Multi-drive control and condition monitoring in networked electric drives with EtherCAT

G. Zanuso, *Student Member, IEEE*, V. Fodor, *Member, IEEE*, L. Peretti, and O. Wallmark, *Senior Member, IEEE*

Abstract—Multi-drive systems and condition monitoring could benefit from a larger exchange of information between the electric drives and the industrial network. This is favored by the Industry 4.0 and the Industrial Internet initiatives together with the emergence of high-speed industrial networks. This paper investigates how recent industrial communication protocols can handle the emerging diversity of service requirements. After a brief overview of multi-drive system and condition monitoring applications for networked electric drives, the paper provides a quantitative evaluation of the performance of EtherCAT-based network of electric drives, supporting both dynamic control and condition monitoring.

Index Terms—Condition monitoring, drives, EtherCAT, industrial networks, electric machines, multi-drive systems.

I. INTRODUCTION

Electric drives are widespread in industrial plants because of their efficient electromechanical conversion. Electric drives consist on an electric motor supplied by a frequency converter, where the latter includes the processing unit of the system and the capability to connect to an industrial network. In current industrial plants, the drive uses the network for limited purposes. From a higher hierarchical layer, often implemented by a Programmable Logic Controller (PLC), the electric drives receive the command references such as position, speed and torque, which the processing unit will use to control the electric motor. On the other direction, the electric drive via the network will respond with alarm conditions when faults occur [1].

The Industry 4.0 and the Industrial Internet initiatives aim to an increased connectivity between the devices in an industrial plant [2], [3], to allow new services and applications that in turn could increase the safety and the efficiency of industrial processes.

The emergence of high-speed industrial networking solutions allow faster communication between the network nodes [4]. However, these networks and network protocols are primarily designed to serve periodic, synchronous traffic with strict delay requirements. Then the question remains, whether the same technologies could serve the variety of traffic types with different service requirements, as it is required by the Industry 4.0 vision.

This work has been supported in part by the Swedish Governmental Agency for Innovation Systems (Vinnova) and the Swedish Electromobility Center (SEC).

The authors are with the School of Electrical Engineering and Computer Science, KTH Royal Institute of Technology, 10044 Stockholm, Sweden (e-mail: zanuso@kth.se; vfodor@kth.se; lucap@kth.se; owa@kth.se).

Considering the electric drives, emerging condition monitoring (CM) techniques and multi-drive systems could benefit from this increased connectivity. Several CM methods for both the electric motor components (e.g. stator windings, rotor bars, bearings) and the converter parts (e.g. DC-bus capacitors, power switches) have been proposed lately [5]. In order to perform CM algorithms, the electric drive could process the sensed information and make decisions locally, or send it to computational units that implement data-analytics-based maintenance. In the latter case, the industrial network needs to cope with CM-related data traffic.

In multi-drive systems, for example, for conveyor belts or for tandem-connected motors, different frequency converters share the same DC bus while the operation of the electric motors aims for a common goal. In these industrial plants, the total losses can be decreased with a cooperative dynamic load sharing strategy between the electric drives [6]. The cooperative strategy is inevitably based on an increased exchange of information within the multi-drive system and thus further traffic in the industrial network.

An initial study about how CM and multi-drive systems benefit from an increased connectivity of electric drives has been presented in [6]. The objective of this paper is to provide a quantitative evaluation of how these two applications, with very different service requirements could be served by the same network. Specifically, the communication requirements of CM and of multi-drive control are derived. An EtherCAT-based connectivity is considered in order to provide a solution for serving the two applications over the same networking infrastructure, and evaluate the effect of the network size and hardware limitations. The obtained results show that while the requirements of multi-drive control limit the number of nodes, the specific features of the EtherCAT protocol allow the transmission of monitoring data with high transmission rate.

This paper is organized in the following parts. In Section II, the background regarding electric drives, multi-drive systems and industrial networks is given. The requirements of electric drives and their communication networks are discussed in Section III for some specific applications. In Section IV a case study for a multi-drive control using an EtherCAT-based communication network is presented. Section V reports final remarks about the work.

II. BACKGROUND

A. Electric drive and multi-drive systems

The schematic of a typical electric drive is shown in Figure 1. The load is mechanically connected to the electric motor, which in turn is fed by the frequency converter through pulse-width-modulated (PWM) voltages. The processing unit is the only intelligence available in the whole drive system and it has the main goal of commanding the power switches based on input signals as the DC-bus voltage, the stator currents and the rotor speed or position. Other than being used for controlling the motor operation, these signals are also employed for performing CM algorithms [7]. The most common CM techniques are based on motor current signature analysis (MCSA), where anomalies in the frequency spectrum of the measured stator currents are related to specific faults [8].

Fig. 1. Schematic overview of an electric drive.

Multiple electric drives can share the same DC bus into a multi-drive system, as shown in Figure 2. Each frequency converter FC_i is equivalent to the one displayed in Figure 1 without the presence of the AC/DC stage. Although not shown in Figure 2, each frequency converter FC_i acquires the motor measurement signals as depicted in Figure 1.

The common DC bus gives the advantages of a reduced number of AC/DC stages (only one in Figure 2) and the possibility to minimize the power flow from the AC grid, because the drives in motoring mode and those concurrently in generating mode can share the power on the DC side [9]. The energy excess present in the DC bus can be redirected to the AC grid, when a regenerative AC/DC converter is employed, or be dissipated by balancing resistors.

B. Status of Industrial Communication Technologies

Industrial communication technologies in the last decades have evolved parallel to internetworking and cellular network technologies [2], [4]. The reason of this parallel development is the special needs of industrial communication, with the main objective of timely and reliable communication between sensors, controllers and actuators. At the same time, industrial communication does not necessarily need to provide scalability which is a basic requirement for the Internet and for cellular networks. Moreover, the network load in industrial networks is usually predictable, consisting mainly of periodic control messages.

Fig. 2. General diagram of a multi-drive system.

The literature classifies wired industrial networks into two large categories. The most traditional category is represented by the so-called fieldbus systems, including for example CAN and Profibus, with specific physical layer technologies that can directly allow synchronous, and preemptive priority based access, and ensure strict delay requirements due proper network dimensioning [10]. The achievable transmission rates of these systems is however low, in the range of 10 Mbps.

The second category of industrial communication protocols is based on the Ethernet technology, and can achieve transmission rates of 100 Mbits. However, the traditional Ethernet networks provide best effort service, and this calls for modifications to ensure strict delivery times and synchronous access. Solutions include scheduling over the transport layer, prioritization over the medium access control layer, and finally, changed medium access control solutions with strict scheduling and clock synchronization. These solutions are however not compatible with the classical Ethernet standards, though still utilize the Ethernet frame format. This last group of solutions include the popular Profinet IRT and EtherCAT.

Finally, wireless networks emerge as the next category of industrial communication, with the promise of more flexible plants and connectivity for moving parts for mobile machinery. A first generation of these networks, for example WirelessHART or Zigbee [11], has relatively low transmission rates, and therefore can not support industrial applications with very low delay requirements.

Emerging technologies based on 5G ultra reliable low latency services and millimeter-wave communication may provide the higher transmission rates. Still, the unreliable wireless medium poses significant new challenges, considering both the reliability and the delay requirements of industrial processes, as well as the more general requirements of security and low downtimes [12], [13].

All in all, today's wired technologies can provide communication with the delay requirements of industrial applications. As the primary objective of these industrial networks is to transmit periodic time-sensitive data, an emerging question is how to include other traffic types, for example the aperiodic monitoring traffic, thus allowing that a single infrastructure serves a multitude of applications. Solutions are proposed for

example for CAN in [14], [15] or for EtherCAT in [16]. This paper contributes to this line of works.

III. APPLICATIONS AND ANALYSIS

A. Dynamic control of multi-drive system

As mentioned in Section I and detailed in [6], multi-drives systems can benefit from a collaborative network of electric drives. The collaboration implies that reference signals for each drives are generated by taking the state of all the other drives into account. In a centralized implementation, it means that a centralized processing unit takes completely care of the reference generation, by taking all the state information data available coming from each of the electric drives into account, while all the local processing units still perform low-level tasks such as communication with drive sensors and PWM.

A distributed implementation does not need any central processing unit. Instead, the drives share their state information with each other, and each local processing unit performs the same optimization task to generate its own reference signals.

The signals exchanged between the central processing unit and each single electric drive in the network have the characteristics displayed in Table I. In the centralized

TABLE I
SIGNALS CHARACTERISTICS FOR MULTI-DRIVE CONTROL.

Parameter	Symbol	Value
Current signals amount	N_c	2
DC-bus voltage signals amount	N_v	1
Speed/position signals amount	N_s	1
PWM reference signals amount	N_{ref}	3
Signals resolution	b	12 bit
Sampling frequency	f_s	10 kHz

implementation, at each PWM period, equal to the sampling period $T_s = 1/f_s$, the measurements of currents, DC-bus voltage and speed (or position) are transmitted to the central processing unit, which calculates the references to be sent back to the local PWM modulator of each electric drive. The data rate of the exchange between the central processing unit and one single local drive is:

$$R_{MD,c} = (N_c + N_v + N_s + N_{ref}) b f_s = 840 \text{ kbps.} \quad (1)$$

The total network transmission rate has to consider the data exchange with each of the electric drives in the system, with the small modification that N_v needs to be transmitted only once, because the DC-bus voltage value is the same for all drives.

The data transmission rate of the distributed implementation depends on the broadcast/multicast capability of the applied network technology. The minimum required rate can be obtained by assuming that each drive needs to transmit its data only once, and this is received by all the other drives. Since in this case the DC-bus voltage value does not need to be exchanged, and the reference signals do not need

to be distributed, the resulting minimum transmission rate requirement becomes

$$R_{MD,d} = (N_c + N_s) b f_s = 360 \text{ kbps.} \quad (2)$$

The results obtained in (1) and (2) should be considered only for their order of magnitude, because overhead bits required in the transmission, as well as possible additional constraints of the communication protocol are not taken into account.

All in all, the transmission rate requirement of the multi-drive system is high compared to the requirement of traditional drive control. The wired Ethernet-based industrial communication protocols described in Section II-B can support this application.

B. Condition monitoring

The communication network requirements for a CM application are summarized in this Section. Among the different CM methods, the MCSA is hereby considered. Further information can be found in [6].

As mentioned in Section II-A, MCSA requires the frequency analysis of the stator current measurements in order to extract the fault information. The use of Fast Fourier Transforms (FFTs) is the usual technique to perform frequency analysis [17]. The effectiveness of these methods strongly depends on the resulting frequency spectrum resolution Δf , which is the inverse of the observation time window T_{obs} , i.e. the time length in which the signal to be analyzed is acquired.

As an example, a CM method is applied to 3 currents and one speed (or position), and each of this signals is assumed to be digitally converted in 12 bit. Moreover, in a single acquisition session, a consecutive number of measurements N_{meas} of currents and speed are assumed to be acquired for a greater accuracy. The relevant data about the acquisition of signals are summarized in Table II. The data size originated

TABLE II
SIGNALS CHARACTERISTICS FOR CM.

Parameter	Symbol	Value
Current signals amount	N_c	3
Speed/position signals amount	N_s	1
Signals resolution	b	12 bit
Sampling frequency	f_s	10 kHz
Observation time window	T_{obs}	5 s
Number of measurements	N_{meas}	10

by a single acquisition session is:

$$P_{CM} = \frac{(N_c + N_s) b f_s T_{obs} N_{meas}}{8} = 3 \text{ MBytes.} \quad (3)$$

The acquisition sessions are repeated periodically with a very low frequency (hours or even days) given the slowness of the degrading phenomena in electric drives. Between consecutive acquisition sessions, the obtained amount of data P_{CM} needs to be adequately processed, according to the specific CM technique. Differently than the case reported in Section III-A,

the transmission of the CM data P_{CM} does not have real-time constraints, and could be transmitted through asynchronous access in the industrial network, in the time that is left by the more demanding control applications. Most of the technologies described in Section II-B would support this communication and provide data delivery within minutes.

Table III summarizes the communication requirements for messages generated in multi-drive and CM applications.

TABLE III
SUMMARY OF MESSAGE REQUIREMENTS FOR MULTI-DRIVE CONTROL APPLICATION (MD) AND CONDITION MONITORING (CM).

Case	Data size	Frequency	Type	Max. delay
MD	~ 10 bits	10 kHz	Periodic	$10 \mu s$
CM	~ 1 MBytes	1 per hour	Aperiodic	-

IV. CASE STUDY: ETHERCAT

EtherCAT based communication is a good candidate to demonstrate the feasibility of monitoring and control of multi-drive systems through an industrial network. EtherCAT is one of the most used industrial network technologies, due to its high bitrate, Ethernet compatibility and real-time transmission capabilities [18]. Use of EtherCAT for multi-drive applications has been reported in [19] for large systems, without strict delay requirements, and more recently in [20] and [21] for delay sensitive step motor control.

EtherCAT is based on a daisy-chain bus topology with a master-slave architecture, in which the master periodically transmits an Ethernet frame, and slaves read and write into this frame. Unlike traditional Ethernet, EtherCAT allows on the fly reading and writing of data as the Ethernet frame is forwarded on the bus. As reading and writing at the slaves do not cause significant additional delay, EtherCAT is suitable for applications with low delay requirement. In addition to master-slave communication, EtherCAT also provides limited multicast communication and slave to slave communication in the direction from the master towards the slaves on the bus, and extensions have been proposed to allow the transmission of aperiodic messages [16]. These capabilities are very well suited for implementing the control and monitoring of multi-drive systems.

A. EtherCAT protocol description

As discussed in Section III-A, multi-drive control can be implemented with centralized decision making, where, reasonably, the central processing unit would be connected to the EtherCAT master node. Since the drives themselves have computational power, even a distributed implementation is possible, where each drive, as EtherCAT slave node, receives updates from all other drives, and computes a control decision locally.

Under centralized control a $(N_c + N_v + N_s)b = 48$ bits state information from each of the slaves needs to be transmitted to the master. After decision making at the master

Fig. 3. EtherCAT communication and processing times for the centralized and for the distributed implementation of multi-drive control.

node, the $N_{ref}b = 36$ bits reference signal information is transmitted back to each slave. Figure 3 shows the periods of communication and computing in the EtherCAT network. At the beginning of the sampling period the slaves communicate with their ADCs for measurement acquisition, but this time is very short and therefore it will be disregarded in the evaluation. This is followed by a period of communication, for the collection of state information from the slaves. After that, the central processing unit computes the reference signals, and the signals are transmitted back to the slaves.

The time period of communication in an EtherCAT bus has to include all the time when the bus still forwards information. This period is called as the cycle time, and it denotes the time from the start of the transmission of an Ethernet frame at the master, until the time the last bit of the frame arrives back to the master. Due to the constraints of the EtherCAT protocol, the start of the frame transmissions has to happen periodically. Considering multi-drive control, this can be turned into the requirement that one transmission cycle and the centralized decision needs to be performed within half of the sampling period.

Under distributed control the state information of $(N_c + N_s)b = 36$ bits from all the slaves is read by all other slaves. Direct slave to slave communication is possible in only one direction on the bus, and therefore the master needs to collect and retransmit the state information of each drive. Thus, even in this case two cycles have to be made on the EtherCat bus, followed by the local decision making at the drives. Figure 3 shows the distributed case as well. Again, the timing requirement becomes the same as under centralized control. The difference is that now extensive calculations happen at the drives, after the second transmission cycle in the sampling period.

Models to derive the cycle time can be found in the literature, see e.g. [12], [16], [22]. To concentrate on the main parameters, here a slightly simplified model is presented that does not take the exact length of the cables into account.

The cycle time T_c , that is, the time from the beginning of the transmission of a frame at the master, until the time the frame returns and is completely received by the master can be expressed as follows. Let the network have S slaves, each

Fig. 4. EtherCAT frame format.

connected to a drive. Let t_p be the end to end propagation time on the bus, and t_f the forwarding time, that is the time from the reception of the first bit of the frame slave until the transmission of the same bit. Let t_{tr} be the transmission time of the Ethernet frame at the master node. Due to the on the fly frame processing at the slaves, the complete frame transmission time needs to be considered only once, and the cycle time is

$$T_c = t_p + N t_f + t_{tr}, \quad (4)$$

where the time to transmit the Ethernet frame is $t_{tr} = l/r$, where l is the number of bits in the frame, and $r = 100$ Mbits/s is the transmission rate. The values of t_p and t_f depend on the EtherCAT hardware, where t_p is in the order of tens of nanoseconds, and t_f is usually less than a microsecond [20].

The frame size l is determined by the structure of an Ethernet frame in EtherCAT. The frame includes the standard Ethernet header and trailer, leading to an Ethernet overhead of $l_{o,e} = 26$ Bytes. Inside the Ethernet data field, there is the EtherCAT header of $l_{o,c} = 2$ Bytes. This is followed by the EtherCAT telegrams. Each telegram consists of the 10 Bytes header, followed by the data payload of length d , and finally 2 Bytes trailer, leading to an overhead of $l_{o,t} = 12$ Bytes in each telegram. In addition, there are constraints on the Ethernet payload size, which has to be between $l_{min,c} = 46$ Bytes and $l_{max,c} = 1500$ Bytes. Smaller payloads are extended to 46 Bytes with padding, while larger data is fragmented into several frames that are transmitted back to back. Considering the multi-drive application, the transmission of a frame of maximum size would violate the constraint of the sampling period, since $l_{max,c}/r > T_s$. Therefore, the case of the transmission of multiple frames in a cycle will not be considered.

B. Performance modeling of multi-drive control

If the network serves only the multi-drive control application, as described in Section III-A, the number of telegrams in each frame is S , since either each drive transmits its state information, or each drive needs to receive reference signals. Thus, the size of the Ethernet frame, including the Ethernet

header and trailer, the EtherCAT header and the headers and trailers of each of the telegrams, is

$$l = l_{o,e} + \max(l_{min,c}, l_{o,c} + N(l_{o,t} + d)). \quad (5)$$

Following the analysis in Section III-A, in the frame that collects the state information, the length of the payload of the datagrams is $d = d_{st} = 6$ Bytes for each slave. For the transmission of the reference signals $d_{ref} = 5$ Bytes is needed in each telegram, sent to each of the slaves. In the case of distributed control, the telegrams always contain the state information, which is now $d_{dist} = 5$ Bytes. Substituting (5) to (4), gives the cycle times $T_{c,st}$, $T_{c,ref}$ and $T_{c,dist}$.

All cycle times increase linearly with the forwarding time t_f , as well as with the number of slaves S , and $T_{c,st} > T_{c,ref} = T_{c,dist}$.

In both the centralized and in the distributed case the transmission of the state information and the processing time needed to calculate the reference signals can occupy the maximum of the half of the sampling period T_s . Denoting the processing time by T_p , the constraint for the centralized case the is

$$T_{c,st} + T_p < \frac{T_s}{2}, \quad (6)$$

and for the distributed case it is

$$T_{c,dist} + T_p < \frac{T_s}{2}. \quad (7)$$

These determine the maximum number of drives the network can serve. The value of T_p depends on the computing hardware and it is usually a couple of tens of microseconds.

Figure 5 shows the limit on the number of slaves, that is the number of connected drives, as a function of T_p for different values of t_f . The forwarding time t_f has significant effect on the maximum number of connected drives. The number of connected devices is significantly lower than the one expected based on the transmission rate of the bus, $r = 100$ Mbit/s, and the data rates per drive, given in (1) and (2). This demonstrates that the communication protocol overheads and constraints have to be considered carefully at the system design.

C. Protocol design and performance modeling of CM

The asymmetries of the communication needs, at least in the centralized case, as well as the computing that happens only once in a sample period, result in regular idle times, when neither communication nor computing happens. These periods can be used by other applications, for example to transmit CM data. Specifically, according to Figure 3, in the case of the centralized control, the cycle time of the frame including the transmissions of reference signals does not occupy the available half of the sampling period, and could be extended to transmit other information. In the case of distributed control the frame for collecting the state information could be extended to occupy half of the sampling period. The solution proposed in [16] for the transmission of aperiodic information within EtherCAT frames can utilize

Fig. 5. Maximum number of drives that can be connected over an EtherCAT bus for multi-drive control ($T_s = 100 \mu s$, $t_f = \{0.5, 1, 1.5\} \mu s$).

Fig. 6. Maximum achievable transmission rate of aperiodic traffic for drive monitoring ($T_s = 100 \mu s$, $t_f = \{0.5, 1, 1.5\} \mu s$).

these idle times for the transmission of CM data. It proposes an EtherCAT telegram format where in addition to the usual $l_{o,t} = 12$ Bytes overhead, an aperiodic message header of $l_{o,a} = 12$ Bytes is introduced. This telegram structure allows the transmission of aperiodic messages with different priorities. Since monitoring data will be transmitted rarely, it can be assumed that at each point of time there is a maximum of one slave that transmits aperiodic data. Thus, the capability of prioritized transmission does not need to be utilized, or considered in the analysis.

The transmission rate that is available for the aperiodic traffic, in the case of centralized and in the case of distributed control can be calculated as follows. Let T_a denote the time available for the transmission of aperiodic data within a sampling period. According to Figure 3, in the centralized case

$$T_{a,c} = \max(0, \frac{T_s}{2} - T_{c,ref} - \frac{l_{o,a}}{r}). \quad (8)$$

In the case of distributed control, the frame collecting status information is suitable for the aperiodic traffic, which

in turn results

$$T_{d,a} = \max(0, \frac{T_s}{2} - T_{c,dist} - \frac{l_{o,a}}{r}). \quad (9)$$

Since aperiodic transmission can happen in $T_{a,*}, * \in \{c, d\}$ time within each sampling period of T_s , the transmission rate becomes

$$R_{a,*} = \frac{T_{a,*}}{T_s} r. \quad (10)$$

In the specific case considered in this study, $T_{c,ref} = T_{c,dist}$ and consequently $R_{a,c} = R_{a,d} = R_a$, meaning that the transmission rate of aperiodic traffic will be the same under the centralized and under the distributed implementation.

Figure 6 shows R_a as a function of S , the number of slaves, considering centralized and distributed control. It is clear that the achievable rate is a linear function of S . Note that the minimum payload size of the EtherCAT frame does not have any effect, since anyway the payload is extended with the aperiodic traffic. The forwarding time t_f has significant effect as the network gets larger. Still, a transmission rate of at least 20-30 Mbit/s is achievable for many cases. These rates are suitable for the CM application, but are significantly lower than the transmission rate of the bus, even at small S . This shows again the limiting effects of constraints of the EtherCAT protocol. In Section III-B the data size of the monitoring data was estimated to reach 3 MBytes, which means that the monitoring data is transmitted within milliseconds.

The analysis above demonstrates that multi-drive control and fast delivery of monitoring data are possible over today's wired industrial networks. The multi-drive control application can be realized both with centralized control, with the need of a powerful central controller, as well as with distributed control, building on the computation capabilities of the drives themselves.

V. CONCLUSIONS

The objective of this work was to evaluate how today's industrial network technologies can serve the large variety of new applications with diverse performance requirements. The specific case of the popular EtherCAT technology was considered together with the emerging applications of condition monitoring and dynamic control of multi-drive systems. The detailed numerical study leads to some general insights for future system and protocol design:

- While many of the industrial network protocols are designed to serve periodic delay-constrained traffic, there is often a straightforward, though protocol specific, solution to integrate low-priority aperiodic communication.
- The available communication rate (for example, 100 Mbit/s for EtherCAT) may give some indication on the expected limits and performance of the application. However, the need of periodic delay-constrained transmissions and the particularities of the protocols may limit the utilization of the communication resource significantly.

- While the constraints of the protocols may have negative effect on the general network utilization, they also lead to tractable performance evaluation and scalability properties that are easy to consider for the dimensioning of the network. For example, in the case of EtherCAT it was shown that the allowed number of nodes decreases linearly with increased computation times, while the achievable transmission rate of the aperiodic traffic decreases linearly with increasing number of network nodes.

The results of this study showed that the high-speed wired industrial network standards can accommodate the emerging applications. However, some of these applications, like multi-drive control, are very demanding, and further research and development is needed to support them with wireless connectivity.

REFERENCES

- [1] F. Benzi, G. S. Buja, and M. Felser, "Communication architectures for electrical drives," *IEEE Trans. Ind. Informat.*, vol. 1, no. 1, pp. 47–53, Feb. 2005.
- [2] M. Wollschlaeger, T. Sauter, and J. Jasperneite, "The Future of Industrial Communication: Automation Networks in the Era of the Internet of Things and Industry 4.0," *IEEE Ind. Electron. Mag.*, vol. 11, no. 1, pp. 17–27, Mar. 2017.
- [3] A. W. Colombo, S. Karnouskos, O. Kaynak, Y. Shi, and S. Yin, "Industrial Cyberphysical Systems: A Backbone of the Fourth Industrial Revolution," *IEEE Ind. Electron. Mag.*, vol. 11, no. 1, pp. 6–16, Mar. 2017.
- [4] B. Galloway and G. P. Hancke, "Introduction to Industrial Control Networks," *IEEE Communications Surveys Tutorials*, vol. 15, no. 2, pp. 860–880, 2013.
- [5] M. Riera-Guasp, J. A. Antonino-Daviu, and G. A. Capolino, "Advances in Electrical Machine, Power Electronic, and Drive Condition Monitoring and Fault Detection: State of the Art," *IEEE Trans. Ind. Electron.*, vol. 62, no. 3, pp. 1746–1759, Mar. 2015.
- [6] G. Zanuso, V. Fodor, L. Peretti, and O. Wallmark, "Networked Electric Drives in the Industry 4.0," in *2018 21st International Conference on Electrical Machines and Systems (ICEMS)*, Oct. 2018, pp. 724–729.
- [7] G. A. Capolino, J. A. Antonino-Daviu, and M. Riera-Guasp, "Modern Diagnostics Techniques for Electrical Machines, Power Electronics, and Drives," *IEEE Trans. Ind. Electron.*, vol. 62, no. 3, pp. 1738–1745, Mar. 2015.
- [8] J. H. Jung, J. J. Lee, and B. H. Kwon, "Online Diagnosis of Induction Motors Using MCSA," *IEEE Trans. Ind. Electron.*, vol. 53, no. 6, pp. 1842–1852, Dec. 2006.
- [9] T. M. Parreiras, J. C. G. Justino, A. V. Rocha, and B. d. J. C. Filho, "True Unit Power Factor Active Front End for High-Capacity Belt-Conveyor Systems," *IEEE Trans. Ind. Appl.*, vol. 52, no. 3, pp. 2737–2746, May 2016.
- [10] R. I. Davis, A. Burns, R. J. Bril, and J. J. Lukkien, "Controller Area Network (CAN) schedulability analysis: Refuted, revisited and revised," *Real-Time Systems*, vol. 35, no. 3, pp. 239–272, Jan. 2007.
- [11] M. Raza, N. Aslam, H. Le-Minh, S. Hussain, Y. Cao, and N. M. Khan, "A Critical Analysis of Research Potential, Challenges, and Future Directives in Industrial Wireless Sensor Networks," *IEEE Communications Surveys Tutorials*, vol. 20, no. 1, pp. 39–95, 2018.
- [12] M. Luvisotto, Z. Pang, and D. Dzung, "High-performance wireless networks for industrial control applications: New targets and feasibility," *Proceedings of the IEEE*, vol. 107, no. 6, pp. 1074–1093, June 2019.
- [13] B. Martinez, C. Cano, and X. Vilajosana, "A square peg in a round hole: The complex path for wireless in the manufacturing industry," *IEEE Communications Magazine*, vol. 57, no. 4, pp. 109–115, April 2019.
- [14] S. Mubeen, J. Mäki-Turja, and M. Sjödin, "Extending worst case response-time analysis for mixed messages in Controller Area Network with priority and FIFO queues," *IEEE Access*, vol. 2, pp. 365–380, 2014.
- [15] G. Zanuso and V. Fodor, "Bulk data transfer in Controller Area Networks for Industry 4.0," in *2020 IEEE International Conference on Industrial Cyber Physical Systems (ICPS)*, Jun. 2020.
- [16] L. Lo Bello, E. Bini, and G. Patti, "Priority-driven swapping-based scheduling of aperiodic real-time messages over ethercat networks," *IEEE Transactions on Industrial Informatics*, vol. 11, no. 3, pp. 741–751, June 2015.
- [17] A. Gandhi, T. Corrigan, and L. Parsa, "Recent Advances in Modeling and Online Detection of Stator Interturn Faults in Electrical Motors," *IEEE Transactions on Industrial Electronics*, vol. 58, no. 5, pp. 1564–1575, May 2011.
- [18] The EtherCAT Technology Group website. [Online]. Available: <https://www.ethercat.org/>
- [19] N. Bibinagar and W. Kim, "Switched Ethernet-Based Real-Time Networked Control System with Multiple-Client-Server Architecture," *IEEE/ASME Transactions on Mechatronics*, vol. 18, no. 1, pp. 104–112, Feb. 2013.
- [20] D. E. Lee, Q. V. Nguyen, T. Kim, J. Y. Moon, and J. W. Jeon, "Development of independent EtherCAT slave module and application to closed loop step motor drive with multi-axis," in *2016 International Conference on Computing, Communication and Automation (ICCCA)*, Apr. 2016, pp. 912–917.
- [21] V. Q. Nguyen, N. Van Phan Tran, H. N. Tran, K. M. Le, and J. W. Jeon, "A closed-loop stepper motor drive based on EtherCAT," in *IECON 2017 - 43rd Annual Conference of the IEEE Industrial Electronics Society*, Oct. 2017, pp. 3361–3365.
- [22] G. Prytz, "A performance analysis of ethercat and profinet irt," in *2008 IEEE International Conference on Emerging Technologies and Factory Automation*, Sep. 2008, pp. 408–415.

VI. BIOGRAPHIES

Giovanni Zanuso received his B.Sc. and M.Sc. in mechatronics engineering from the University of Padua, Italy, in 2007 and 2011 respectively. In 2012, he joined ABB Corporate Research, Sweden, where he worked as a research engineer in the machines and drives team. Since November 2016 he is a PhD student at KTH Royal Institute of Technology in Stockholm, Sweden. His current research interests include diagnostics for electric drives and industrial communication networks.

Viktoria Fodor received her M.Sc. and Ph.D. degrees in computer engineering from the Budapest University of Technology and Economics in 1992 and 1999, respectively. In 1999 she joined KTH Royal Institute of Technology, where she now acts as professor in the Division of Network and Systems Engineering. Her current research interests include network performance evaluation and protocol design for internet of things, distributed machine learning and edge computing.

Luca Peretti was born in 1980. He received the M.Sc. degree in Electronic Engineering in 2005 from the University of Udine, Italy, and the Ph.D. degree in Mechatronics and Industrial Systems from the University of Padova, Italy, in 2009. Currently, he is an Associate Professor with the Division of Electric Power and Energy Systems, KTH Royal Institute of Technology, Stockholm, Sweden. His current research interests include diagnostics and sensorless control for electric drives, multiphase machines and drives, for industrial, wind energy and automotive applications.

Oskar Wallmark (S01M06SM18) was born in 1976. He received the M.Sc. degree in Engineering Physics and the Ph.D. degree in Electric Power Engineering from the Chalmers University of Technology, Gothenburg, Sweden, in 2001 and 2006, respectively, and the Docent degree from the KTH Royal Institute of Technology, Stockholm, Sweden. He is currently an Associate Professor with the Division of Electric Power and Energy Systems, KTH Royal Institute of Technology, Stockholm, Sweden. His current research interests include control and analysis of electric drives, especially for automotive applications.