The IMOCO4.E reference framework for intelligent motion control systems

Sajid Mohamed[1], Gijs van der Veen[1], Hans Kuppens[2], Matias Vierimaa[3], Tassos Kanellos[4], Henry Stoutjesdijk[5], Riccardo Masiero[6], Kalle Määttä[7], Jan Wytze van der Weit[8], Gabriel Ribeiro[9], Ansgar Bergmann[10], Davide Colombo[11], Javier Arenas[12], Alfie Keary[13], Martin Goubej[14], Benjamin Rouxel[15], Pekka Kilpeläinen[3], Roberts Kadiķis[16], Mikel Armendia[17], Petr Blaha[18], Joep Stokkermans[1], Martin Čech[14], Arend-Jan Beltman[2]

[1]ITEC B.V., [2]Sioux Technologies, [3]VTT, [4]GNT, [5]Philips Medical Systems, [6]CRIT, [7]Normet, [8]Philips Consumer Lifestyle, [9]Edilasio, [10]Kion Group, [11]WEG Automation Europe, [12]Fagor AOTEK, [13]Tyndall National Institute, [14]Univ. of West Bohemia, [15]Univ. of Modena, [16]Institute of Electronics & Computer Science, [17]TEKNIKER, [18]Brno Univ. of Technology

***Abstract*—Intelligent motion control is integral to modern cyber-physical systems. However, smart integration of intelligent motion control with commercial and industrial systems requires domain expertise, industrial 'know-how' of the production processes, and resilient adaptation for the various engineering phases. The challenge is amplified with the adoption of advanced digital twin approaches, big data and artificial intelligence in the various industrial domains. This paper proposes the IMOCO4.E reference framework for the smart integration of intelligent motion control with commercial platforms (e.g. from SMEs) and industrial systems. The IMOCO4.E reference framework brings together the architecture, data management, artificial intelligence and digital twin viewpoints from the industrial users of the large-scale 'Intelligent Motion Control under Industry4.E' (IMOCO4.E) consortium. The framework envisions a generic platform for designing, developing, and implementing novice and complex motion-controlled industrial systems. Refinements and instantiations of the framework for the IMOCO4.E industrial cases validate the framework's applicability for various industrial domains throughout the engineering phases and under different constraints imposed on the industrial cases.**

***Index Terms*—reference framework, smart system integration, AI, digital twin, data management, cyber-physical systems, mechatronics, motion control, edge computing**

I. Introduction

Intelligent motion control systems are critical in modern industrial environments that require precise and complex motion control [1] for efficient and reliable operations. These systems integrate advanced technologies such as smart sensors [2], intelligent algorithms [3], novel computing platforms [4] and smart actuators to enable real-time monitoring and control of cyber-physical systems [5], [6]. Smart sensors play a critical role in intelligent motion control systems [7] by providing real-time feedback on the position, velocity, acceleration, and other parameters of the physical system being controlled. Smart actuators, on the other hand, enable precise control of motion by applying forces and torques to the physical system.

Intelligent algorithms [8] are the backbone of intelligent motion control systems as they process the data from smart sensors to decide how to control the system. These algorithms can be designed using various techniques, including artificial intelligence (AI) and machine learning [9], [10], to enable intelligent control of complex systems. They are increasingly used in various industrial applications and domains, including (semiconductor) manufacturing [11], packaging, robotics [12], autonomous vehicles [13], [14], and healthcare. Intelligent motion control systems offer numerous benefits over traditional motion control systems, including improved accuracy, reliability, and efficiency, with reduced maintenance costs and downtime.

However, the challenge is that the smart integration of intelligent motion control systems with commercial and industrial systems [15] is a complex task requiring significant domain expertise, industrial 'know-how' of the industrial processes and resilient adaptation for the various engineering phases. Advancements in digital twin (DT) approaches, big data, and AI amplify this challenge due to its complexity and the need to integrate these novel methods with the existing brownfield systems [16]. As intelligent motion control is a fundamental building block for systems in multiple industrial domains [17], a standardised integration approach that considers digital twins, data management, and AI is lacking. This makes it difficult for start-ups, small and mid-size enterprises (SMEs) and academic partners to develop commercial platforms seamlessly integrating with brownfield industrial systems.

This paper summarises the reference framework developed within the IMOCO4.E project [18] that is a successor of the I-MECH project [19]. The framework envisions a generic platform for designing, developing, and implementing novice and complex motion-controlled industrial systems. Refinements and instantiations of the framework for the IMOCO4.E industrial cases are also provided, and they validate the framework's applicability for various industrial domains throughout the engineering phases and under different constraints imposed on the industrial cases. These results were obtained after the project's first year with contributions from many European partners (industrial, SME and academic).

This paper is organised as follows: The IMOCO4.E concepts and terminology is explained in Section II. The methodology for defining the reference framework is explained in Section III. The main part is the description of a reference framework provided in Section IV. Some refinements and instantiations are proposed in Section V. Finally, conclusions and ideas for future work are given in Section VI.

Corresponding authors: S. Mohamed (sajid.mohamed@itecequipment.com, ITEC B.V.), M. Čech (mcech@kky.zcu.cz, Univ. of West Bohemia).

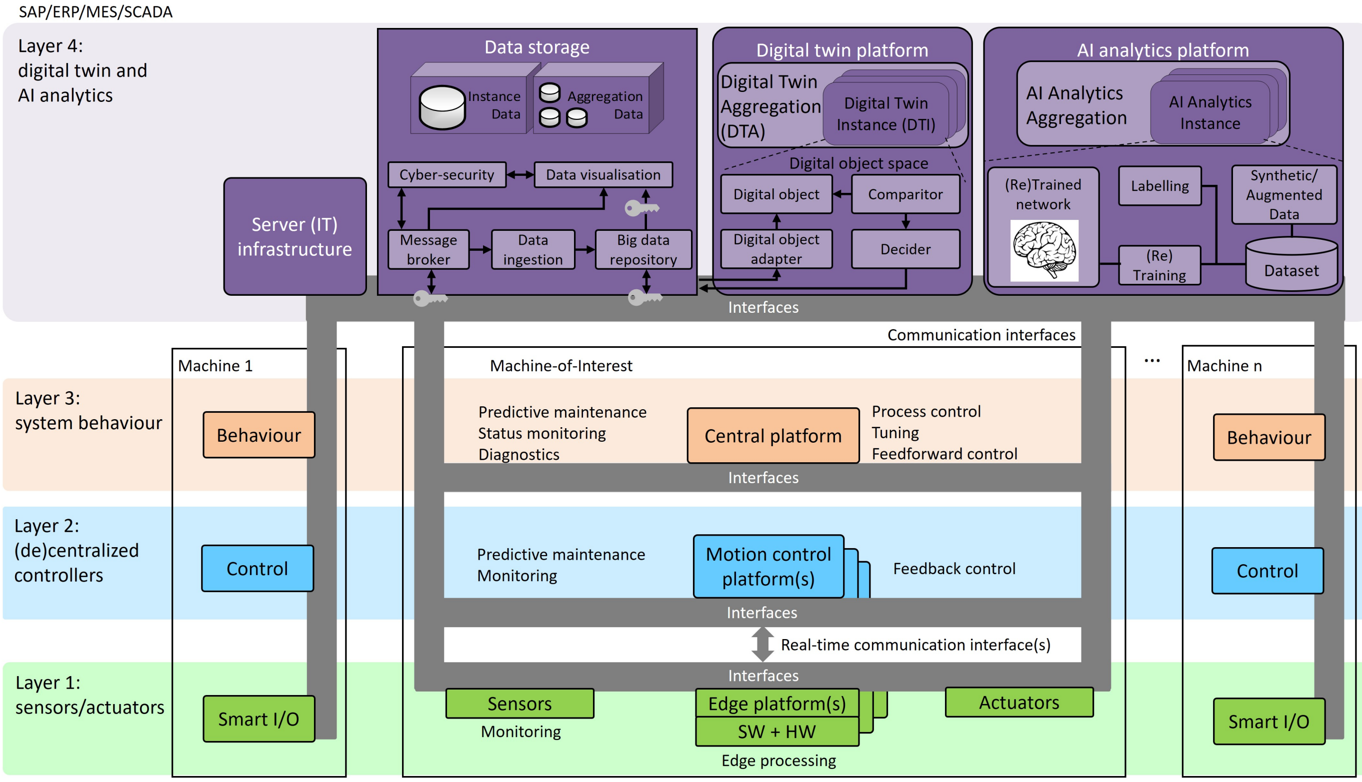


Fig. 1: The IMOCO4.E reference framework's architecture viewpoint

II. IMOCO4.E Concepts and Terminology

- *Brownfield* and *Greenfield* in the context of the IMOCO4.E project - The brownfield architecture characterises the existing legacy systems (software, hardware, or platform) that would be part of the corresponding Pilots, Demonstrators or Use cases during the IMOCO4.E framework development and integration. The greenfield architecture characterises the envisioned systems (software, hardware, or platform) on top of the existing legacy systems that would be part of the corresponding Pilots, Demonstrators or Use cases during the IMOCO4.E framework development and integration.
- *BBs* (Building blocks) – technological system components (hardware, software, sensors, actuators, etc.) which can be reused in various industrial domains. There are 10 BBs defined in the IMOCO4.E project [18].
- *Layers* – Cloud-to-edge vertical view that related reference framework to well-adopted automation pyramid. The four layers are illustrated in Fig. 1.
- *Pilots* – Framework applications defined at the machine level. In the IMOCO4.E project, five Pilots are defined - tissector, semiconductor manufacturing, packaging, healthcare robotics and mining/tunneling.
- *Demonstrators* – Framework applications defined at the industrial production line level. Four Demonstrators are defined - shaver blades, plastic molding, warehouse logistics and cosmetics production.
- *Use Cases* – Partial applications on commercial productions of motion control ecosystem. Four Use cases are defined - industrial drive for smart mechatronics applications, CNC for production, tactile robot teleoperation and collaborative robotic platform.
- *Digital twin (DT)* – Special real-time aspect of connecting virtual machine models to physical devices. In the IMOCO4.E framework, a digital twin comprises five dimensions [20] – the physical entity, virtual model, data, service, and connection or interaction.

III. Methodology

The design process includes identifying the shortcomings of the state-of-the-art reference architectures, identifying the needs for future smart production from the mechatronics and robotic point of view, gathering and characterizing the brownfield and greenfield architectures of the Pilots, Demonstrators, and Use cases from the industrial partners in the IMOCO4.E consortium, defining the requirements and specifications of the different layers in the project on a BB level, and adhering to the guidelines of the IMOCO4.E methodology and toolchains.

The steps leading to the design of the IMOCO4.E reference framework are detailed below.

1) Identifying shortcomings from the state-of-the-art reference architectures [21]. Understanding the current state-of-the-art and identifying the gaps were essential for defining the IMOCO4.E reference framework. The state-of-the-art methods for motion-driven high-tech applications and shortcomings were reported in [22].
2) Identifying the needs for future smart production from the mechatronics and robotic point of view. The reference framework should address these needs through

relevant requirements. The needs summary gathered per BB defined the requirements for (some of) the relevant topics. The needs were reported in [23].
3) Gathering and characterising the brownfield architectures of the Pilots, Demonstrators, and Use cases from the industrial partners in the IMOCO4.E consortium. The constraints and interfacing options with the brownfield architecture in the generic IMOCO4.E reference framework were reported in [24].
4) The first iteration of the detailed requirements and specifications of the different layers in the IMOCO4.E project was reported in [25], [26]. The requirements and specifications were also detailed on a BB level and must be adhered to while designing the final IMOCO4.E reference framework.
5) The guideline of the IMOCO4.E methodology and toolchains reported in [27] must be adhered to.
6) Gathering and characterising the greenfield architectures from industrial partners in the IMOCO4.E consortium.

Gathering the greenfield reference architectures from Pilots, Demonstrators, and Use case owners of the IMOCO4.E consortium is critical to the design process. A request was sent to the industrial partners to gather the envisioned (greenfield) architecture that the Pilots, Demonstrators, and Use case owners will demonstrate in the IMOCO4.E project. The suggested way of gathering the greenfield architecture was provided, which involves starting with the envisioned (greenfield) architecture(s) that will be demonstrated in the project, modifying the brownfield reference architecture, and providing two versions of the diagram – one public and one confidential.

The final design of the IMOCO4.E reference framework is based on the above considerations and the refinements provided by the industrial partners. The paper presents an example of refinement of the reference architecture and the way of working for refining the architecture. The public diagrams of the greenfield reference architectures gathered from the industrial partners are showcased in this paper to illustrate the refinements from the IMOCO4.E consortium.

IV. IMOCO4.E Reference Framework

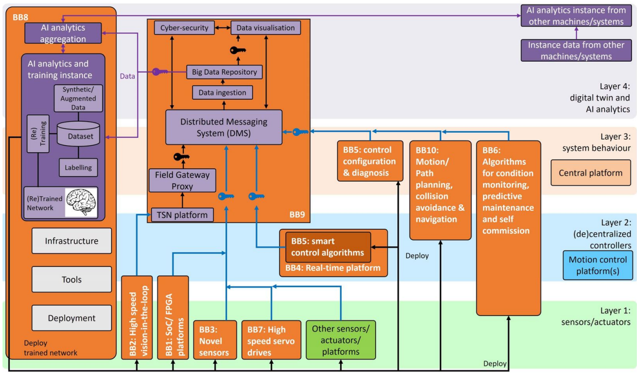
A. Architecture viewpoint

The architecture viewpoint of the IMOCO4.E reference framework is illustrated in 1. The functional breakdown of the components of the reference framework and its overall specifications at an abstract level is detailed below.

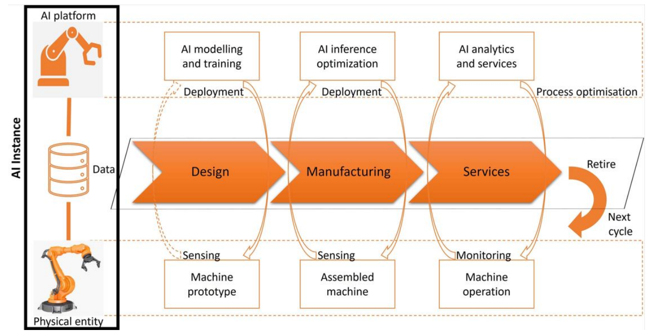
- *Sensors* are the input devices which provide an output with respect to a specific physical quantity. It is a hardware component for detecting or measuring physical properties or parameters by converting signals from one energy domain to the electrical domain. E.g., temperature sensor, proximity sensor, pressure sensor, position sensor, touch sensor, etc.
- *Actuators* are components that tie a control system to its environment. The actuator is a machine component responsible for moving and controlling a mechanism or system, for example, opening a valve.
- *Smart I/O* refers to smart sensors and smart actuators. Smart sensors may include some local (or edge) processing. Smart actuators may include some local intelligence (and processing).
- *Server (IT) infrastructure* is the backbone of a factory or production line for interactions with human users and factory operations, e.g. integrating the machine operation with SAP (systems, applications and products in data processing), ERP (enterprise resource and planning), MES (manufacturing execution system) or SCADA (supervisory control and data acquisition) solutions.
- *Data storage* for the IMOCO4.E reference framework refers to the data necessary for the AI and DT platforms. Instance data refers to the data for a machine instance, and the aggregation data refers to the data for the digital twin aggregation (DTA) and AI analytics aggregation. Section IV-B explains the data management infrastructure for data storage.
- *Interfaces* are the most necessary components in the IMOCO4.E reference framework. Interfaces can be fieldbuses, real-time communication protocols, wireless communication protocols, internet communication and so on. As shown in Fig. 1, interfaces can be present between any two architecture layers on the same machine or between architecture layers on two different machines through Layer 4. The interface between Layer 1 and Layer 2 is typically a real-time communication interface, e.g. ethernet for control automation technology (EtherCAT) or time-sensitive networking (TSN).
- A *platform* refers to the combination of software (tasks, messages, mapping, scheduling) and hardware (computation, communication, memory). The software performance relies heavily on the predictability and reliability of the deployed hardware. The software can overcome errors to a certain degree when a few hardware functions fail. Still, the overall performance will degrade when input signals, i.e., data, are corrupted in hardware before these are in the 'digital' domain. Therefore, the hardware used within the IMOCO4.E project must provide reliable signals and data.

The platform components considered in the IMOCO4.E reference architecture are the following.

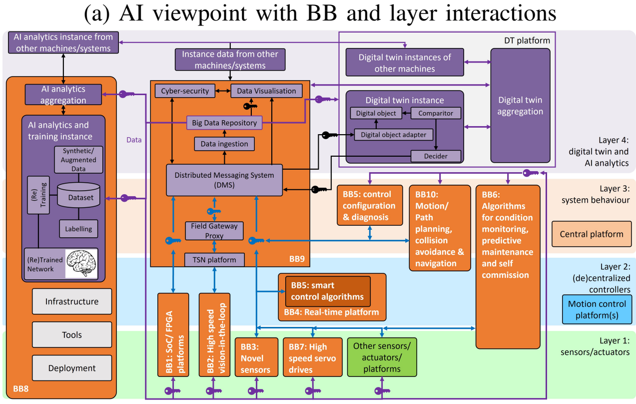
- *Edge platforms (Layer 1)*: When data needs to be processed at the edge, the framework will rely on edge platforms. E.g. for high-speed vision processing, an edge platform is essential since sending image streams over the fieldbus is not ideal due to limited fieldbus bandwidth.
- *Motion control platforms (Layer 2)* are mainly required for accurate and predictable feedback control at a high sampling rate (in the kHz range). Such control can be centralised or decentralised. The state of the components controlled by the platforms can also be monitored, and necessary predictive maintenance actions can be taken.
- A *central platform (Layer 3)*, e.g. a personal computer (PC), is required for coordinating the machine opera-



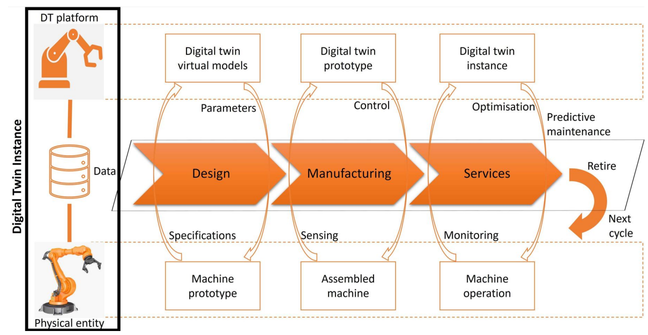
(a) AI viewpoint with BB and layer interactions



(b) AI lifecycle viewpoint



(c) Digital twin viewpoint with BB and layer interactions



(d) Digital twin lifecycle viewpoint

Fig. 2: The IMOCO4.E reference framework's AI and digital twin (DT) viewpoints

tion. Process control, feedforward control, parameter setting/tuning, machine status monitoring, predictive maintenance and diagnostics are some of the tasks/applications that run on the central platform.

- *AI platform (Layer 4)* refers to the AI analytics and training infrastructure. The AI platform consists of AI analytics instances and AI analytics aggregation (explained in Section IV-C).
- *The DT platform (Layer 4)* refers to the software and hardware necessary for the DT instances (DTI) and DT aggregations (DTA) modelling, operation, monitoring, maintenance, and update. DTI and DTA are explained in Section IV-D.

B. Data management viewpoint

The data management viewpoint of the IMOCO4.E reference framework is part of the BB9 solution (cybersecurity and trustworthy data management) [18] and is outlined in the data storage block in Layer 4 of Fig. 1 and the interfaces are detailed in Fig. 2a and Fig. 2c. The BB9 facilitates trustworthy data management and network-based data exchange between IMOCO4.E components by aggregating data from multiple and diverse providers at a unified resource and redistributing the data to the intended recipients both in real-time and asynchronously. The functional components in the data management viewpoint are detailed below.

- *Distributed Messaging System (DMS)* enables the real-time data exchange of heterogeneous, non-binary, semi-structured information among multiple IMOCO4.E components in parallel. The DMS implementation relies on the Apache Kafka framework [28].
- *Big Data Repository (BDR)* ingests, transforms and persistently stores the data flowing through the BB9 DMS. The BDR exposes the stored data to other IMOCO4.E components through an efficient query and retrieval system based on the Elastic Search technology [29].
- *Field Gateway Proxy* collects data from devices at the edge, transforms it to Kafka events and publishes the events to the BB9 DMS.
- *Cyber-Security Module* provides comprehensive protection against unauthorised access to the BB9 DMS through end-to-end authentication mechanisms for devices, microservices and users.
- *Data Visualisation Toolkit* visualizes live data flowing through the DMS and historical data stored in the Big Data Repository.
- *Time-Sensitive Networking (TSN) Platform* guarantees packet transport, in terms of bounded low latency, low packet delay variation and low packet loss in data traffic managed by BB9.

The BB9 solution is optimised for supporting Big Data analytics operations involving AI. It is delivered as a highly scalable system with a customisable microservice architecture composed of re-usable components that can be adapted to meet the needs of brownfield implementations. BB9 offers increased reliability and fault tolerance based on advanced replication features, supports secure interfaces and data access based on comprehensive authorisation and authentication features, and ensures network communication performance levels for time-critical applications.

C. Artificial intelligence (AI) viewpoint

The AI viewpoint with BB interactions of the IMOCO4.E reference framework is illustrated in Fig. 2a. The general principle is that data is collected from Layer 1 (sensors, edge platforms and actuators) and used by the AI framework for modelling, training, optimisation, analytics and/or services. Additionally, it is convenient to have data from Layer 2 (e.g. from BB5 [18] internal signals) and configuration data (e.g. from Layer 3) available in the data collection, so that the dataset is always complete and consistent.

The dataflow from Layer 1, Layer 2 and Layer 3 to the AI framework and back to the corresponding BBs is illustrated in Fig. 2a. BB8 deals with AI-based components and forms the core BB for integrating the AI framework in the IMOCO4.E methodology. BB8 will specify in detail the AI infrastructure, tools and deployment methods in future deliverables. The data necessary for the AI framework is collected, secured and stored based on the methodology developed as part of BB9.

The AI instance refers to the AI framework for a machine instance (or some machine components). AI aggregation refers to the aggregation of all AI instances. The functionality of an AI instance and AI aggregation varies based on the stage in the machine lifecycle and is illustrated in Fig. 2b. During the machine lifecycle's design phase, an AI instance's main functionality is modelling and training. The AI model that is suited for the design objective and satisfies the requirements is identified. Typically, the machine prototype data is used to train the AI model. The AI modelling and training could also start with machine simulation (before the machine prototype is available). Then, the trained AI model is deployed in the machine prototype for testing and validation. The AI instance is optimised for inference performance using the assembled machine data and characteristics during the manufacturing phase. The optimised inference AI model is then deployed in the assembled machine for testing and validation. Finally, during the services phase of the machine lifecycle, the AI instance is used for data analytics and offering other services, e.g. process optimisation. The data monitored by the machine in operation is the input for the AI analytics algorithm, and the AI instance offers optimal services. The AI platform coordinates the AI instance. If required, the AI platform can be independent (with its own hardware and software).

D. Digital twin (DT) viewpoint

The DT viewpoint with BB interactions is illustrated in Fig. 2c. The general principle here is that the physical entity comprises the machine (the sensors, platforms, actuators and interfaces represented through the various BBs and other components, e.g. COTS). The DT platform represents the virtual entity of the DT. The DT virtual models are part of the DT platform. The services and analytics are performed through the AI framework (BB8). The BB9 handles the data collection, storage and cyber-security. The DT platform uses the data from the physical twin, services, and models. Finally, the DT platform sends the parameter changes for optimal machine performance to the relevant physical components or provides warnings or predictive maintenance schedules to the human users, e.g. operators and service engineers.

The DT viewpoint during a machine's lifecycle is illustrated in Fig. 2d. A DT prototype is a virtual description of a prototype machine containing all the information to create the physical twin prototype. The DT prototype can vary from component level to system level. A DTI refers to a specific instance of a physical machine that remains linked to the specific machine throughout its lifecycle. A DTA combines all the digital twin instances.

A DT is helpful throughout the machine's lifecycle. During the design phase, virtual design models form the basis of the machine prototype development. Machine prototype specifications are required by the virtual models for designing an efficient system through iterative optimisation and virtual verification. A DT can be used during the design phase - to design and test a new algorithm and explore use cases before deploying it to the actual physical system. The physical system may not yet be available at this point. A DT also expedites the test time (and hence faster time-to-market) since the physical system has limited test capacity. Testing on the physical system can be expensive if hardware fails due to testing. The DT prototype is used for real-time sensing, control, and process optimisation during the manufacturing phase. A DTI during the service phase enables predictive maintenance, fault detection and diagnosis, state monitoring, process optimisation and so on.

V. Refinements and instantiations

In this section, we will refine the IMOCO4.E reference framework for some of the industrial cases in the IMOCO4.E project. The refinements also detail how to use the reference framework during the various engineering phases and by various personnel (developer, operator, service engineer, etc.). Due to space constraints, we limit the refinements and instantiations to 3 Pilots, 2 Demonstrators and one Use case. The instantiations for all 13 industrial cases will be reported in future work.

A. Refinement and instantiation for tissector (Pilot 1)

Fig. 3 outlines the IMOCO4.E reference framework refinement for Pilot 1, tissector. Layer 1 of tissector abstracts I/O, (quadrature) encoders, physical hardware, motors, and cameras. The I/O, motors (smart actuators), and encoders (smart sensors) are interfaced with the embedded real-time motion controller (Layer 2) through a field bus. The standard EtherCAT interface will be used during the development phase, and a customised hardware interface will be used during the operational phase. The cameras (smart sensors) are interfaced with the application processor (Layer 3). Layer 2 for the tissector has the embedded real-time motion controller platform that controls multiple (servo/stepper) axes, provides

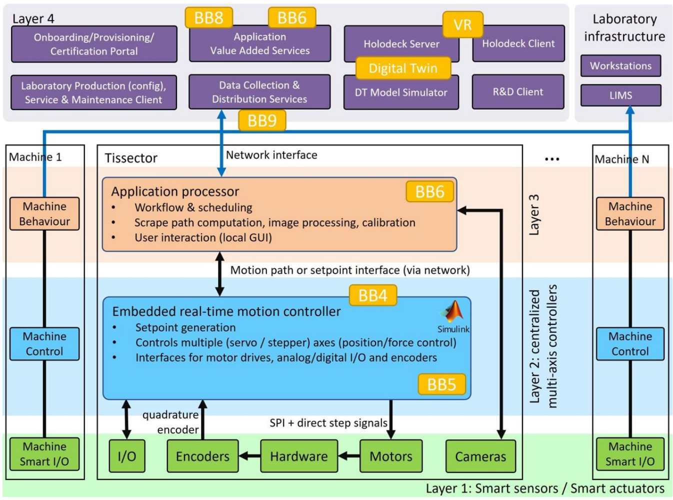


Fig. 3: Pilot 1 tissector architecture refinement from Sioux. The envisioned BBs that will be integrated are annotated.

interfaces for smart sensors and smart actuators in Layer 1 and generates setpoints based on the input from the application processor (Layer 3). The algorithms in Layer 2 are also developed using Simulink software. Layer 2 interfaces with Layer 3 through a motion path or setpoint interface (via network). Layer 3 performs the workflow and scheduling, scrape path computation, image processing, calibration and provides a local GUI for user interaction.

Layer 4 comprises data collection and distribution services, DT model simulator, research and development (R&D) client, laboratory production (config), service and maintenance client, holodeck server, holodeck client, application value-added services, certification portal and the laboratory infrastructure. The holodeck server and client enable virtual reality (VR) integration for the tissector. The DT model simulator enables the digital twin.

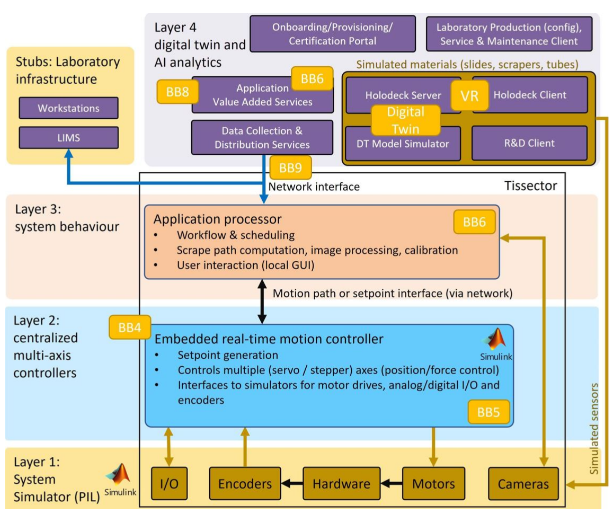


Fig. 4: Pilot 1 tissector simulation infrastructure instantiation based on the refined reference framework.

Fig. 4 instantiates the Pilot 1 reference framework for defining the simulation infrastructure during the development phase. The simulated components are highlighted in the figure.

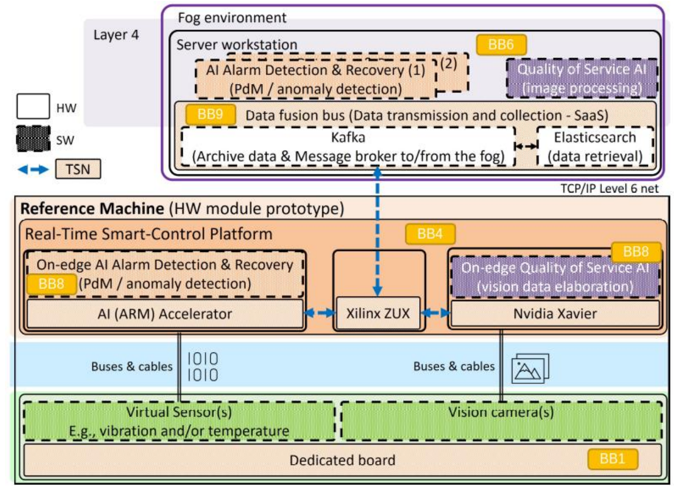


Fig. 5: Pilot 3 architecture and example of refinement in the absence of a machine prototype.

The materials (slides, scraper, tubes) can also be simulated using a DT and VR. Stubs are also provided for the laboratory infrastructure. In addition, the Layer 1 sensors are simulated, and Layer 1 behaves as a system simulator.

B. Refinement for Pilot 3 - High-speed packaging

Fig. 5 outlines the IMOCO4.E reference framework refinement for Pilot 3. This architecture refinement is to face the absence of a machine prototype and real-world sensors for Pilot 3. The main objective of this Pilot is to assess the feasibility of improving automation for quality checks and alarm detection throughout the high-speed packaging process. The AI-based algorithms, in combination with the smart control platform, will help to ensure good quality output and prompt reaction to possible alarms.

To cope with the absence of a machine prototype for Pilot 3, the open data set(s) available online will be selected and used. Some of these data will be used to train suitable AI algorithms, and the remaining part of the data set will be streamed in real time to the real-time smart-control platform. In detail, this approach will cope with the fact that in Pilot 3, sensors are also unavailable; therefore, simulated sensors are enabled to verify and validate the BBs. In this perspective, the real-world sensors will be replaced by simulated instances of real-time data generated via dedicated boards and transmitted to the Real-Time Smart-Control Platform of Layer 3.

C. Refinement for Pilot 4 - Healthcare robotics

Fig. 6 outlines the Pilot 4 architecture refinement along the continuous integration/continuous development (CI/CD) infrastructure. A clear distinction and interface between the development phase and the production phase components are illustrated. The motors and encoders are the actuators and sensors (Layer 1). For the development phase, additional (smart) obstacle avoidance sensors are envisioned and interface to the application layer (Layer 3). The servo drives and the I/O module span across Layers 1 and 2. For CI/CD, the developers use a motion control platform that spans Layers

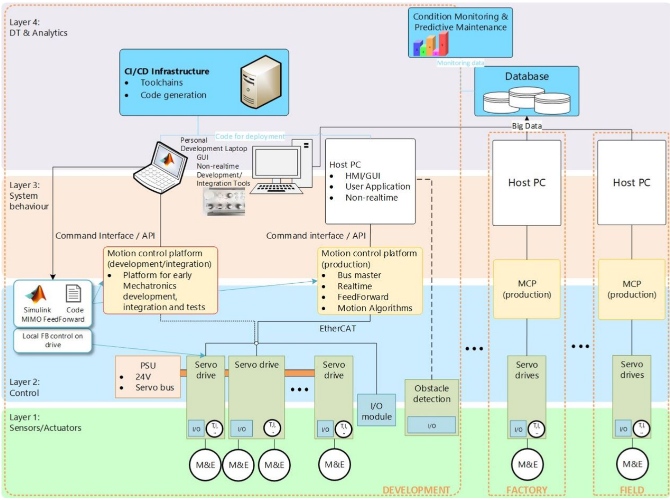


Fig. 6: Pilot 4 refinement with CI/CD infrastructure.

2 and 3 for mechatronics development, integration and tests. The mechatronics development uses Simulink software and custom codes for multi-input multi-output (MIMO) feedforward and feedback control. A personal development laptop with Simulink software and the required tools is used for development/integration. The laptop can access the CI/CD infrastructure (Layer 4), and the code for deployment to the production machine can be uploaded/generated. DTs, combined with a Virtual (Reality) Test environment, are part of the infrastructure (Layer 4) and enable fast and safe development based on these models.

The motion control platform for the production machine executes the feedforward and motion algorithms in real time. The motion control platform is also the bus master for the EtherCAT that interfaces with the servo drives. Local feedback control is executed directly on the drives. The operator of the machine interacts with the host PC, and the code for deployment is updated using the CI/CD infrastructure. During each phase in the machine lifecycle (development, factory and field), the 'big data' from the machine is fed to the cloud (Layer 4) database. Monitoring data from the machines and training DT models enable condition monitoring and predictive maintenance services.

D. Refinement for Demonstrator 2 - Plastic molding

Fig. 7 outlines the refinement for Demonstrator 2 for a plastic molding production line. Nowadays, equipping tools with AI to allow continuous monitoring is a key component in industrial production. Demonstrator 2 targets to over mold wireless sensors (temperature, pressure) and radio-frequency identification (RFID) tags on plastic parts. A controller device could read the data transmitted by the sensors in real-time. The Demonstrator intends to transpose the logic of Industry 4.0 into the final product to introduce new functionalities in tools for plastic injection and create an innovative product with more added value and high incorporation of R&D.

E. Refinement for Demonstrator 3 - Warehouse logistics

Demonstrator 3 includes all the elements required for the transformation from a classic automated system to a modern

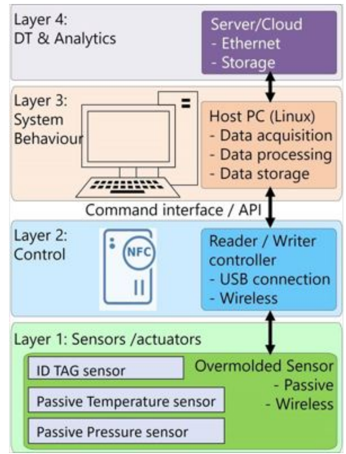


Fig. 7: Demonstrator 2 architecture refinement

autonomous system for internal transport tasks. This means that known standardised brownfield solutions from automation must be extended with new (greenfield) capabilities, so-called senses, for the necessary perception of the environment. This leads to an extension of sensor technology and the addition of novel intelligent solution strategies.

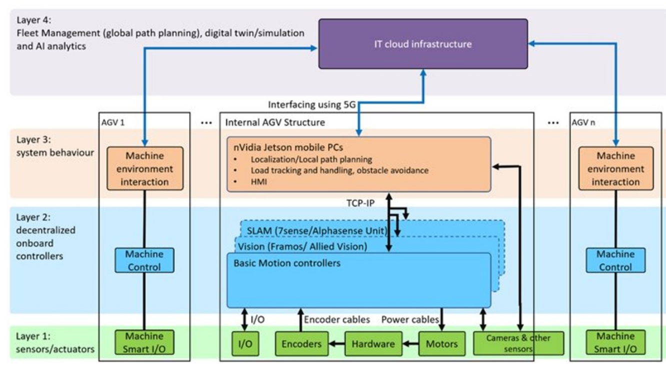


Fig. 8: Demonstrator 3 architecture refinement.

Fig. 8 illustrates framework refinement for Demonstrator 3, where machines are replaced by autonomous ground vehicles (AGVs). In Layer 1, in addition to the existing vehicle control sensors, camera and radar sensors for intelligent detection of semantics like pallets, vehicles, persons and labels were added. Layer 1 is tasked with the perception of objects (persons, vehicles or load), pose estimation of the load carrier, short-term navigation and collision avoidance, and pose (e.g. pallet) load handling.

Using industry-proven training methods (acquisition, simulation and labelling), the functional output of the green-field vision modules of Layer 2 is validated, and functional interaction with the behaviour-relevant brownfield modules of Layer 3 can be guaranteed. The basic functions of the modules from Layer 3 are localisation, mapping and the resulting planning for autonomous load handling. Industry-relevant modules such as global and local planning rely on industry standards such as robot operating system (ROS). In Demonstrator 3, the vehicle's behaviour (control) is composed of the pre-processing by Layer 2 and the final implementation of the vehicle instructions in Layer 3. Based on the information flow from the vehicles combined with external conditions, clear instructions to the subordinate vehicles come from Layer 4. Several vehicle systems are combined to form a fleet and are controlled in Layer 4.

F. Use case 4 - Collaborative robotic platform

The collaborative robotic platform provides a 7DoF robotic arm that can be employed in applications requiring fast and flexible adoption of complex motion trajectories in hybrid environments requiring close cooperation of robots and human workers. Potential application domains include nondestructive inspection and testing or material handling. Fig. 9 provides a refinement of the platform's control architecture.

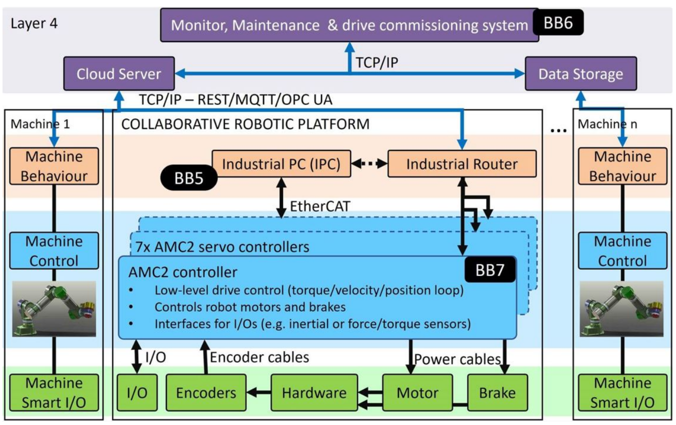


Fig. 9: Use case 4 architecture refinement

Layer 1 contains sensors and actuators embedded in the integrated joints of the robotic arm. Layer 2 deals with decentralised control of the servo drives. Layer 2 is connected to the sensors/actuators layer by corresponding cabling. EtherCAT communication link is established to Layer 3, with an industrial PC (IPC) serving as a centralised motion controller responsible for coordinated motion planning and synchronisation. Layer 4 establishes a monitoring, maintenance and drive commissioning system capable of retrieving and processing relevant machine data.

VI. Conclusions and Future Work

The IMOCO4.E reference framework for intelligent motion control systems is detailed in this paper, along with refinements and instantiations from some of the industrial cases from the IMOCO4.E project. The proposed reference framework is summarised based on the inputs from many European partners of the large-scale IMOCO4.E consortium. A stand-alone designer (e.g. start-ups, SMEs or academia) of an intelligent motion control system can refer to the IMOCO4.E reference framework's architecture, data management, AI and digital twin viewpoints for developing their system adhering to the industry standards and interfaces and for smart integration with existing industrial systems. Future work involves validating the IMOCO4.E reference framework instantiations by demonstrating the smart integration of tangible building block solutions with the defined industrial cases.

Acknowledgment

This work was supported by ECSEL JU in the H2020 project IMOCO4.E, grant agreement No.101007311.

References

[1] A. Sabanovic and K. Ohnishi, *Motion control systems*. John Wiley & Sons, 2011.
[2] A. Schütze, N. Helwig, and T. Schneider, "Sensors 4.0–smart sensors and measurement technology enable industry 4.0," *Journal of Sensors and Sensor systems*, 2018.
[3] S. J. Russell, *Artificial intelligence a modern approach*. Pearson Education, Inc., 2010.
[4] C. Jugade, D. Hartgers *et al.*, "An evaluation framework for vision-in-the-loop motion control systems," in *ETFA*, 2022.
[5] N. Jazdi, "Cyber physical systems in the context of industry 4.0," in *AQTR*, 2014.
[6] S. Mohamed, *Multiprocessor Image-Based Control: Model-Driven Optimisation*. Eindhoven University of Technology, 2022.
[7] C. Jugade, D. Hartgers *et al.*, "Improved positioning precision using a multi-rate multi-sensor in industrial motion control systems," in *European Control Conference*, 2023.
[8] K. Kubíček, M. Čech, and J. Škach, "Continuous enhancement in model-based software development and recent trends," in *ETFA*, 2019.
[9] S. De, Y. Huang *et al.*, "Hardware- and situation-aware sensing for robust closed-loop control systems," in *DATE*, 2021.
[10] V. Jain, S. Mohamed *et al.*, "Vision-based multi-size object positioning," in *DSD*, 2023.
[11] G. van der Veen, J. Stokkermans *et al.*, "How learning control supports industry 4.0 in semiconductor manufacturing," in *ASPE Spring Topical Meeting on Design and Control of Precision Mechatronic Systems*, 2020.
[12] S. Flores and J. Jost, "Person re-identification on a mobile robot using a depth camera," in *ISIE*, 2022.
[13] S. Mohamed, D. Goswami *et al.*, "Optimising multiprocessor image-based control through pipelining and parallelism," *IEEE Access*, 2021.
[14] S. Mohamed, N. Saraf *et al.*, "Adaptive predictive control for pipelined multiprocessor image-based control systems considering workload variations," in *CDC*, 2020.
[15] D. D. Manickam, S. Mohamed *et al.*, "A structured inference optimization approach for vision-based DNN deployment on legacy systems," in *ETFA*, 2023.
[16] M. Čech, J. Königsmarková *et al.*, "Novel tools for model-based control system design based on FMI/FMU standard with application in energetics," in *International Conference on Process Control*, 2017.
[17] D. Bazylev, "Approaches for stabilizing of biped robots in a standing position on movable support," *Scientific and Technical Journal of Information Technologies, Mechanics and Optics*, 2015.
[18] M. Čech, A.-J. Beltman, and K. Ozols, "Digital Twins and AI in Smart Motion Control Applications," in *ETFA*, 2022.
[19] M. Čech, A.-J. Beltman, and K. Ozols, "Pushing mechatronic applications to the limits via smart motion control," *Applied Sciences*, 2021.
[20] Q. Qi, F. Tao *et al.*, "Enabling technologies and tools for digital twin," *Journal of Manufacturing Systems*, 2021.
[21] E. Y. Nakagawa, P. O. Antonino *et al.*, "Industry 4.0 reference architectures: State of the art and future trends," *Computers & Industrial Engineering*, 2021.
[22] H. Kuppens, "D2.1 State-of-the-art methods in Digital Twinning for motion-driven high-tech applications," *Zenodo*, 2023.
[23] M. Vierimaa, "Needs for future smart production in Europe from the mechatronics and robotic point of view," *Zenodo*, 2022.
[24] S. Mohamed, "D2.3 Overall requirements on IMOCO4.E reference framework," *Zenodo*, 2022.
[25] D. Navarro-Cabrera, N. R. Luque Sola, and E. Ros Vidal, "Requirements for advanced motion control (first iteration)," *Zenodo*, 2022.
[26] R. Tomar, "D5.1 Integral (system level) requirements for valuable twinning methods (first iteration)," *Zenodo*, 2022.
[27] H. Kuppens, "D6.1 Guideline of IMOCO4.E Methodology and Toolchains," *Zenodo*, 2022.
[28] J. Kreps, N. Narkhede *et al.*, "Kafka: A distributed messaging system for log processing," in *Proceedings of the NetDB*, 2011.
[29] C. Gormley and Z. Tong, *Elasticsearch: the definitive guide: a distributed real-time search and analytics engine*. O'Reilly Media, 2015.