptor objects are illustrated in Fig. 7.

The allowed relationships between different types of DET objects are illustrated in Fig. 8 and information about the various data types is give in Table 3.

Function DET objects are no longer used to represent booms. Instead, booms are represented with boom DET objects. A boom DET object is required to expose its work state, its offset from the center of the device coordinate system, its working width, total area it has treated, total effective and ineffective distance it has travelled, total effective and ineffective time it has spent, its prescription control state, its section control state and its section control turn on and turn off times. Each boom DET object must be parent to at least one section DET object and connected to at least one bin DET object with an RSC object.

To enable modelling booms that allow application rates to be controlled at a section level, the application rate from a bin to such boom should be modelled as a one-dimensional array with length equal to the number of sections in the modelled boom. The DTC object for these application rate controls is placed either under the RSC object that connects the boom and bin DET objects or the bin DET object depending on whether or not the bin DET object is connected to other boom DET objects.

7. OPC UA applications

To test OPC UA TC in practice, an OPC UA Implement Server and OPC UA Task Controller were created. The OPC UA Implement Server utilized the information model built on top of a simulated implement subprogram. Both were designed and developed with Microsoft's Visual Studio 2013 Ultimate (Microsoft, Redmond, USA) and Unified Automation's.NET Based OPC UA Client/Server SDK version 2.6.1.422 (Unified Automation, Kalchreuth, Germany). The code for OPC UA TC was generated with Unified Automation's UaModeler version 1.6.2 (Unified Automation, Kalchreuth, Germany).

7.1. OPC UA implement Server

The OPC UA Implement Server is an OPC UA Server that utilizes OPC UA TC. It consists of three layers that provide values for variables and properties of a simulated implement, an object-based DDOP that models the implement, and an AddressSpace based on the DDOP. These layers are illustrated in Fig. 9.

The OPC UA Implement Server was used with multiple different simulated implements to test OPC UA TC in different use cases. The simulated implements were a sprayer and a seed drill as described in Table 4. Simulated delays were added when responding to new setpoints for section work states and application rates received from Clients.

7.2. OPC UA Task Controller

The OPC UA Task Controller is a proof-of-concept TC with a focus on the features related to the TC interacting with implements. The main duties of the OPC UA Task Controller are:

a) Connecting to an implement via an OPC UA Session
b) Transferring the implement's DDOP to local memory with *Browse* and *Read* Services
c) Performing a compatibility check for a selected task and the connected implement
d) Controlling the implement based on the selected task and a simulated tractor position with *Write* and *Call* Services
e) Logging data from the implement with *Subscription* and *MonitoredItem* Services.

The software components of the OPC UA Task Controller are illustrated in Fig. 10. The OPC UA Task Controller represents the main task controller and the OPC UA Implement Server represents the implement.

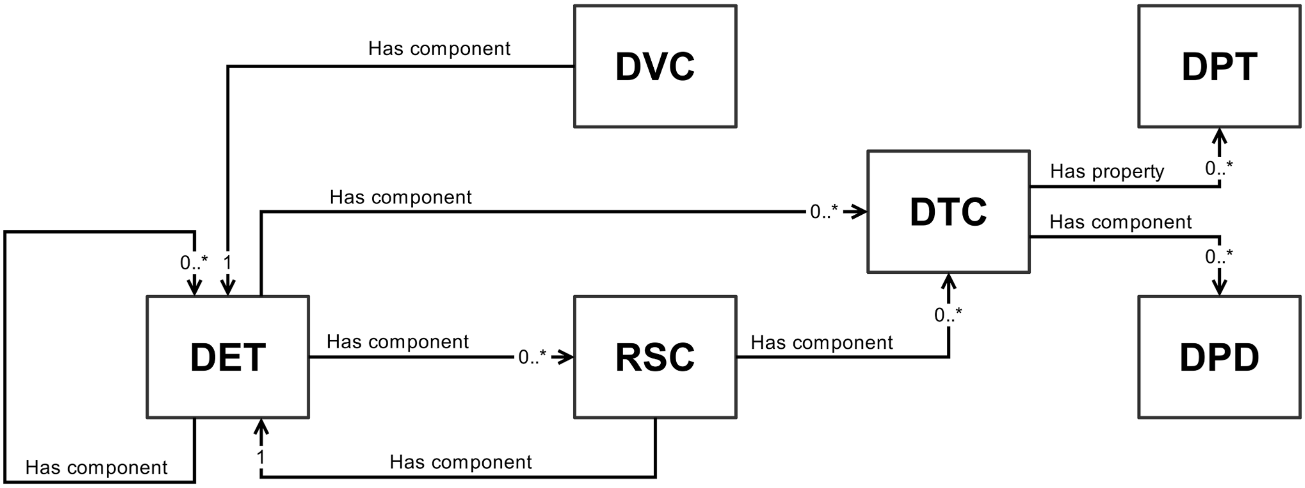

Fig. 7. Allowed relationships between different device descriptor objects. Arrows point from a parent to a child and specify the ReferenceType used to connect the Nodes representing these device descriptor objects; the numbers indicate how many such children the parent may have.

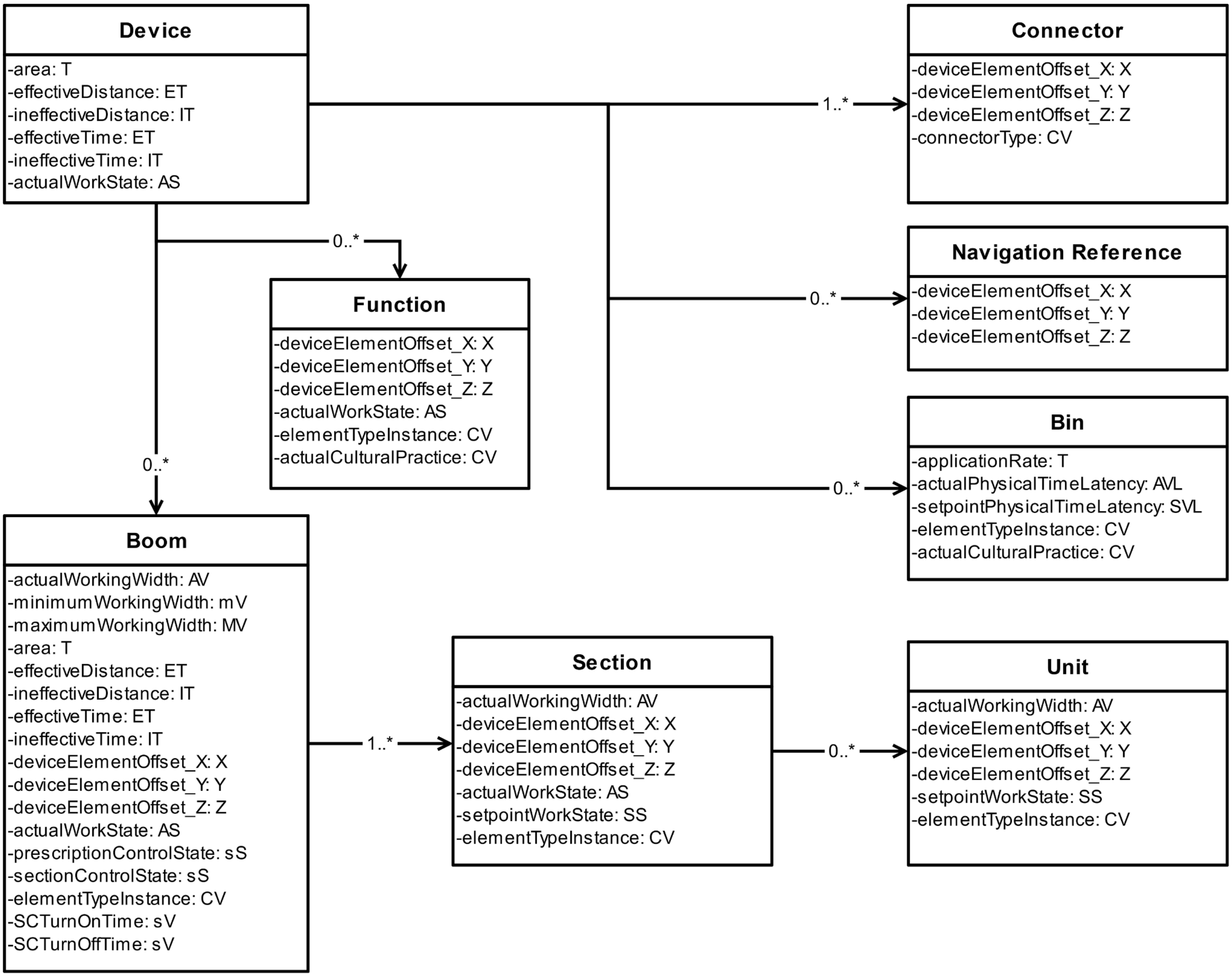

Fig. 8. Allowed relationships of different types of DET objects, described using UML convention. The DET objects are represented as containers that provide a list of their DTC objects and in parenthesis are the Roles that form the Structure Configuration of the DTC object. The notation for the Roles is defined in Table 3.

The OPC UA Task Controller has various features that would be expected from a TC such as: a Command Algorithm to command the sections on/off, a Task File, (simulated) Position reader, and a data logger. The most notable differences of the OPC UA TC are that the DDOP consists of OPC UA nodes as previously discussed.

Once the OPC UA Task Controller has been started, it prompts the user for the *Endpoint* URL of the OPC UA Implement Server. After being connected to the simulated implement represented by the OPC UA Implement Server, the OPC UA Task Controller creates a local copy of the DDOP exposed by the OPC UA Implement Server. This local DDOP is then optimized for more efficient processing by creating a *Virtual Device* that models the connected implement sufficiently for controlling it.

After the initialization of the local DDOP and the Virtual Device, the user is allowed to select a task and the OPC UA Task Controller checks whether or not the selected task and the connected simulated implement are compatible with each other. For this article, tasks were simplified to lists of products to be applied. Each product has a prescription map, a cultural practice and an application rate unit that are compared against cultural practices and application rate units of bins of the connected implement instead of comparing NAME of the implement to a NAME provided in task data. If a compatible bin can be found for each product, the selected task and the connected implement are compatible.

After a successful compatibility check, the user is allowed to start the task. Once the task has been started, the OPC UA Task Controller performs the following actions: (a) controls application rates based on prescription maps of applied products (b) turns sections on and off based on the coverage map approximated with Bresenham’s line algorithm (Bresenham, 1965) and (c) logs the position of the simulated tractor, the current time and data from the simulated implement. The command algorithm of the OPC UA Task Controller is illustrated in Fig. 11.

8. Results

8.1. Modelling implements with OPC UA TC

OPC UA TC enables the modelling of implements that are not possible to model with the DDOP format as standardized in the current most recent version of ISO 11783. Fig. 12 demonstrates how a product-applying implement with multiple bins and multiple booms can be modelled in OPC UA TC. In the figure, DET and RSC objects are represented with rounded rectangles that provide a list of their DTC objects with the Roles that form the Structure Configuration of the DTC object listed in parenthesis. The notation for the Roles is explained in Table 3. Solid arrows connect a parent DET object to its child DET or RSC object and dotted arrows connect an RSC object to its resource DET object. To simplify the diagram, unit DET objects are not included.

In the modelled implement from Fig. 12:

- Boom 1 is applying products from bins 1 and 2
- Boom 2 is applying products from bins 1 and 3

Table 3
Notations for Roles.

Notation	Role	Represents
AV	Actual value	A measured value of a physical process or quality
SV	Setpoint value	A value that is used for controlling a physical process or quality
MV	Maximum value	A maximum value of a physical process or quality
mV	Minimum value	A minimum value of a physical process or quality
DV	Default value	A default value of a physical process or quality
sV	Settable value	A settable value of a physical process or quality
CV	Constant value	A constant value of a physical process or quality
T	Total	A resettable total value of a physical process
LT	Lifetime Total	A non-resettable total value of a physical process
ET	Effective total	A resettable total value of a physical process accumulated when the device has been working
IT	Ineffective total	A resettable total value of a physical process accumulated when the device hasn't been working
AS	Actual state	A measured value of the state of a device or a process
SS	Setpoint state	A setpoint state that affects another state of a device or a process
sS	Settable state	A state of a device or a process that can be read and written
X	X-offset	An offset along the x-axis
Y	Y-offset	An offset along the y-axis
Z	Z-offset	An offset along the z-axis
SVL	Setpoint value latency	The time lapse between receiving and applying a setpoint value
AVL	Actual value latency	Represents the time lapse between communicating an actual value to the task controller and applying it

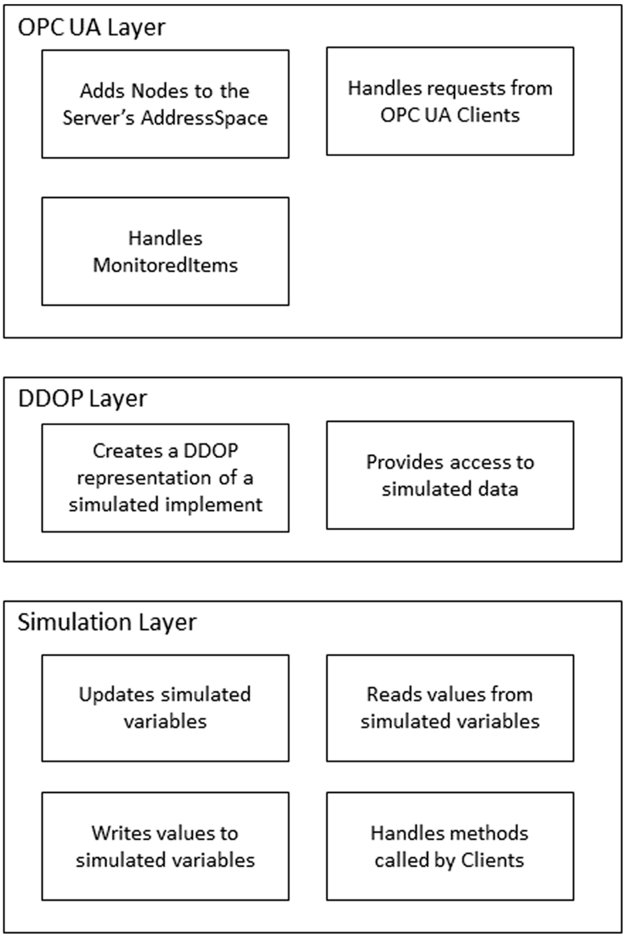
OPC UA Layer
Adds Nodes to the Server's AddressSpace
Handles requests from OPC UA Clients
Handles MonitoredItems
DDOP Layer
Creates a DDOP representation of a simulated implement
Provides access to simulated data
Simulation Layer
Updates simulated variables
Reads values from simulated variables
Writes values to simulated variables
Handles methods called by Clients

Fig. 9. Layers of the OPC UA Implement Server.

Table 4
Simulated implements.

	Sprayer	Seed drill
Number of booms	4	1
Sections per boom	200	400
Number of bins	4	6
Total number of control channels	4	2400
Cultural practices of bins	One for fertilizing, three for crop protection	Seeding
Relationships between booms and bins	One-to-one	One-to-many
Application rate control level	Boom	Section

- Boom 3 is applying products from bins 2 and 3

The original DDOP format does not provide means for modelling such complex relationships between bins and booms, which demonstrates the benefits of enhancing the DDOP format with RSC objects.

The figure shows how the model facilitates complex relationships between booms and bin. In addition, it is shown how RSC and DTC objects enable the modelling of implements in an enhanced way, allowing the application rate from a bin to connected booms be controlled separately for each bin-boom pair. As shown in Fig. 12, application rate DTC objects are present in both RSC objects as well as bin DET objects; in the of RSC objects these consist of: the actual value, setpoint value, default value, minimum value and maximum value; while in bin DET objects these consist of just the total. Both the actual value and the setpoint value may be either scalars or one-dimensional arrays. This enables implements that allow application rates to be controlled at a section level to be modeled, without changing the hierarchical structure of DET, RSC and DTC objects.

8.2. *Testing the OPC UA Task Controller*

8.2.1. *Test setup*

The OPC UA Task Controller and the OPC UA Implement Server were tested by running the applications on two separate Windows 10 PCs connected with via Ethernet cable. The data logged by the OPC UA Task Controller during the test runs was analyzed with MATLAB R2018b (Mathworks, USA) to evaluate the correctness of the command algorithm with regards to site-specific application and section control.

When a task is being executed, the OPC UA Task Controller provides simulated tractor positions based on predetermined paths illustrated in Fig. 13 and Fig. 14. These paths have been chosen such that they involve returning to a treated area in 90° and 45° angles. In the former case, all sections should be turned off at the same time. In the latter case, sections should be turned off one by one.

8.2.2. *Site-specific application*

Tractor positions were read from the logged data to calculate the positions of the booms. The setpoint application rates from bins to booms were read from the logged data and compared with the calculated positions to verify the correctness of the command algorithm in site-specific application.

Fig. 15 shows the cell grid prescription map for the simulated sprayer. The prescribed application rates range from 100 cl/m^2 to approximately 2000 cl/m^2. In the prescription map, the width and the height of a cell are 0.25 m.

Fig. 16 shows the logged setpoint application rate of the simulated sprayer and the setpoint application of the prescription map. The setpoint application rates match each other.

Fig. 17 shows the absolute value of the difference between logged and prescription setpoint application rates for the simulated sprayer. The maximum for this difference is less than 3 cl/m^2.

Fig. 18 shows the cell grid prescription map for the simulated seed

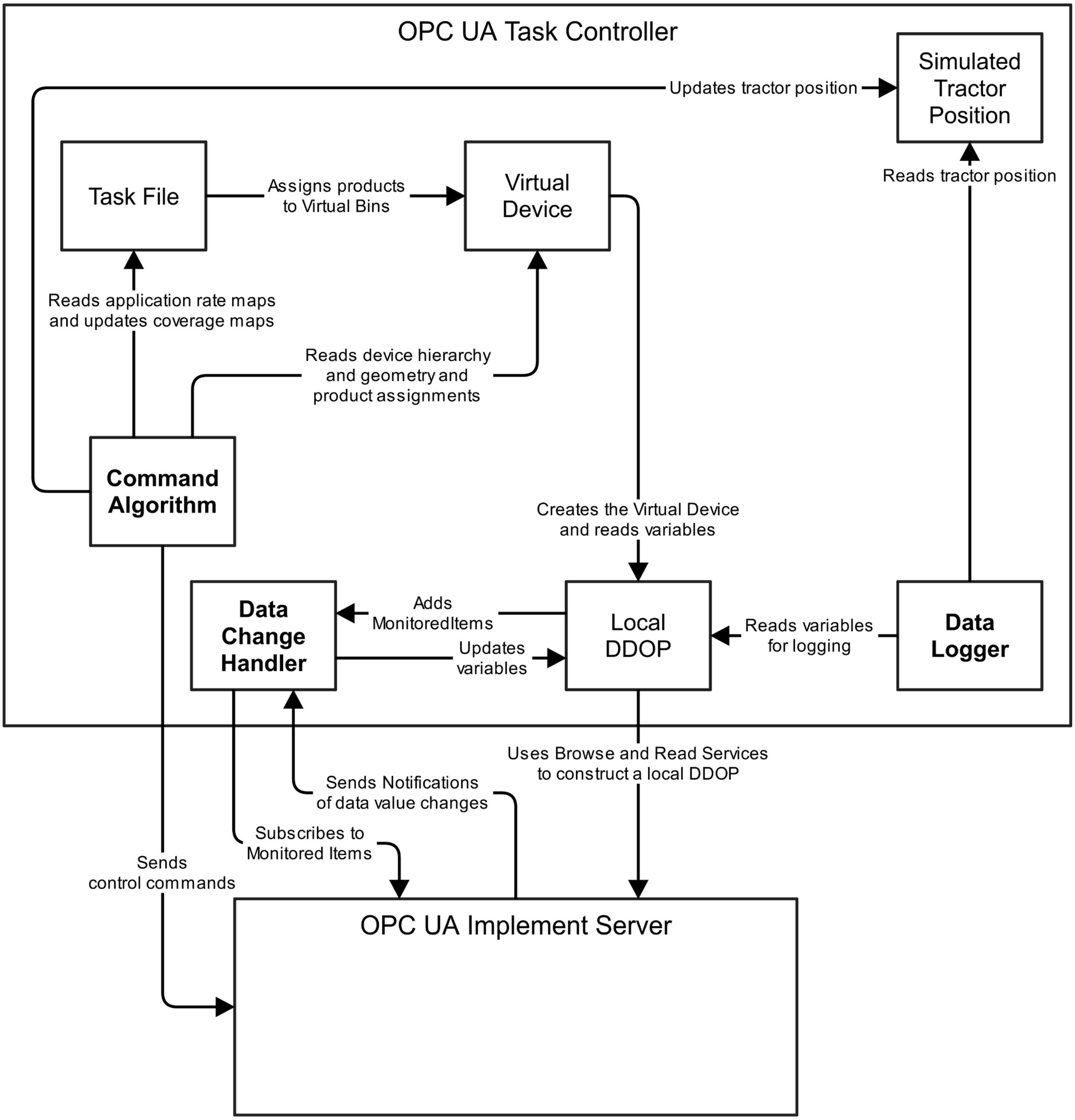
OPC UA Task Controller
Simulated Tractor Position
Updates tractor position
Reads tractor position
Task File
Assigns products to Virtual Bins
Virtual Device
Reads application rate maps and updates coverage maps
Reads device hierarchy and geometry and product assignments
Command Algorithm
Creates the Virtual Device and reads variables
Data Change Handler
Adds MonitoredItems
Updates variables
Local DDOP
Reads variables for logging
Data Logger
Uses Browse and Read Services to construct a local DDOP
Sends Notifications of data value changes
Subscribes to Monitored Items
Sends control commands
OPC UA Implement Server

Fig. 10. The components of the OPC UA Task Controller. The components in bold typeface are threads and other components are objects. The arrows point from the component that initiates the interaction to the participating components.

drill. The prescribed application rates range from 1 g/m^2 to approximately 10 g/m^2. In the prescription map, the width and the height of a cell are 0.125 m.

Fig. 19 shows the logged setpoint application rate of the simulated seed drill and the setpoint application of the prescription map. Again, the setpoint application rates match each other.

Fig. 20 shows the absolute value of the difference between logged and prescription setpoint application rates for the simulated seed drill. Again, the maximum for this difference is less than 0.03 g/m^2.

The errors in the applications presented are so small that the OPC UA Task Controller is considered to be performing site-specific application correctly.

8.2.3. Section control

Tractor positions were read from the logged data to calculate boom positions. The working widths of the booms were read from the logged data and compared with the calculated positions to verify the correctness of the section control command algorithm. Fig. 21 shows the working width of the boom of the simulated seed drill based on its position. The plotted working width shows that sections were turned on and off correctly.

Fig. 22 shows the coverage map created by the Bresenham's line algorithm for the simulated seed drill. Gray areas were not marked as treated and green areas were marked as treated. A tiny gap after exiting the treated areas can be seen on the map which shows the functioning of the section as they turn on late to avoid overlapping application areas.

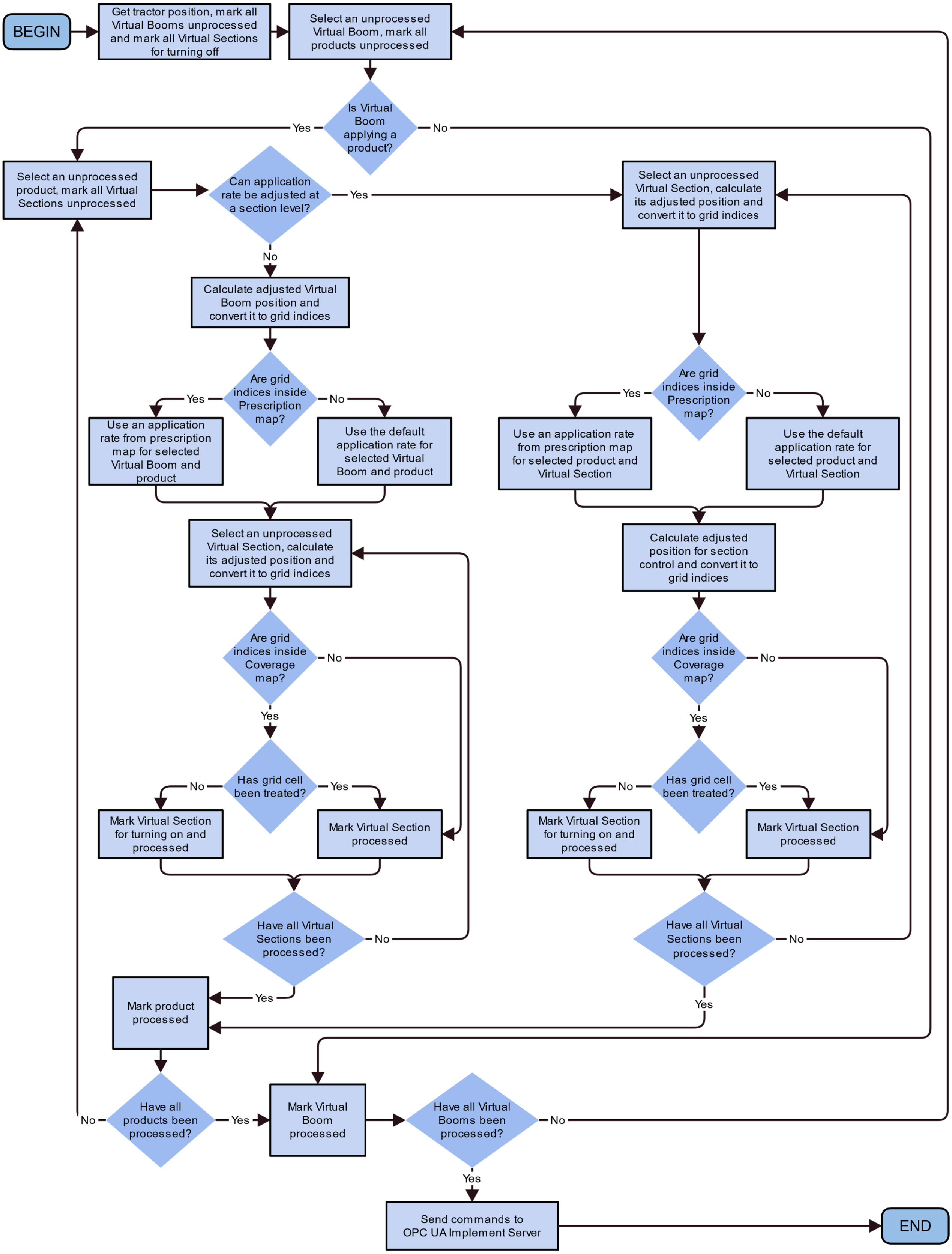
BEGIN
Get tractor position, mark all Virtual Booms unprocessed and mark all Virtual Sections for turning off
Select an unprocessed Virtual Boom, mark all products unprocessed
Is Virtual Boom applying a product?
Yes
No
Select an unprocessed product, mark all Virtual Sections unprocessed
Can application rate be adjusted at a section level?
Yes
No
Select an unprocessed Virtual Section, calculate its adjusted position and convert it to grid indices
Calculate adjusted Virtual Boom position and convert it to grid indices
Are grid indices inside Prescription map?
Yes
No
Use an application rate from prescription map for selected Virtual Boom and product
Use the default application rate for selected Virtual Boom and product
Are grid indices inside Prescription map?
Yes
No
Use an application rate from prescription map for selected product and Virtual Section
Use the default application rate for selected product and Virtual Section
Select an unprocessed Virtual Section, calculate its adjusted position and convert it to grid indices
Calculate adjusted position for section control and convert it to grid indices
Are grid indices inside Coverage map?
No
Yes
Are grid indices inside Coverage map?
No
Yes
Has grid cell been treated?
No
Yes
Mark Virtual Section for turning on and processed
Mark Virtual Section processed
Has grid cell been treated?
No
Yes
Mark Virtual Section for turning on and processed
Mark Virtual Section processed
Have all Virtual Sections been processed?
No
Yes
Have all Virtual Sections been processed?
No
Yes
Mark product processed
Have all products been processed?
No
Yes
Mark Virtual Boom processed
Have all Virtual Booms been processed?
No
Yes
Send commands to OPC UA Implement Server
END

Fig. 11. Command algorithm of the OPC UA Task Controller.

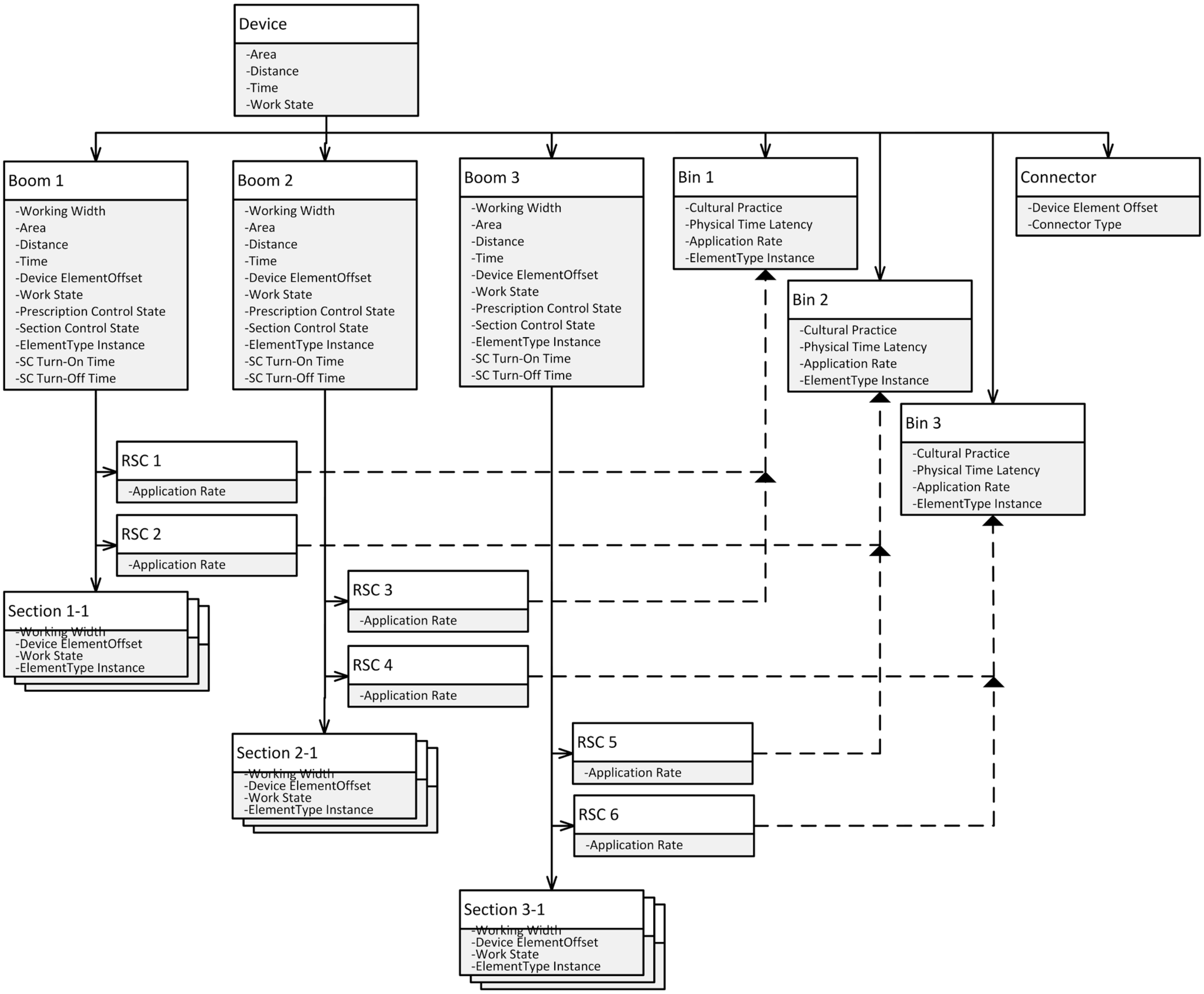

Fig. 12. Implement with three booms and three bins.

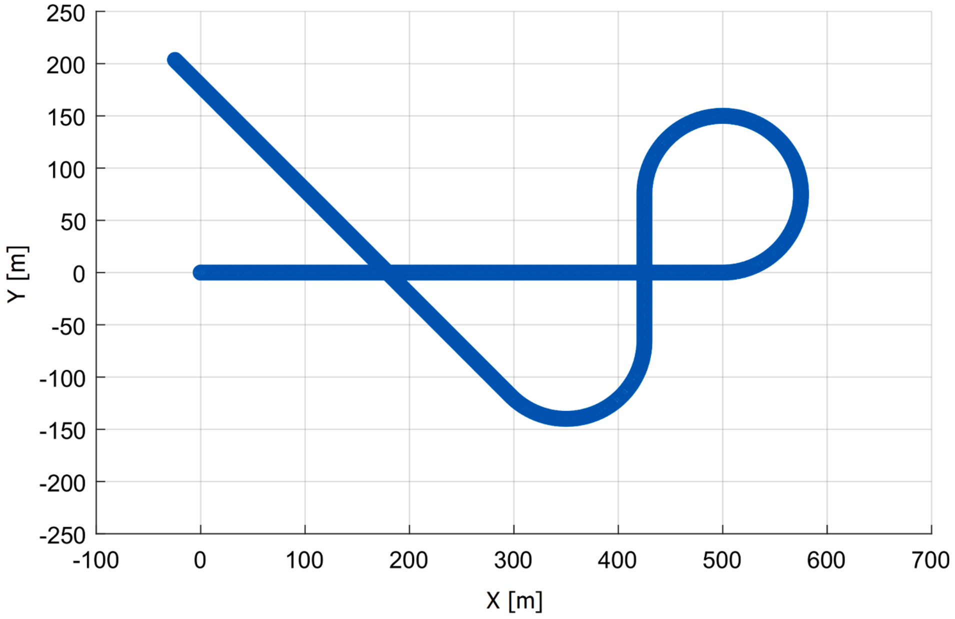

Fig. 13. The path for simulated sprayer.

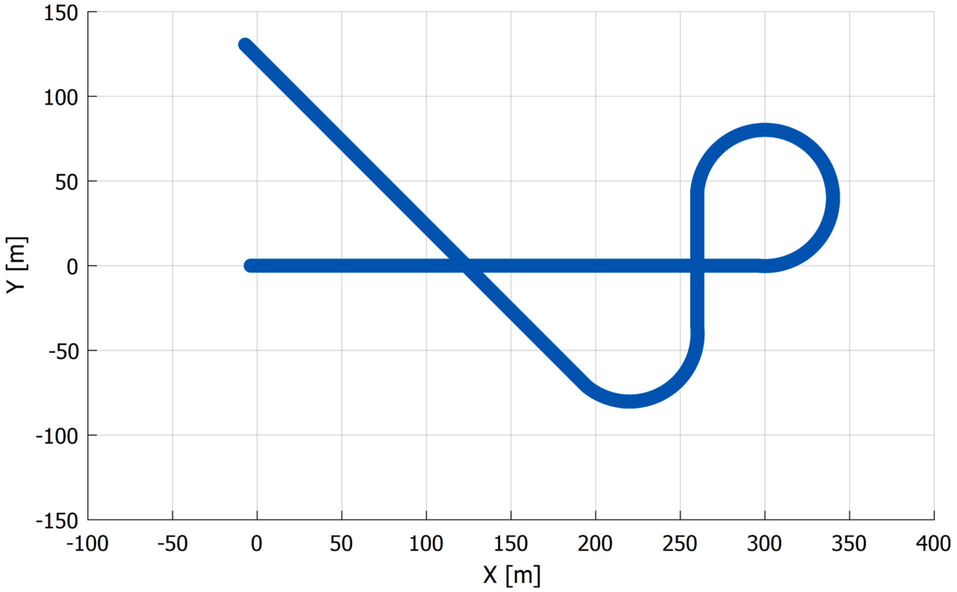

Fig. 14. The path for simulated seed drill.

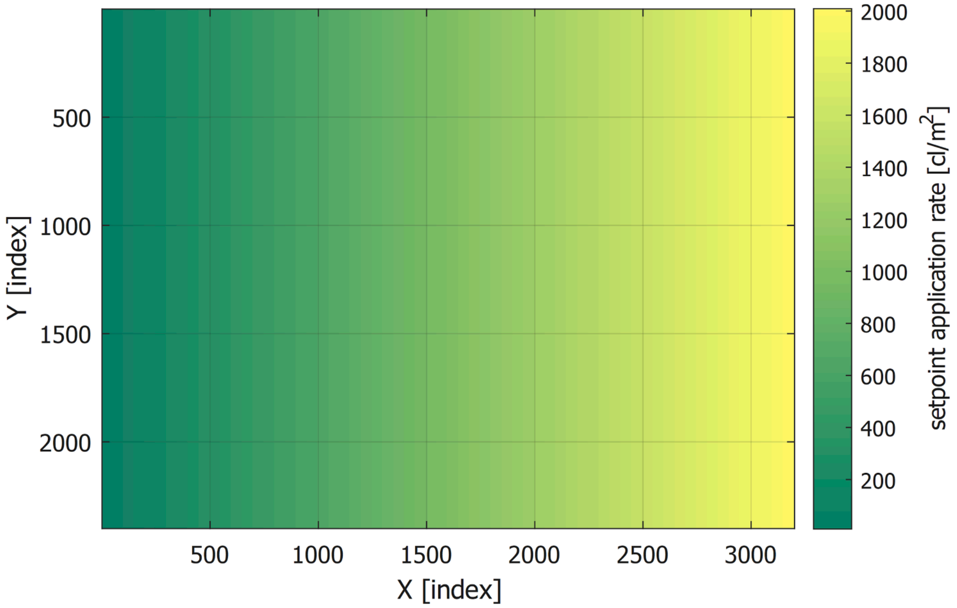

Fig. 15. Prescription map for the simulated sprayer.

Fig. 23 shows the working width of the first boom of the simulated sprayer compared to its position based on data logged when the simulated tractor was moving at 2 m/s and at 5 m/s. This plot shows that the command algorithm was able to adjust the timing of turning sections on and off based on the speed of the tractor.

Fig. 24 shows the coverage map created by the Bresenham's line algorithm for the simulated sprayer. Grey areas were not treated, dark green areas were marked as treated when moving at both 2 m/s and 5 m/s and light green areas were marked as treated only when moving at 2 m/s. While the command algorithm is not directly affected by the speed of the simulated tractor, the coverage maps drawn by Bresenham's line algorithm became less accurate as the speed of the simulated tractor was increased, which could affect the command algorithm.

8.2.4. Response time

The TC client/server setup was tested in laboratory conditions using two Windows 10 PCs directly connected to each other via a single Ethernet 1000BASE-TX crossover cable. The OPC UA implementation was based on the.NET SDK (Unified Automation, Kalchreuth, Germany). The performance of Write and Call Services used in the command algorithm (as shown in Fig. 11) were evaluated by analyzing the time intervals between OPC UA request and response Messages with Wireshark network protocol analyzer (The Wireshark Foundation, Davis, California, USA). Fig. 25 below shows how the number of products affects the time it takes for the OPC UA Task Controller to exchange data with the OPC UA Implement Server per iteration of the command algorithm.

The proposed OPC UA implementation presented in this paper is shown to take 2.8 ms on average (see Fig. 25) to control 6 products in each of 200 sections. In the same 2.8 ms it could be expected that ISO 11783 would only be able to control 80 (16 sections multiplied by 5 messages which could be sent in 2.5 ms) sections or update the rate of

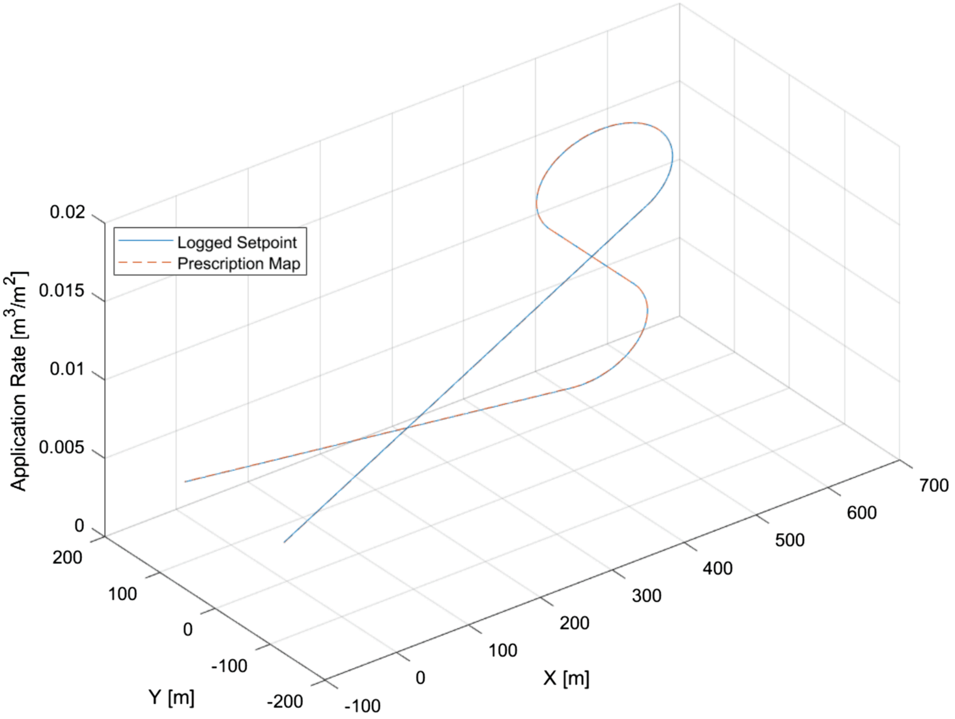

Fig. 16. Logged and prescription setpoint application rates of the simulated sprayer.

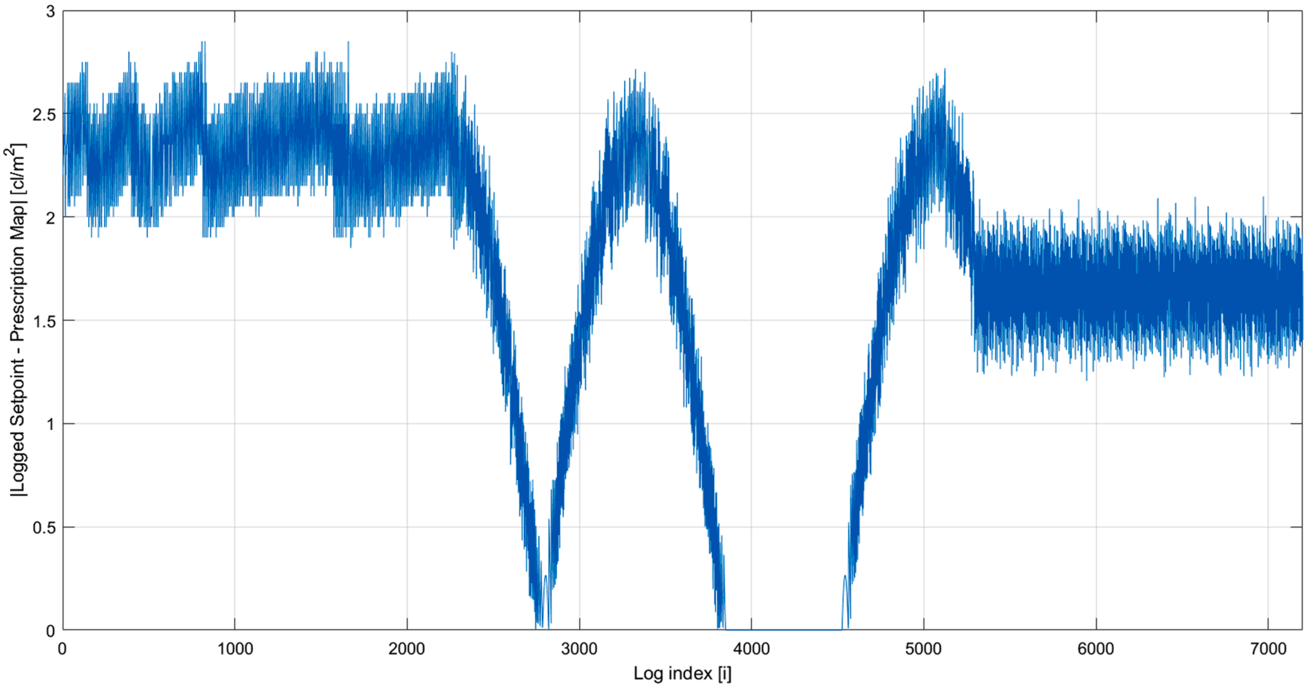

Fig. 17. Difference of logged and prescription setpoint application rates of the simulated sprayer.

one product.

9. Discussion

When work on this article began in March 2019, the specification for the OPC UA PubSub communication model had been released approximately-one year earlier. To the knowledge of the authors, there were no OPC UA PubSub SDKs available at the time and it was decided that the OPC UA Task Controller and the OPC UA Implement Server would follow the OPC UA Client-Server communication model. Support for the OPC UA PubSub communication model has now been added to some OPC UA SDKs and its potential benefits in data exchange between the TC and implements should be studied in future research. As discussed in Chapter 4, the SDKs which implement OPC UA can have a notable impact on performance.

During the research of this article, a new version (1.02) of OPC UA DI was released and its status was changed from *companion specification* to *specification*. This new version enhances the device model with new

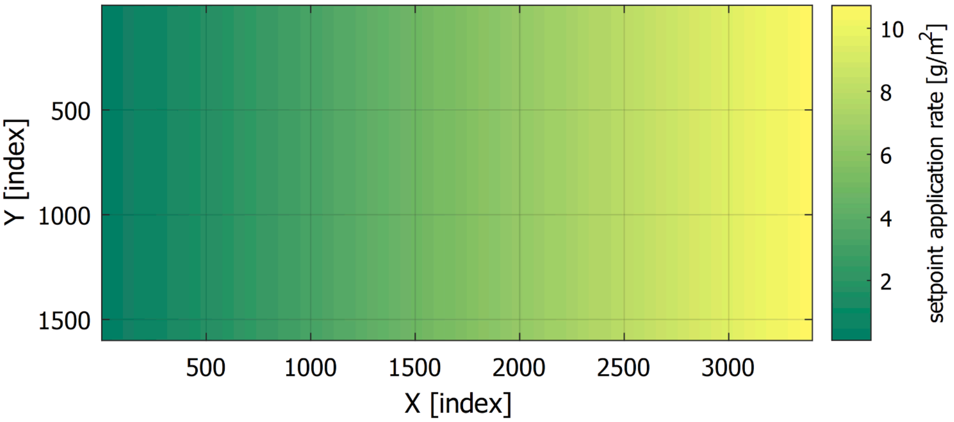

Fig. 18. Prescription map for the simulated seed drill.

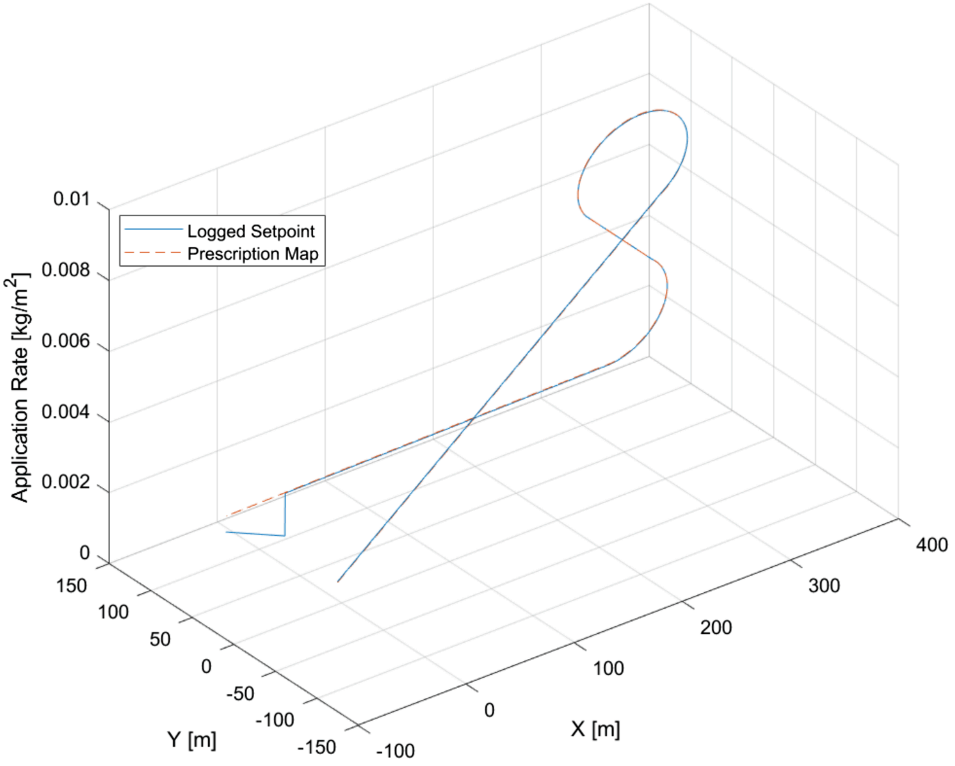

Fig. 19. Logged and prescription setpoint application rates of the simulated seed drill.

features, such as *ComponentType* ObjectType for modelling individual components of a device, and utilizes the *Interface Model* defined in OPC UA Amendment 7 (OPC Foundation, 2019). The Interface Model allows defining ObjectTypes as composites of multiple different *Interfaces*, which enables multiple inheritance. Naturally, it takes time before new features such as the Interface Model become available in OPC UA development tools and thus OPC UA TC is based on version 1.01 of OPC UA DI supported by the latest version of the UaModeler tool.

While the OPC UA Implement Server was designed with simulated implements in mind, its layered structure should allow the simulation layer to be replaced with the interface of a real implement which would in turn allow the upper layers to read and write live data to/from the implement. Modifying the OPC UA Implement Server to be compatible with real implements would be an interesting topic for future research.

OPC UA allows Variables to have both scalar and multidimensional array *Values*. OPC UA TC utilized one-dimensional arrays when modelling application rates that could be controlled at a section level. One-dimensional arrays of fixed length could also be used to model device element offsets, angles and dimensions. More interestingly, arrays of variable length could be used as lists. For example, instead of bins providing a single cultural practice they are compatible with, they could provide a list of all cultural practices they are compatible with. The use of arrays could also make it more efficient to model sections of a boom with a single DET Object that provides values for variables and properties of all sections as arrays.

As mentioned earlier in the article, there is a need to allow many-to-many connections between booms and bins. This could be leveraged effectively for example, in sprayers which may have the functionality of broadcast spraying and precision spot-spraying simultaneously. Depending on the products used, the booms will want to select different bins depending on the situation.

This article focused on designing an OPC UA information model for data exchange between the TC and implements. In future research, OPC UA information models for other CFs of ISO 11783 network, such as

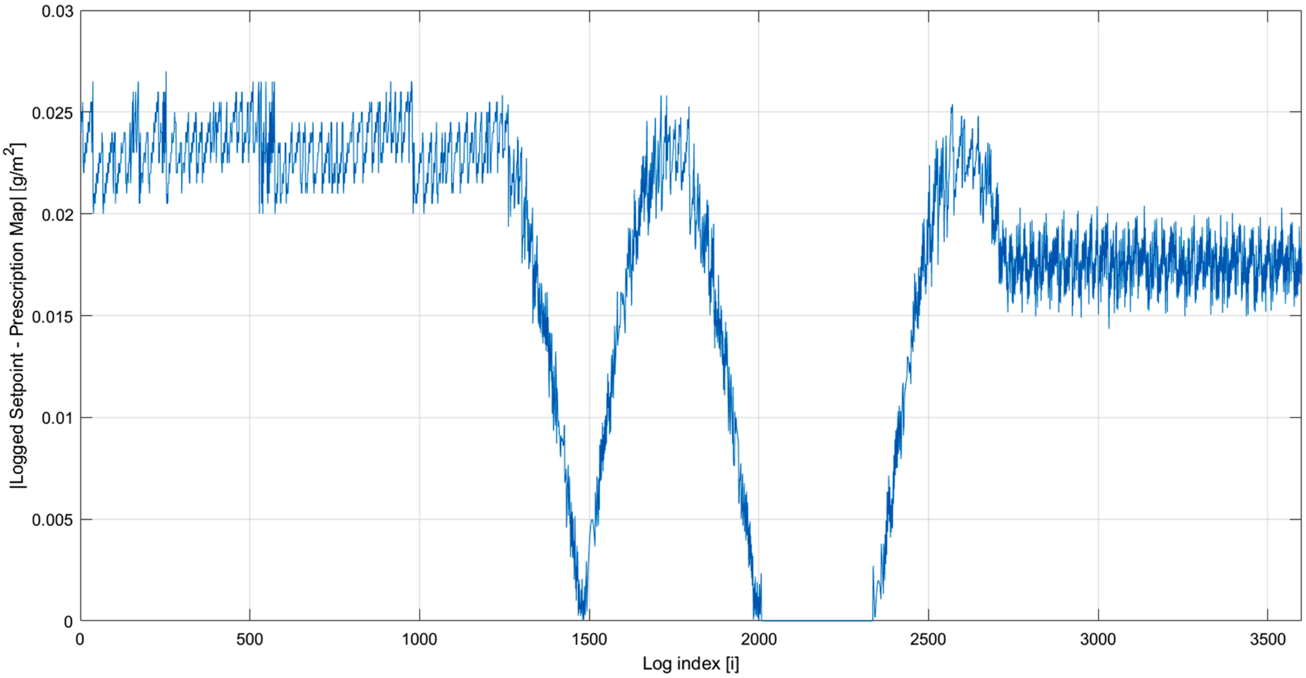

Fig. 20. Difference of logged and prescription setpoint application rates of the simulated seed drill.

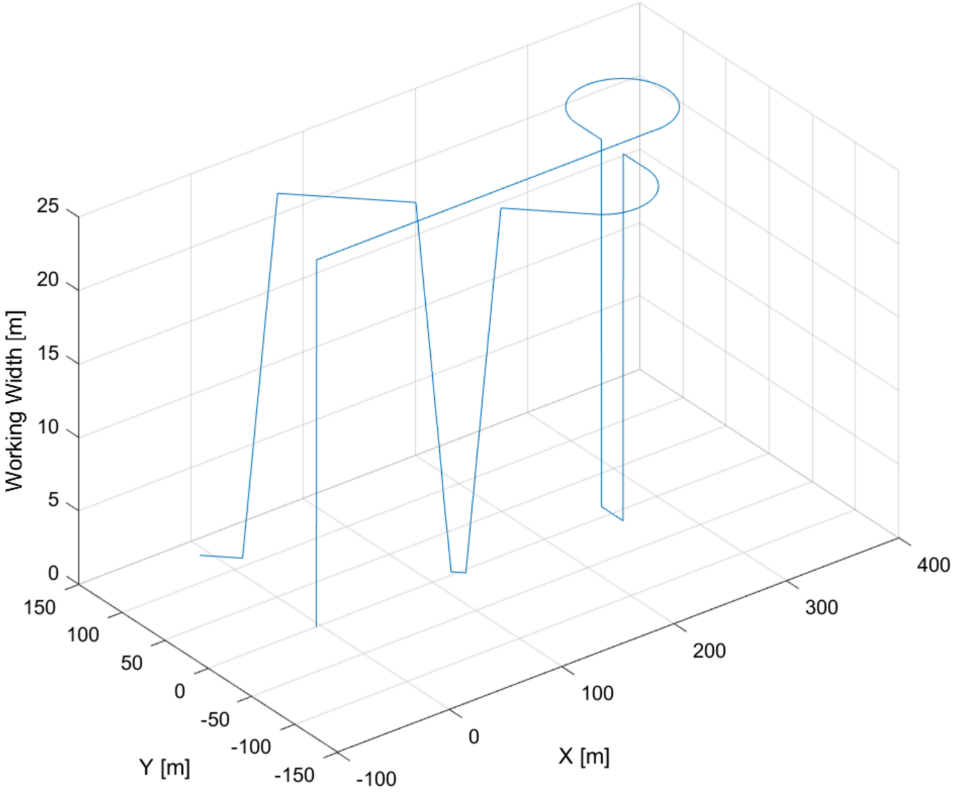

Fig. 21. The working width of the boom of the simulated seed drill.

TECUs, could be designed. Additionally, the use of OPC UA PubSub communication model in data exchange between the TC and implements should be tested.

In the Results chapter of this paper the performance of a simulated implement is presented and analyzed. This type of implement is not possible with current TCs. Using CAN 2.0B extended frame format at 250 kbit/s, a frame takes approximately 500 μs to send and in the current ISO 11783 protocol, each frame can hold the section control information (on/off) of 16 sections or one single rate-control value. To control the rate and section control state of 6 products on 200 sections (as in Table 4) the ISO 11783 standard would require 1275 messages (200 rate messages per product and 75 messages to command the sections on/off) which would take 637.5 ms. Therefore, this type of machine control scenario would not be feasible currently due to both, bandwidth

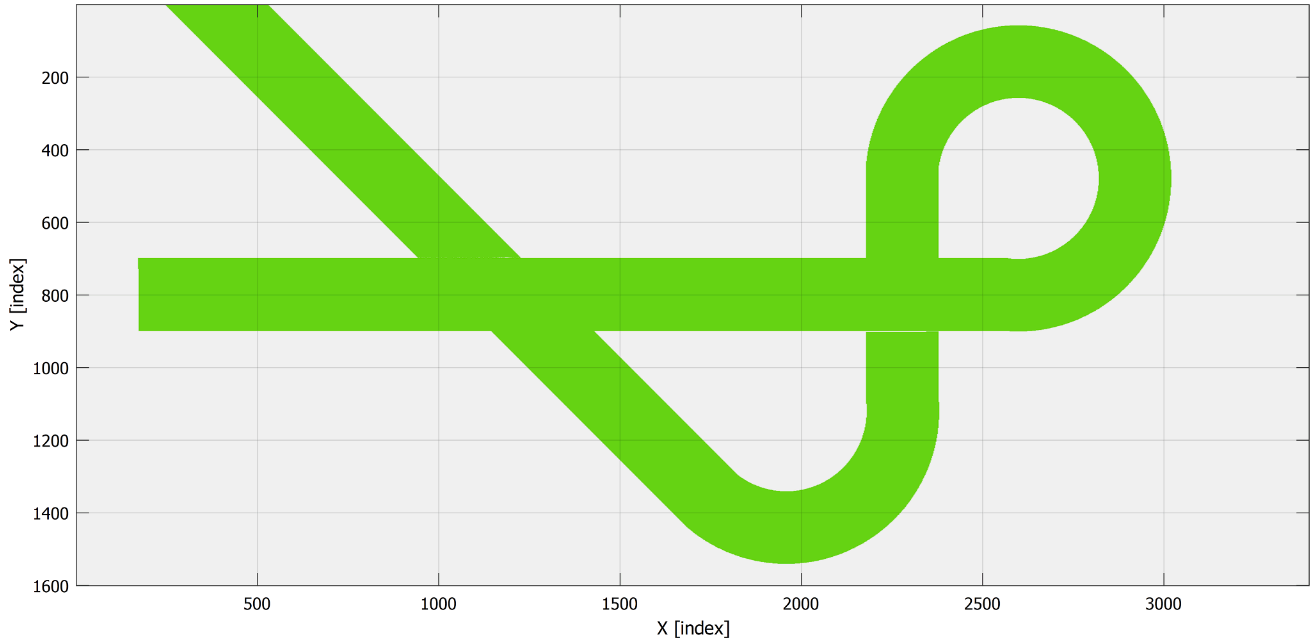

Fig. 22. The coverage map for the simulated seed drill.

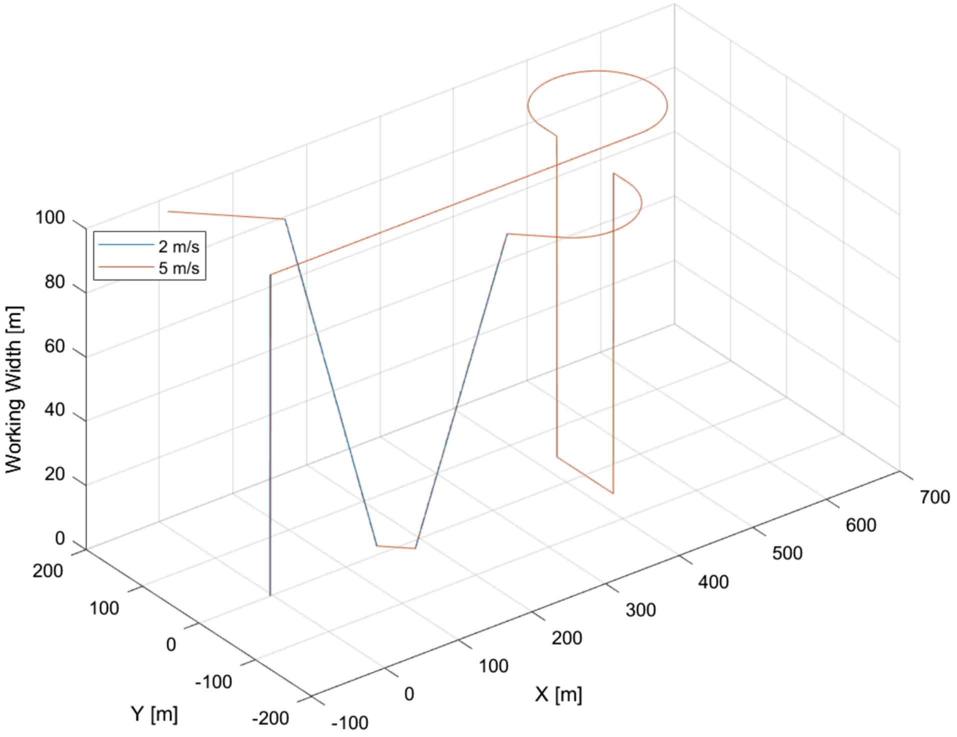

Fig. 23. The working width of the first boom of the simulated sprayer.

constraints, and requirements relating to the update rate. This speed benefit mostly comes from the lower-level upgrade from a CAN-based to an Ethernet-based network, while the focus of this work is not on the lower layers, but the higher-level presentation and application OSI layers which will be in need of updating given the decision to use Ethernet in a future high-speed network.

Limitations and future research directions.

A limitation of this work is that it was performed in a simulated environment on (comparatively) powerful PC hardware, whereas the real-life use-case for such a network would be in less powerful embedded hardware and via multiple switches, over wires which may have some low level of interference.

Future research could take several directions. Firstly, network properties such as latency, processing load etc. should be investigated, especially for different typical tractor-implement network configurations and different Ethernet switches.

This work explored using OPC UA for TC control algorithms, however the full ISO 11783 range includes general purpose ECU messages (speed, engine RPM etc.) as well as diagnostics connections and VTs (GUI display units). Research should be conducted into how best to

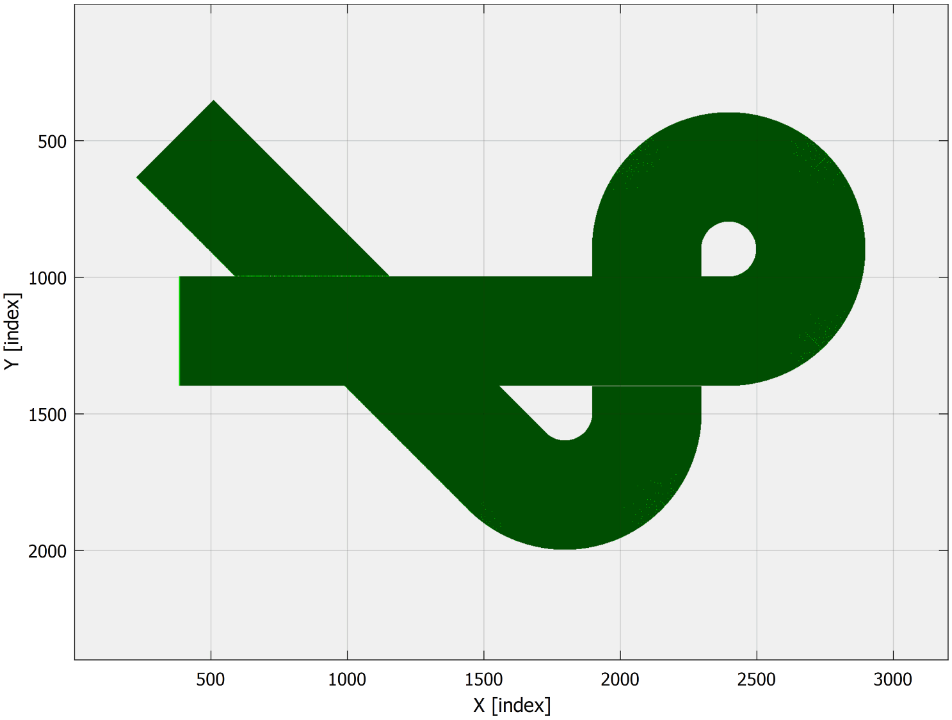

Fig. 24. The coverage maps of the simulated sprayer.

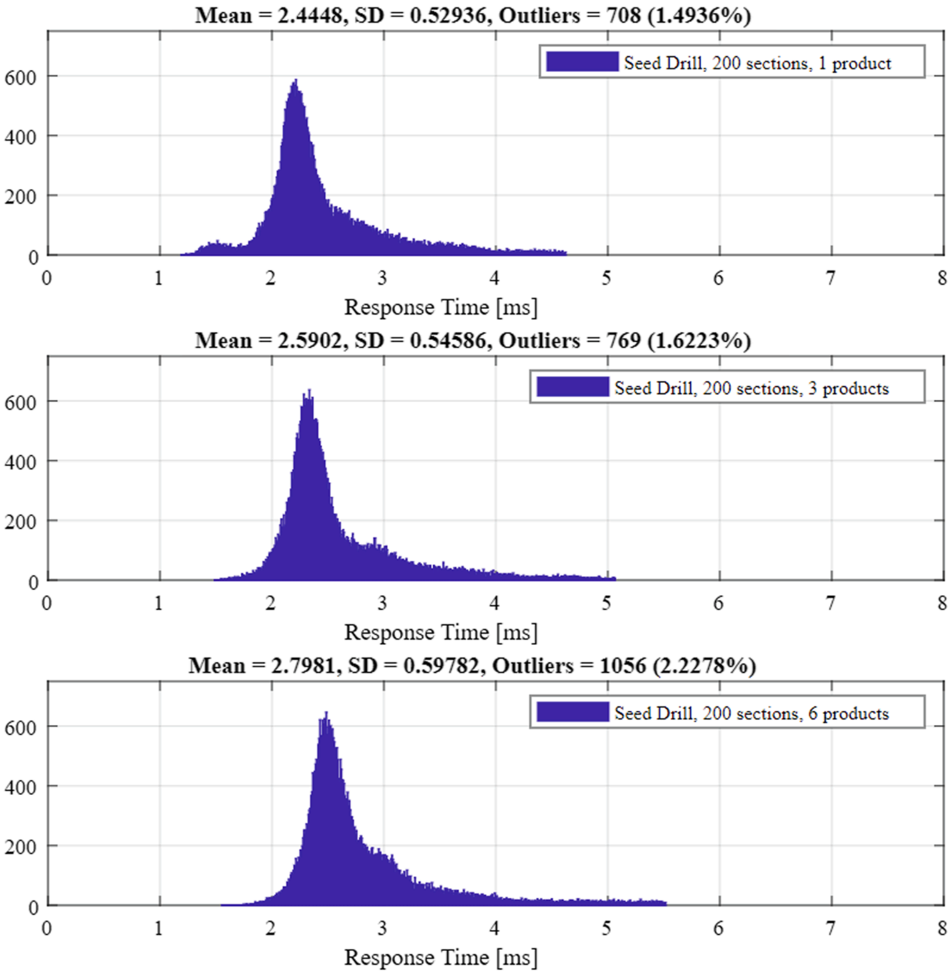

Fig. 25. Seed drill Duration of the OPC UA data exchange.

incorporate all of the required elements together into one network, possibly with OPC UA in control of all aspects, or possibly with multiple protocol stacks on the network simultaneously.

It has been shown that OPC UA over an Ethernet network is in theory able to control many more sections than the current IOS 11783 standard would be able to, and at a much faster rate. It remains to be seen what new implements will be able to accomplish with this, and what the benefits will be to the end users (farmers). Future research opportunities exist in analysing and quantifying the agricultural benefits that could be brought about by improved control and communication networking.

This work was undertaken using the OPC UA client–server model. Further research may investigate if the newer PubSub model is more performant for future agricultural use cases.

Additionally, OPC UA is not the only available middleware, and others such as Data Distribution Service (DDS) should be researched from an information modelling perspective. The next step would be to perform like-for-like comparisons to determine the benefits and drawbacks of each, and may suggestions about which is the most suitable overall.

10. Conclusion

The goal of the work was to develop an information model, for real-time process control of a tractor-implement network, using OPC UA. It was found out that the information model should be based on OPC UA for Devices (DI). It was presented how the current DDOP model of the ISO 11783 Task Controller can be mapped to our novel information model and also extending the modelling rules to allow complex relationships between certain elements. This was done by using container objects for grouping DPD and DPT objects, resource connector objects for modelling complex relationships between boom and bin DET objects, boom device element type for DET objects and new cultural practice identifiers. The previous work by Oksanen et al. (2015) showed that OPC UA was capable of extending the ISO 11783 standard to allow remote condition monitoring. The OPC UA aspects were a separate add-on to the existing ISO 11783 system which did not interact with it. There are several important distinctions which greatly separate that work from this one. Firstly, this work is about addressing the current issues with the ISOBUS TC DDOP by extending the modelling rules as discussed. Secondly it has control elements to enable interaction between the ECUs in the network, thirdly the previous work was a system for converting CAN bus messages and sending them via OPC UA but this work proposes a standalone system without CAN.

OPC UA for the Next Generation ISO 11783 Task Controller information model for data exchange between the TC and implements was designed as a collection of ObjectType and VariableType Nodes. OPC UA TC allows device descriptor objects and their relationships to be modelled as Object and Variable Nodes connected by References. Additionally, it allows the TC to control implements by calling functions presented with Method Nodes.

Modelling rules for using OPC UA TC to model implements were defined. The modelling rules define the allowed relationships between different types of device descriptor objects and different types of DET objects. Required DTC objects for all types of DET objects were defined. Instructions on how RSC objects should be used to connect booms to bins when the application rate of a product is controlled at either boom level or section level were presented.

Tests to evaluate information model were done in an office network, with two office computers connected to each other. The tests covered using different task files and the OPC UA Implement Server was configured to represent different simulated implements. The tests showed that the OPC UA Task Controller was controlling the simulated implement via OPC UA Implement Server correctly. The algorithm that verifies whether or not an implement is compatible with a task was also working as intended. The results of these tests conclude that OPC UA using Ethernet-based networks is a viable alternative to the SAE J1939 protocol and CAN bus in data exchange between the TC and implements.

This article presented benefits of OPC UA information modelling and possible way to standardize the Information model, which follows the same concept as the original DDOP of ISO 11783.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

No data was used for the research described in the article.

Acknowledgments

This work was done as part of research project MaTyKo 2025 (Maataloustyökone 2025 - yhteensopivuus, palvelut ja tulevaisuuden standardit).

References

AEF. 2012b. Agricultural Industry Electronics Foundation. *AEF Guideline Implementation Specification Functionality TC-GEO*. AEF Guideline TC-GEO V1.0, 2012.

AEF. 2012c. Agricultural Industry Electronics Foundation. *AEF Guideline Implementation Specification Functionality TC-SC*. AEF Guideline TC-SC V1.0, 2012.

AEF. 2012a. Agricultural Industry Electronics Foundation. *AEF Guideline Implementation Specification Functionality TC-BAS*. AEF Guideline TC-BAS V1.0, 2012.

AEF. 2012d. Agricultural Industry Electronics Foundation. *AEF Guideline Implementation ISO11783 Exceptions*. AEF Guideline Implementation ISO11783 Exceptions V1.0, 2012.

Beňo, L., Pribiš, R. and Leskovský, R. 2019. *Processing data from OPC UA server by using Edge and Cloud computing*. IFAC-PapersOnLine, Volume 52, Issue 27, 2019, Pages 240-245. IFAC. DOI: 10.1016/j.ifacol.2019.12.645.

Bresenham, J.E., 1965. Algorithm for computer control of a digital plotter. IBM Systems Journal 4 (1), 25–30. https://doi.org/10.1147/sj.41.0025.S.

Cavalieri, S., Chiacchio, F., 2013. Analysis of OPC UA performances. Computer Standards & Interfaces 36, 165–177. https://doi.org/10.1016/j.csi.2013.06.004.

Cavalieri, S., Mulè, S., Salafia, M.G., Scroppo, M.S. 2019. *Towards Integration between OPC UA and OCF*. Proceedings of the 21st International Conference on Enterprise Information Systems (ICEIS 2019), pages 555-562. SciTePress. DOI: 10.5220/0007672205550562.

El Hajj, M., Baghdadi, N., Zribi, M., & Bazzi, H. 2017. *Synergic Use of Sentinel-1 and Sentinel-2 Images for Operational Soil Moisture Mapping at High Spatial Resolution over Agricultural Areas*. Remote Sensing, Volume 9, Issue 12, Page 1292. DOI: 10.3390/rs9121292.

Farid, H.U., Bakhsh, A., Ahmad, N., Ahmad, A., Mahmood-Khan, Z., 2016. Delineating site-specific management zones for precision agriculture. The Journal of Agricultural Science 154 (2), 273–286. https://doi.org/10.1017/S0021859615000143.

Foundation, O.P.C., 2013. OPC Unified Architecture for Devices. OPC UA Companion Specification Release 1, 01.

Foundation, O.P.C., 2017a. OPC Unified Architecture Specification Part 1: Overview and Concepts. OPC UA Specification Release 1, 04.

Foundation, O.P.C., 2017b. OPC Unified Architecture Specification Part 5: Information Model. OPC UA Specification Release 1, 04.

Foundation, O.P.C., 2017c. OPC Unified Architecture Specification Part 3: Address Space Model. OPC UA Specification Release 1, 04.

Foundation, O.P.C., 2017d. OPC Unified Architecture Specification Part 8: Data Access. OPC UA Specification Release 1, 04.

Foundation, O.P.C., 2017e. OPC Unified Architecture Specification Part 10: Programs. OPC UA Specification Release 1, 04.

Foundation, O.P.C., 2017f. OPC Unified Architecture Specification Part 6: Mappings. OPC UA Specification Release 1, 04.

Foundation, O.P.C., 2017g. OPC Unified Architecture Specification Part 4: Services. OPC UA Specification Release 1, 04.

Foundation, O.P.C., 2017i. OPC Unified Architecture Specification Part 9: Alarms & Conditions. OPC UA Specification Release 1, 04.

Foundation, O.P.C., 2017j. OPC Unified Architecture Specification Part 13: Aggregates. OPC UA Specification Release 1, 04.

Foundation, O.P.C., 2018a. OPC Unified Architecture Specification Part 14: PubSub. OPC UA Specification Release 1, 04.

Foundation, O.P.C., 2018c. OPC Unified Architecture Specification Part 11: Historical Access. OPC UA Specification Release 1, 04.

Foundation, O.P.C., 2019. OPC 10001–7 - Amendment 7: Interfaces and AddIns. OPC UA Specification Release 1, 04.

Grisso, R., Alley, M., Thomason, W., Holshouser, D., Roberson, G.T., 2011. Precision farming tools: Variable-rate application. Precision, Geospatial & Sensor Technologies, Publication 442–505.

Haskamp, H., Meyer, M., Mollmann, R., Orth, F., Colombo, A., 2017. Benchmarking of existing OPC UA implementations for Industrie 4.0-compliant digitalization solutions. IEEE. https://doi.org/10.1109/indin.2017.8104838.

ISO. 2011a. *Tractors and machinery for agriculture and forestry - Serial control and communications data network - Part 4: Network layer*. ISO 11783-4:2011(E), International Organization for Standardization, Geneva, Switzerland.

ISO. 2011b. *Tractors and machinery for agriculture and forestry - Serial control and communications data network - Part 11: Mobile data element dictionary*. ISO 11783-11: 2011(E), International Organization for Standardization, Geneva, Switzerland.

ISO. 2015b. *Tractors and machinery for agriculture and forestry - Serial control and communications data network - Part 7: Implement messages application layer*. ISO 11783-7:2015(E), International Organization for Standardization, Geneva, Switzerland.

ISO. 2017. *Tractors and machinery for agriculture and forestry - Serial control and communications data network - Part 1: General standard for mobile data communication*. ISO 11783-1:2017(E), International Organization for Standardization, Geneva, Switzerland.

ISO. 2018b. *Tractors and machinery for agriculture and forestry - Serial control and communications data network - Part 6: Virtual terminal*. ISO 11783-6:2018(E), International Organization for Standardization, Geneva, Switzerland.

ISO. 2019a. *Tractors and machinery for agriculture and forestry - Serial control and communications data network - Part 2 : Physical layer*. ISO 11783-2:2019(E), International Organization for Standardization, Geneva, Switzerland.

ISO. 2015a. *Tractors and machinery for agriculture and forestry - Serial control and communications data network - Part 10: Task controller and management information system data interchange*. ISO 11783-10:2015(E), International Organization for Standardization, Geneva, Switzerland.

ISO. 2018a. *Tractors and machinery for agriculture and forestry - Serial control and communications data network - Part 3: Data link layer*. ISO 11783-3:2018(E), International Organization for Standardization, Geneva, Switzerland.

ISO. 2019b. *Tractors and machinery for agriculture and forestry - Serial control and communications data network - Part 5: Network management*. ISO 11783-5:2019(E), International Organization for Standardization, Geneva, Switzerland.

Khanal, S., Fulton, J., Shearer, S., 2017. An overview of current and potential applications of thermal remote sensing in precision agriculture. Computers and Electronics in Agriculture. Volume 139, 22–32. https://doi.org/10.1016/j.compag.2017.05.001.

Khosla, R., Inman, D., Westfall, D.G., Reich, R.M., Frasier, M., Mzuku, M., Koch, B., Hornung, A., 2008. A synthesis of multi-disciplinary research in precision agriculture: site-specific management zones in the semi-arid western Great Plains of the USA. Precision Agriculture 9 (1), 85–100. https://doi.org/10.1016/j.csi.2016.09.004.

Kim, W. and Sung, M. 2017. *OPC-UA Communication Framework for PLC-based Industrial IoT Applications: Poster Abstract*. 2017 IEEE/ACM Second International Conference on Internet-of-Things Design and Implementation (IoTDI). IEEE. DOI: 10.1145/3054977.3057305.

Kožár, S. and Kadera, P. 2016. *Integration of IEC 61499 with OPC UA*. 2016 IEEE 21st International Conference on Emerging Technologies and Factory Automation (ETFA). IEEE. DOI: 10.1109/ETFA.2016.7733538.

Kuska, M.T., Mahlein, A.-K., 2018. Aiming at decision making in plant disease protection and phenotyping by the use of optical sensors. European Journal of Plant Pathology 152 (4), 987–992. https://doi.org/10.1007/s10658-018-1464-1.

Larson, J.A., Velandia, M.M., Buschermohle, M.J., Westlund, S.M., 2015. Effect of field geometry on profitability of automatic section control for chemical application equipment. Precision Agriculture 17 (1), 18–35.

Lee, B., Kin, D.-K., Yang, H. and Oh, S. 2017. *Model transformation between OPC UA and UML*. Computer Standards & Interfaces, Volume 50, February 2017, Pages 236-250. Elsevier B.V. DOI: 10.1016/j.csi.2016.09.004.

Lehnhoff, S., Rohjans, S., Uslar, M. and Mahnke, W. 2012. *OPC Unified Architecture: A Service-Oriented Architecture for Smart Grids*. 2012 First International Workshop on Software Engineering Challenges for the Smart Grid (SE-SmartGrids). IEEE. DOI: 10.1109/SE4SG.2012.6225723.

Mahnke, W., Leitner, S.-H., Damm, M., 2009. OPC Unified Architecture. Springer.

Morato, A., Vitturi, S., Tramarin, F., Cenedese, A., 2021. Assessment of Different OPC UA Implementations for Industrial IoT-Based Measurement Applications. IEEE Transactions on Instrumentation and Measurement 70, 1–11. https://doi.org/10.1109/tim.2020.3043116.

Mulla, D.J., 2013. Twenty five years of remote sensing in precision agriculture: Key advances and remaining knowledge gaps. Biosystems Engineering 114 (4), 358–371. https://doi.org/10.1016/j.biosystemseng.2012.08.009.

National Research Council, 1997. Precision Agriculture in the 21st Century: Geospatial and Information Technologies in Crop Management. The National Academies Press.

Oksanen, T., Auernhammer, H., 2021. ISOBUS— The Open Hard-Wired Network Standard for Tractor-Implement Communication, 1987–2020. Agricultural Equipment Technology Conference. ASABE, Louisville, Kentucky, USA.

Oksanen, T., Piirainen, P., Seilonen, I., 2015. Remote access of ISO 11783 process data by using OPC Unified Architecture technology. Computers and electronics in agriculture 117, 141–148.

OPC Foundation and AutomationML, 2016. OPC 30040 - UA for AutomationML. Release 1 (00), 00.

OPC Foundation and HKI, 2019. OPC 30200 - Commercial Kitchen Equipment. Release 1, 00.

OPC Foundation and PLCopen, 2016. OPC 30001 - IEC 61131–3 Client Function Blocks for OPC UA. Release 1, 1.

OPC Foundation and VDMA, 2019. OPC 40010–1 - Robotics Part 1: Vertical Integration. Release 1, 00.

Pauker, F., Frühwirth, T., Kittl, B. and Kastner, W. 2016. *A Systematic Approach to OPC UA Information Model Design*. Procedia CIRP, Volume 57, Pages 321-326. Elsevier B. V. DOI: 10.1016/j.procir.2016.11.056.

Piirainen, 2014. P. *OPC UA Based Remote Access to Agricultural Field Machines*. Master's Thesis, Aalto University School of Electrical Engineering, Espoo, Finland.

Piotrowska-Długosz, A., Dlugosz, J., Namysłowska-Wilczyńska, B., Zamorski, R., 2011. Field-scale variability of topsoil dehydrogenase and cellulase activities as affected by variability of some physico-chemical properties. Biology and Fertility of Soils 47, 101–109. https://doi.org/10.1007/s00374-010-0507-3.

Pribiš, R., Beňo, L. and Drahoš, P. 2019. *Implementation of Micro embedded OPC Unified Architecture server-client*. IFAC-PapersOnLine, Volume 52, Issue 27, 2019, Pages 114-120. IFAC. DOI: 10.1016/j.ifacol.2019.12.742.

Profanter, S., Tekat, A., Dorofeev, K., Rickert, M. and Knoll, A. 2019. *OPC UA versus ROS, DDS, and MQTT: Performance Evaluation of Industry 4.0 Protocols*. IEEE International Conference on Industrial Technology (ICIT), 955-962, doi: 10.1109/ICIT.2019.8755050.

Schleipen, M., Gilani, S.-S., Bischoff, T., Pfommer, J. 2016. *OPC UA & Industrie 4.0 - Enabling Technology with High Diversity and Variability*. Procedia CIRP, Volume 57, Pages 315-320. Elsevier B.V. DOI: 10.1016/j.procir.2016.11.055.

Smart, D. and Brill, V. 2022. *AEF – High Speed ISOBUS – Technology Readiness for a Next Generation Network*. 79th International Conference on Agricultural Engineering - Land.Technik AgEng 2022 conference. VDI, Düsseldorf. Pages 551-558.

Smart, D. and Brill, V. 2019. *High Speed ISOBUS, an AEF Project for next generation Ag networking*. 77th International Conference on Agricultural Engineering - Land. Technik AgEng 2019 conference. VDI, Hannover. Pages 91-106.

Stark, B., Smith, B., & Chen, Y. 2014. *Survey of thermal infrared remote sensing for Unmanned Aerial Systems*. 2014 International Conference on Unmanned Aircraft Systems (ICUAS), Pages 1294-1299, DOI: 10.1109/ICUAS.2014.6842387.

Tantik, E. and Anderl, R. 2019. *Concept for Improved Automation of Distributed Systems with a Declarative Control based on OPC UA and REST 2019*. 7th International Conference on Control, Mechatronics and Automation (ICCMA). IEEE. DOI: 10.1109/ICCMA46720.2019.8988777.

Tian, S. and Hu, Y. 2019. *The Role of OPC UA TSN in IT and OT Convergence*. 2019 Chinese Automation Congress (CAC). IEEE. DOI: 10.1109/CAC48633.2019.8996645.

Tummers, J., Kassahun, A., Tekinerdogan, B., 2019. Obstacles and features of Farm Management Information Systems: A systematic literature review. Computers and Electronics in Agriculture 157, 189–204. https://doi.org/10.1016/j.compag.2018.12.044.

Vdma, 2020. ISOBUS Data Dictionary. Available at: Referenced 27 (1), 2020 https://www.isobus.net/isobus/.

Veichtlbauer, A., Ortmayer, M. and Heistracher, T. 2017. *OPC UA integration for field devices*. IEEE 15th International Conference on Industrial Informatics (INDIN). IEEE. DOI: 10.1109/INDIN.2017.8104808.

Wally, B., Huemer, C., Mazak, A. and Wimmer, M. 2018. A *Variability Information Model for OPC UA*. IEEE 23rd International Conference on Emerging Technologies and Factory Automation (ETFA). IEEE. DOI: 10.1109/ETFA.2018.8502502.

Weiss, M., Jacob, F., Duveiller, G., 2020. Remote sensing for agricultural applications: A meta-review. Remote Sensing of Environment 236, 111402. https://doi.org/10.1016/j.rse.2019.111402.

Ye, X. and Hong, H. 2018. *An AutomationML/OPC UA-based Industry 4.0 Solution for a Manufacturing System*. 2018 IEEE 23rd International Conference on Emerging Technologies and Factory Automation (ETFA). IEEE. DOI: 10.1109/ETFA.2018.8502637.

Zhang, N., Wang, M., Wang, N., 2002. Precision agriculture—a worldwide overview. Computers and Electronics in Agriculture 36 (2–3), 113–132. https://doi.org/10.1016/s0168-1699(02)00096-0.

Zunino, C., Malena, D., Cena, G., Scanzio, S., Valenzano, A. 2021. *Black-Box Analysis of the Publish-Subscribe Notification Latency in Real OPC UA Servers*. 26th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA). Pages. 1-4. DOI: 10.1109/etfa45728.2021.9613259.