

A Case for Integrated Data Processing in Large-Scale Cyber-Physical Systems

René Glebke*, Martin Henze*, Klaus Wehrle*, Philipp Niemietz[§], Daniel Trauth[§], Patrick Mattfeld[§], Thomas Bergs[§]

*Communication and Distributed Systems, RWTH Aachen University, Germany

[§]Machine Tools and Production Engineering, RWTH Aachen University, Germany

{glebke, henze, wehrle}@comsys.rwth-aachen.de

{p.niemietz, d.trauth, p.mattfeld, t.bergs}@wzl.rwth-aachen.de

Abstract

Large-scale cyber-physical systems such as manufacturing lines generate vast amounts of data to guarantee precise control of their machinery. Visions such as the Industrial Internet of Things aim at making this data available also to computation systems outside the lines to increase productivity and product quality. However, rising amounts and complexities of data and control decisions push existing infrastructure for data transmission, storage, and processing to its limits. In this paper, we exemplarily study a fine blanking line which can produce up to 6.2 Gbit/s worth of data to showcase the extreme requirements found in modern manufacturing. We consequently propose integrated data processing which keeps inherently local and small-scale tasks close to the processes while at the same time centralizing tasks relying on more complex decision procedures and remote data sources. Our approach thus allows for both maintaining control of field-level processes and leveraging the benefits of “big data” applications.

1. Introduction

Modern manufacturing systems generate orders of magnitude more data than actual goods. Cyber-physical information systems aim at making these vast amounts of data accessible, interpretable, and connected, to derive insights and knowledge [1]. These serve as a foundation for realizing visions variously termed the Industrial Internet of Things, Industry 4.0, or the Internet of Production [2]. Increasingly digitized production processes are envisioned to leverage data from various sources, including suppliers, manufacturing, development steps, and product usage. This enables highly-integrated value chains that employ data analytics to derive insights about detailed aspects of the production processes, yielding increased product quality [3] and more efficient manufacturing environments [4].

One distinct characteristic of cyber-physical information systems in the context of production is their ability to utilize real-time data collected during one manufacturing process to affect this or other processes [5], potentially integrated into larger-scale information systems for supply chain management. For example, a close monitoring of manufacturing processes makes it easier to detect and react to sudden changes, e.g., in raw material properties, environmental conditions, or equipment failures, consequently reducing product rejects. At the same time, it also enables unprecedented traceability of goods: Individual products can be traced down to the manufacturing conditions of individual parts or materials [6], enabling downstream producers to assess overall quality and quickly react in cases of fluctuations in quality.

However, as the amount of data generated by manufacturing systems continues to increase, more pressure is put on the infrastructure transmitting, storing, and processing this data [7]. Eventually, existing communication infrastructure will be saturated. Likewise, processing and storage capacities close to manufacturing processes are typically highly specialized, offering reactions to input signals down to the low millisecond range but very limited potential for scaling or adapting to increasing data rates. In contrast, off-site computing and storage [8] can easily scale with increasing demands but suffer from throughput limitations and unpredictable latencies and jitter, rendering completely remote data handling infeasible.

In this paper, we focus on the potential of integrated data processing in large-scale cyber-physical systems to cope with the latency and bandwidth impacts of ever-increasing data generation. We propose an integrated and adaptive approach which facilitates data processing appropriate for the scopes and complexities of the respective tasks: Local and small-scale problems shall be solved locally, while more complex decision processes can be handled in increasingly centralized fashions. Consequently, we present methods to (i) selectively preprocess data

close to its origin to reduce network load; (ii) distribute parts of the control logic into the network to cater for scalability limitations of current control equipment and increasingly tight decision time bounds in manufacturing; and (iii) integrate off-the-field processing capabilities to allow for the creation of detailed process models that can help increasing efficiency and product quality. By integrating data processing in manufacturing systems, we pave the way towards large-scale cyber physical information systems.

In the following, we first highlight the value of big data in manufacturing (Section 2), before addressing best practices in data generation, communication, and processing in this field (Section 3). We then present a fine blanking plant as the ideal use case to study the future demands for data processing in manufacturing systems (Section 4). Finally, we discuss the different building blocks to realize our proposed integrated data processing approach (Section 5).

2. The value of big data in manufacturing

Since its inception in the early 2000s, the term “big data” has been used both for the fact and the hope that ever-increasing numbers of data generated at ever-shorter time scales can be utilized to improve several aspects of society and business, e.g., through personalized suggestions in online shops, improved policy making [9], and advances in healthcare [10].

Especially in manufacturing and production engineering, data has already become an important part and can serve as a foundation for substantial revenue increases [11]. Here, big data can improve aspects such as supply chain management, marketing, and support by harnessing readily-available or easily-obtainable “soft” data on supply product prices, customer satisfaction, or market analyses. A study of the McKinsey Global Institute [11] argues that an increased use of big data can lead to decreases of, e.g., 25 % to 50 % in product development costs and time to market based on crowdsourcing and advanced life cycle management. Likewise, the study anticipates a gain in profit margins by 2 % to 3 % based on better demand forecasting.

While these numbers already constitute substantial gains, we, in this paper, focus on another aspect, which is more directly related to the core of manufacturing: The manufacturing process itself. Most data currently generated on shop floors is discarded once a perceived “key value” such as controlling the machinery has been extracted [9, 12]. Why then would manufacturers, given their large-scale, heavy machinery and thus expensive long-term investments, want to invest into big data processing technology also in their manufacturing lines,

especially when these lines already efficiently produce high-quality products and when there are gains (as above) that are probably more easily obtainable?

The answer to this question are possible *competitive advantages in production and product quality*. Following the vision of the Internet of Production, context-aware data collected during the manufacturing process itself shall, together with data derived during development and product usage, be made available in real-time to enable cross-domain collaboration during the complete life cycle of a product [2]. As a result, manufacturers can reduce material costs, tool abrasion, and machine failures while improving product quality. We study the benefits resulting from an integrated processing of big data in a case study of a fine blanking plant in Section 4.2, showing that also gradually retrofitting existing machinery can result in gains worthwhile the initial investments.

However, simply tapping into each available data source might not yield much additional gain from a certain point on – known as the “low value density” of big data [13]. The value that can be created from machinery data will strongly depend on whether useful portions can be properly identified and processed, requiring access to all data in the first place. We, in this paper, thus focus on the challenge of *processing vast amounts of data*. As we show in the following, current technology and data processing prevents – or at least severely limits – the processing of large amounts of data in industrial settings. Data processing in cyber-physical information systems hence needs to be fundamentally rethought to fully embrace the value of big data.

3. Data processing in industrial scenarios

While data processing in the consumer and office market is clearly dominated by few technological standards such as Ethernet/IEEE 802.11, IPv4/IPv6, and TCP/UDP, more than 20 interconnection technologies (referred to as *fieldbuses*) are currently employed on the field and process level at the bottom of the *automation pyramid* in cyber-physical information systems as shown in Figure 1 [14]. Prominent examples for protocols at the field and process level in industrial scenarios are the early HART developed by Rosemount, Profibus and Profinet endorsed by Siemens, Modbus by Schneider Electric, and EtherCAT by Beckhoff.

This high variety of technologies is rooted in fundamentally different requirements of process and machinery control when compared to the consumer world: Physical processes operating at high speeds or with hazardous materials have to be tightly controlled, i.e., the control system needs to guarantee *low* and

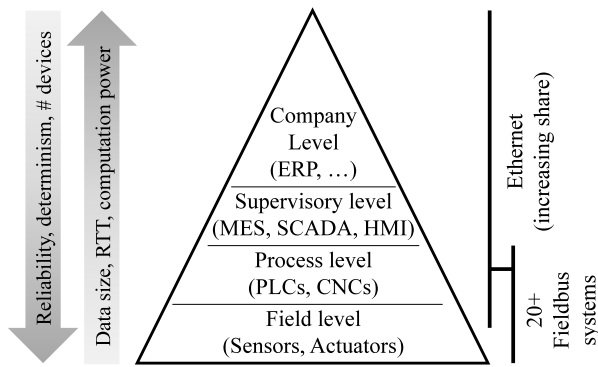


Figure 1. In the typical organization of industrial systems, constraints increase towards the bottom, while computation power increases towards the top.

predictable upper bounds on data communication and calculation times. Otherwise, reactions to conditions or events on the field level may arrive too late, with potentially catastrophic consequences [15]. Consumer equipment, such as PCs or smart TVs, in contrast, rarely has to meet hard deadlines; misses mean inconveniences to the users but no potential loss of control over intricate processes. As the arrows in Figure 1 indicate, demands with regard to time-critical handling of data decrease towards the top of the automation pyramid in traditional industrial systems as the upper layers increasingly integrate humans into the decision loops, leading to less time-critical tasks. For example, company-level planning tools require large volumes of data to be communicated and processed, but not in real-time.

To address determinism and reliability requirements, vendors of control systems have adopted a communication and computation model on the field level as shown in Figure 2. In this regime, the central entity is the eponymous *controller*, usually a so-called *Programmable Logic Controller (PLC)*, a ruggedized computing system specifically designed for industrial purposes. Readings from sensors are either transmitted over a dedicated, per-sensor connection directly to the controller, or via a shared communication channel in which the controller mediates channel access.

The controller *periodically* computes the next actions to take based on the current inputs and uses similar output channels to instruct actuators on their operation until the end of the control round, when the new instructions are sent. Most of the employed sensors and actuators are built with broad applicability in mind and possess little to no internal processing logic. Instead, they rely on the controller for the interpretation of their readings or an adaption of their settings. The system behavior is hence completely determined by the software in the controller, which acts as the sole communication partner of all sensors and actuators.

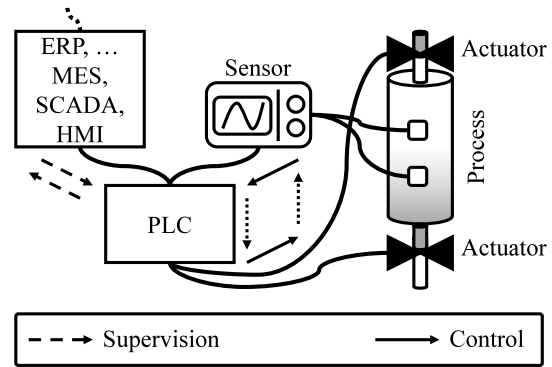


Figure 2. Physical processes enforce tight constraints on the control loops at the field level. Vendor-specific solutions hinder advances below the supervisory level.

The controller-centric design at the lower levels of the automation pyramid has a number of ramifications. First, the concentration of processing logic within the controller reduces costs by rendering an implementation of such functionality on the individual devices unnecessary. Second, encapsulation overheads for data items are reduced to a minimum as the controller exactly knows the used data formats. Similarly, as only a strictly defined set of devices is communicating, only minimal addressing overhead is incurred. On the downside, despite a rise of Ethernet / IP also on the field level [14], the employed communication technologies still differ between controller manufacturers, resulting in a fragmented market and potential technology lock-ins.

Most importantly, however, the systems scale only vertically, i.e., size and capabilities of the systems are limited by the controllers, which are often expensive to extend or replace. Thus, it is strenuous and expensive to augment existing system architectures with additional equipment and functionality, especially when considering time-critical decision cycles in the context of cyber-physical information systems. Furthermore, several of the scenarios envisioned for future production processes do not only require real-time decisions to be also taken on higher levels of the automation pyramid (e.g., changing field-level parameter sets based on a large-scale model of the process), they also necessitate increasing amounts of data to be transmitted.

Consequently, to take full advantage of the potentials of big data, we have to integrate data processing functionalities into the lower levels of the automation pyramid or even rethink the pyramidal structure of data communication and processing in general. In the following, we present a detailed analysis of a manufacturing line which would particularly benefit from integrated data processing and derive the demands that this manufacturing line poses with respect to computation and communication.

4. Case study: A fine blanking line

We selected a fine blanking line as use case to study the future demands for data generation and processing in manufacturing systems, especially considering bandwidth and latency requirements. In a nutshell, fine blanking is an effective and economical precision forming process for manufacturing billions of identical work pieces. Fine blanking reduces costs in the production of various products such as metal components used in automobile or aerospace industries [16]. However, due to uncertainties prevalent in manufacturing, manufactured pieces are not identical, requiring cost-intensive secondary finishing steps.

A typical setting of a fine blanking line as deployed in our shop floor is shown in Figure 3. First, a decoiler decoils the raw material, which is usually a <1 mm to 20 mm thick and 50 mm to 250 mm wide metal coil which then enters the leveler [16] where the material is as far as possible freed of its residual stresses. Since pressing is a discontinuous process but decoiling and leveling are continuous ones, there is a need for a sling which acts as a buffer between both processes. Eventually, before the actual fine blanking process can take place in the press, a lubricant film has to be applied on the metal sheet which is then cut by the tool of the press. At each step in the manufacturing line, machines and additional sensors generate data that needs to be propagated to and processed by the other machines involved in the process. As a simple example, the state of the press influences the working speed of the decoiler, leveler, and lubricator. In a manufacturing plant of a fine blanking company the number of fine blanking lines can vary in a range of 4 to 24. Additionally, those plants can also contain machines used to operate secondary operations that are required to finish work pieces.

The goal of fine blanking is to produce identical work pieces. However, regardless of the good quality in general, all work pieces differ in their quality features that are the tear-off surface, fine cracks in the functional surface, the burr and, especially, die roll [17]. The die roll is a roundish defect at the cutting edge of the work piece, caused by the plasticity of the material, and it is the most important potential defect in fine blanking as it reduces the dimensional accuracy of the cutting edge [18]. Thus, thicker sheet metal strips and secondary machining must be used to compensate for the die roll and improve dimensional accuracy. There are several process parameters that influence the die roll size such as the design of the tool used, the work piece geometry and thickness, material properties [19], the cutting speed, and other process parameters that are set during the process [20–22]. Thus, the fine blanking

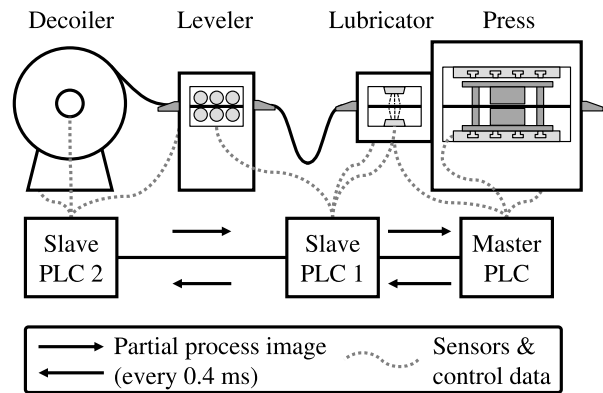


Figure 3. The four main processing steps of the fine blanking process considered in our case study are controlled by a series of hierarchically organized PLCs.

process is an intricate process with no generic process setup for each component produced.

In the following, we specifically study a fine blanking line consisting of a coil system and leveler from ARKU and a servo-mechanical Feintool XFT 2500 speed fine blanking press that achieves both high cycle rates of 140 strokes/min and variable stroke characteristics. As we operate our line in a research setting, we are able to completely monitor all process data and to attach additional sensors, such as for important forces within the cutting process, to fully exploit the potential of the fine blanking process in the context of cyber-physical information systems.

4.1. Control and communication structure

To better understand how the fine blanking process is controlled and how it can be optimized based on data, we analyze both the data involved as well as its flow through the machinery and control systems.

The control system follows the structure introduced in Section 3. A simplified schematic of the wiring is shown in the lower part of Figure 3. Sensors and actuators of the four processing steps (decoiler, leveler, lubricator, and press) are connected to a number of PLCs, which form a hierarchical network that is overseen by the PLC at the press (the so-called *master PLC*; all other PLCs are referred to as *slaves*). The master PLC runs the general control program of the line and instructs its slaves. The slaves control sensors and actuators in specific parts of the line on behalf of the master via internal fieldbus networks (dotted gray lines).

The master PLC maintains a process image comprising all data necessary for controlling the line. To control the equipment attached to the slaves, a part of the master image is conveyed towards the slaves at a frequency of 2.5 kHz (equal to one signal every 0.4 ms);

the slaves send select information to the master at the same rate. As broken down in Table 1, the process image of the master consists of 1406 individual sensor readings and actuator control signals in the final configuration level. Built-in signals of the machinery amount to 1333 or 94.8% of the generated data items, with the rest resulting from additional sensors attached or planned in our research environment. The built-in signals are updated at the same frequency as the process images are exchanged, yielding a data rate of 43.12 Mbit/s.

Today, machine operators have a wealth of experience of the processes and can sense possible defects in the process setup or upcoming failures due to vibration, noise, or visual effects. Hence, fully benefiting from the advantages of big data requires to deploy sensors in- and outside the machine. Within the machine itself, additional internal sensors can be deployed in the press, to measure forces in the leveler and the press, and to detect vibrations in the press as shown in Table 1. These sensors, however, need to be paced at frequencies as high as 1 MHz and may produce additional data rates of roughly 137.6 Mbit/s, $3.19\times$ the data rate produced by the built-in signals. Beyond that, some of the sensors are envisioned to perform best when paced at higher frequencies than the 2.5 kHz of the built-in equipment. Given the current process image distribution, it may hence become necessary to send process images or parts thereof more often, resulting in increases by several orders of magnitude in data rates.

Outside the machine, additional external sensors measuring the environment, infrared temperature sensors, and industrial cameras for monitoring the produced work pieces may add signals with a rate of up to 6.03 Gbit/s, resulting in an increase of two orders of magnitude from the baseline. More sensors to monitor influencing factors not yet covered by our setup may be added to the system at later points, further increasing the mass of data our infrastructure needs to handle.

Summing up our analysis, operating fine blanking lines with additional sensors will quickly overload the controller and network technologies at the field level. Considering larger plants with several lines operated in parallel, this effect will also become noticeable in the higher levels of the automation pyramid.

4.2. Potential of integrated data processing

The fine blanking process comprises various steps which can benefit from an increased availability of data. In the following, we give an overview of possible optimizations, ranging from cost reductions to improvements in the quality of the produced pieces.

Reducing material costs. The cost of produced work pieces in a fine blanking process is affected by material

Table 1. A fine blanking line equipped with built-in and additional sensors produces data at a frequency of up to 1 MHz and a data rate of 6.2 GBit/s.

	Signal group	Size (Bit)	Freq. (kHz)	Count	Data rate (Mbit/s)
Built-in sensors and control	Leveler	32	2.5	36	2.88
	Lubricator	32	2.5	32	2.56
	Press	64	2.5	3	0.48
		32	2.5	44	3.52
		16	2.5	237	9.48
		8	2.5	428	8.56
	Other equipment	1	2.5	481	1.2
		1984	2.5	1	4.96
		64	2.5	32	5.12
		48	2.5	35	4.2
Additional internal sensors	Press	16	2.5	4	0.16
		32	50	9	14.4
	Forces in leveler	32	20	2	1.28
		32	10	7	2.24
	Forces in press	32	50	4	6.4
		32	10	4	1.28
	Vibrations in press	32	1000	2	64
24		1000	2	48	
Additional external sensors	Environment	32	1	36	1.15
	Infrared temperature	2.458e6*	0.032	2	157
	Industrial cameras	1.68e8†	0.007	5	5880
			Sum	1406	6218.87

* 640x480 pixels, 8 bits per pixel † 14 megapixels, 12 bits per pixel

costs to a large extent, since only material with very high quality standards can be used for fine blanking. Consequently, a central optimization of the fine blanking process is to reduce the amount of used material. Because material in the height of the die roll has to be ground off in a secondary step to achieve dimensional accuracy of each work piece, an effective reduction of the die roll enables the manufacturer to buy thinner and, thus, cheaper metal coils. It is known that process parameters and material properties influence the die roll [19], therefore, an analysis of data related to these parameters and properties can result in models that automatically adjust process parameters in real-time to reduce the die roll, substantially reducing material costs.

Improving work piece quality. Beside the die roll, there are several other quality features of fine blanking work pieces, such as demolition and fine cracks in the functional surface [17]. Because work pieces produced by the fine blanking process are often security-critical components, a defect in one of those measures results in a rejected part. Hence, using the available process data to derive an accurate process quality prediction model for the sheared functional surface can possibly reduce the amount of rejected pieces drastically.

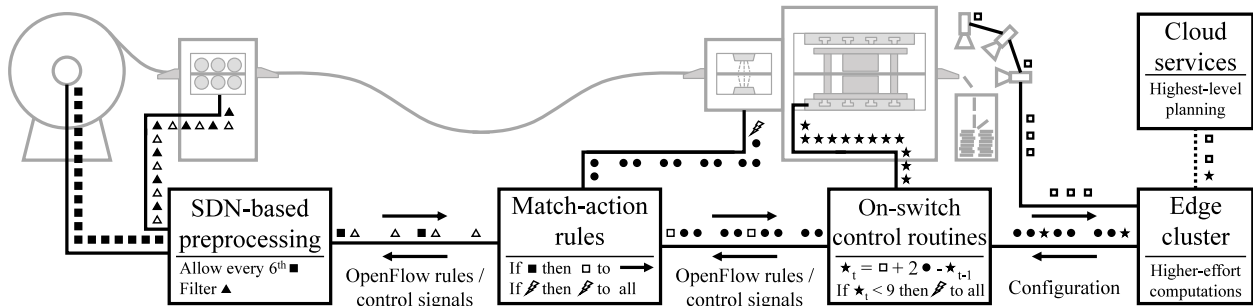


Figure 4. Our proposed architecture allows local tasks such as data preprocessing (left), emergency handling (center left), and small control routines (center right) to be solved on off-the-shelf hardware in the field-level network. It reduces network load and allows a tight integration of more powerful edge/cloud services (right).

Increasing tool life cycle. Another important potential for optimization lies in the punch abrasion within the tool used to cut the pieces. The blanking punch within the tool is the component that actually cuts the material. Hence, it has to be made of high-strength material and requires complex, and thus expensive, casting procedures [23]. However, as the coil strength can differ drastically over the coil length [24], there is a risk to wear off or even damage the punch tool when using the same process parameters (e.g., forces) for each punch. In the worst case, a complete sequence of the coil has to be considered to be not usable due to its inferior quality, since the cutting process could damage the blanking punch or other components of the tool. If we were able to monitor material properties and the applied lubricant film in real-time, sequences with greater thickness, higher strength, or bad lubricant distribution could be detected and the different forces used in the cutting process could be adjusted to avoid abrasion. Thus, we can increase the lifetime of costly tools and avoid interruptions due to broken tools.

Reducing machine failures. Just within the fine blanking press itself there are more than 100 different components, each producing control and status data. We can use this data to create an event log of the components in the press, lubricator, and leveler. This log can then be utilized to derive a process-flow model representing the complete flow of events during a fine blanking process. Computing such a model is non-trivial and computation-intensive but enables the manufacturer to validate the current process state and stop the process if it enters a state pointing towards an upcoming machine failure with potential tool damage. The speed of the fine blanking process (up to 140 strokes/min in the press) requires low latency computations since some failures can only be predicted a few strokes beforehand.

Although conceived with our distinct manufacturing line in mind, above optimizations are general enough to be applied to similar fine blanking lines in other

environments, emphasizing the general potential of cyber-physical information systems in manufacturing. However, as we have shown in Section 4.1, when maintaining our current control and communication structure, we will not be able to leverage this potential.

5. Integrated data processing for large-scale cyber-physical systems

Our case study of a fine blanking line highlights the future demands with respect to data generation and processing in manufacturing systems. Most notably, these demands result in huge bandwidth consumption and low latency requirements. To enable manufacturers to handle the ever-increasing amount of data collected in interconnected manufacturing systems, we propose an integrated and adaptive architecture for data processing in large-scale cyber-physical systems as shown in Figure 4. In a nutshell, our approach aims to solve those problems locally that are local and small-scale in nature while realizing processes more centrally that require access to data from different sources and often more complex computations. To achieve such an integrated and adaptive architecture, we in the following discuss approaches to (i) select, reduce, and preprocess field-level data; (ii) allow the processing of data also in the low-level interconnection fabrics; and (iii) offload processing steps that cannot be performed by in-field equipment to edge and cloud resources.

5.1. Data selection, reduction and preprocessing

Our analysis of a fine banking plant revealed a vast number of data points generated per second. Indeed, if only the built-in data of a fine blanking line’s machinery were to be recorded, e.g., to reduce machine failures (cf. Section 4.2), this would result in a total of approx. 19.4 GB or 12 billion (short scale) of individual data items per hour for that line only.

For the additional sensors, this number rises to approx. 2.78 TB or 29.41 billion items, respectively. Operating more than one line will hence quickly overload most current network and storage technologies, and the pressure will become even more pronounced with larger manufacturing lines or additional data sources. Due to limitations in the built-in logic of the field-level equipment (cf. Section 3), it is often impossible to selectively transmit only specific data items, although some of these items might yield only marginal gains in analyses (cf. Section 2).

Consequently, it may become inevitable to allow future networks in large-scale cyber-physical systems to permit only those sensor values to travel through the network which have been deemed necessary either for the operation of the machinery or selected by downstream processes. Considering the increasing proliferation of Ethernet- and IP-based communication on the field level (cf. Section 3), such functionality can be based on software-defined networking (SDN). SDN protocols such as OpenFlow [25] allow the installation of fine-grained forwarding rules in routers and switches (as shown on the bottom left of Figure 4), including possibilities to drop packets (e.g., those not currently needed by downstream processes) or to reroute them (e.g., towards analyzers) without intervening with the operation of the sensors themselves. Likewise, rate limitation may be used match signal frequencies to a controller's or downstream analysis process' periodicity.

By selecting which data to relay through the network, we can already relieve some stress exerted on the infrastructure. Still, recent advances in SDN technology also allow for tackling further problems: Considering our fine blanking case study (cf. Section 4), we notice sensors that are organized into arrays, e.g., the force sensors in the press unit, which are uniformly distributed over the punch. While this yields a finer resolution of the forces exerted on the material, an individual value only conveys a meaning in relation to the other values, i.e., when it largely diverges from the forces sensed by its counterparts. Hence, when the flow of the force sensor values is transmitted through the network, we can apply statistical aggregation methods, e.g., mean or median, as well as outlier detection on the data. We can then relay only such data items that deviate from the rest or surpass a defined threshold, indicating trends or changes in the operational state of the system.

Such methods have previously been successfully applied in wireless sensor networks [26], where the nodes suffer from comparably limited processing abilities as devices at the bottom of the automation pyramid. While SDN methods have not originally been developed to perform elaborate operations on payloads

(in this case, the sensor values), techniques such as P4 [27] enable the implementation of more elaborate functionality including a rudimentary manipulation of the payloads, and have recently witnessed an increase in interest by network equipment vendors [28].

The aforementioned methods rely on a common understanding of the involved equipment with regard to data formats. However, the highly fragmented fieldbus market (cf. Section 3) also resulted in a plethora of higher-level data exchange protocols. Only very few of these protocols are directly interoperable and we expect that selected messages may need to be converted by the SDN switches to interconnect previously incompatible equipment. Some formats such as CANopen and CIP have already seen adoptions by multiple vendors sensors and actuators [14], so that the number of conversions necessary will likely decrease with an increased adoption of such standards.

5.2. Distributed and in-network processing

While data selection, reduction, and preprocessing realizes integrated data processing for data used for monitoring and later offline analyses, data used for actually *controlling* equipment interacting with the physical world needs to be treated such that tight control loops can still function within the limits of their respective time bounds. Most time-critical calculations in manufacturing are currently performed on centralized PLCs. Our analysis of the fine blanking line in Section 4.1 has shown that in practice, especially the scalability limitations of centralized controllers are overcome by introducing a hierarchy of controllers exchanging their process images. Since each PLC maintains its own, centralized view of an albeit smaller problem, this approach is both costly (PLCs are expensive) and does not alleviate the fundamental limitation of only vertical scalability (cf. Section 3).

We thus need to introduce methods that allow an *inherent horizontal scalability* of processing on the field level, essentially by enabling additional equipment (not just controllers) to analyze and act upon data that is generated on the field level, thus relieving stress from centralized controllers. There have been first attempts at designing controllers that are distributed within networks. The IEC-61499 standard, e.g., addresses the question of how to create control subroutines that can be distributed among multiple controllers that communicate with each other [29]. The standard is relatively young (the first edition was published in 2005) and has remained widely unused due to inconsistencies in its interpretation by the forerunners of its adoption [30], leaving plant owners unsure

about which of the approaches to choose. However, the standard's general approach of allowing *various* elements within the network to process data, as long as they can meet the required time bounds, is compelling.

The most likely (since ubiquitously available) resources for additional computations are personal computers and their low-cost miniature counterparts such as Raspberry Pis [31]. However, most operating systems, including the popular Linux kernel, by design introduce packet processing overheads due to scheduling, context switches, and security precautions, so that small data items – such as sensor readings – may be hard to process in a timely fashion. Various attempts have been made to counter these effects, ranging from optimizing packet handling [32, 33], partially offloading computations from the application level to the kernel [34], or bypassing the kernels altogether [35].

These approaches are especially useful for small, relatively static answers to specific requests, e.g., for DNS or HTTP servers. Related work shows noticeable gains compared to kernel-based approaches in both response time (e.g., down to one-digit milliseconds [34]) and general throughput (more than one order of magnitude [35]). As most sensor and actuator signals are small, when the control laws are rather static and straightforward, similar gains can be achieved in cyber-physical systems. Ultimately, we can use off-the-shelf hardware in combination with – or in place of – traditional controllers in such situations.

While these techniques still employ general-purpose hardware and software, more radical approaches aim at using *network hardware* for the execution of programs (as depicted in the center of Figure 4). By employing networking programming languages such as P4 [27] (cf. Section 5.1), we can offload generalized match-action-based rules to switches and routers. For example, simple reactions such as stop signals in emergency situations (as indicated by specific sensor readings) can be pushed directly to the network hardware closest to the respective sensor(s), thus reducing the reaction time to the minimum possible.

Following a different approach, the expressiveness of programs that can be executed on network devices can be further increased: By offloading so-called Extended Berkeley Packet Filter (eBPF) programs, originally developed for fast handling of network packets in the kernel [36], to network hardware, control algorithms can be run within the network [37, 38]. This leads to dramatic increases in the control quality under deteriorating latency and jitter conditions. To empirically verify that eBPF programs exhibit the upper bound guarantees of PLCs, the runtime bounds of in-network functions can be predicted [39]. Although

eBPF programs are not Turing complete [40], these works show the general feasibility of realizing control operations using in-network computation. If even more complex computations are required, Tasklets allow distributing small computation units to lightweight virtual machines distributed along the communication path under various optimization criteria [41].

The idea of distributed control constitutes a considerable deviation from the prevailing design space of control applications. Furthermore, the conditions under which theoretically optimal and tractable distributed control laws can be found is an open question [42]. However, we did not identify fundamental limitations pertaining distributed control in large-scale cyber-physical systems. Hence, the above approaches are viable options for alleviating the scalability limitations of centralized controllers.

5.3. Handling higher-effort computations

Data selection, reduction, and processing as well as distributed and in-network processing are beneficial especially to operations which are somewhat confined, e.g., because they deal only with relatively small amounts of data at once or are of local nature. However, the visions of the Industrial Internet of Things, Industry 4.0, and the Internet of Production also necessitate handling complex operations on very large amounts of data at once. Since the sheer numbers of generated data items exceed the capabilities of field-level devices (cf. Section 3), we need the ability to process data also outside the field to enable more complex computations.

Cloud services with their virtually unlimited scalability offer viable solutions for many use-cases that incur massive amounts of data to be processed [8,43,44] and can be used to store and analyze data generated at different manufacturing sites, which may be physically far apart. In scenarios of plants with dozens of lines generating data, limitations in the Internet uplinks of the plants may however ultimately limit the amount of data items that can be transferred, even if data reduction (cf. Section 5.1) is applied. Yet, it is not only the quantity but also the topicality of data that provides challenges: Many of the data items generated by manufacturing machinery are *streaming data*, so that in cases of latencies in their processing, the items may lose value [45]. Very complex computations such as updating a model to detect machine failures based on current and historical data (cf. Section 4.2) might take so long on unsuited systems that benefits are nullified.

On-premise (edge) clusters (shown on the right of Figure 4) enable manufacturers to perform those

analyses and model generations that cannot be outsourced to off-premise cloud infrastructure because of bandwidth restrictions [46]. Yet, as on-premise clusters need to provide local data storage, this approach quickly becomes costly and inefficient. Hybrid cloud solutions can use on-demand data storage and computing power to dynamically outsource computation efforts based on resource demands [47].

To manage data processing in such distributed systems, the lambda architecture realizes a software stack for distributed computation on virtually arbitrary amounts of data [48]. It consists of a streaming layer and a batch layer, both designed to scale horizontally with the cluster size. The task of the streaming layer is to fulfill low latency requirements and high throughput demands in a distributed computation model with a built-in, non-persistent, in-memory streaming database [49]. The batch layer, in turn, stores all generated data in a data lake, i.e., a distributed file system for different data formats usually realized on relatively cheap object storage, and can use the whole historical dataset to generate new models. For both layers, various distributed processing models exist, all focusing on specific optimization goals such as latency, throughput, fault tolerance, usability, or scalability [48].

Eventually, the derived models and predictions are persisted in non-relational “NoSQL” databases. Classical relational databases provide atomicity, consistency, isolation, and durability of transactions, which however renders them hard to scale out and vulnerable to partitioning [50]. NoSQL databases, in contrast, target both horizontal scalability with frequent reads and writes [51] and partition tolerance, with the drawback of lacking either full availability or consistency [52]. Best-effort-availability systems and eventually consistent models aim to mitigate these problems [53]. The Hadoop ecosystem offers a variety of options to implement such architectures with open-source frameworks [54].

6. Conclusion

Large-scale cyber-physical systems such as manufacturing lines nowadays generate massive amounts of data, the majority of which is currently being discarded after it has been used for controlling the manufacturing machinery. Recent academic and industrial trends such as the Industrial Internet of Things, Industry 4.0, or the Internet of Production however advocate taking advantage of the generated data for improving productivity and product quality.

Based on the real-world use case of a fine blanking line, we argue that the controller-centric organization of

data processing on the field level – the de-facto standard in manufacturing for decades – will prevent further advancements for data-driven manufacturing. Thus, we suggest an integrated and adaptive architecture which allows further devices in the field-level networks and at the nearby network edges to contribute processing power. We provide an overview of emerging networking and distributed computing technologies that are well-suited for realizing such an architecture.

Our approach allows for a distinction between local, time-bound and small-scale problems solvable in a distributed fashion as well as data-intensive, complex decision procedures that require potentially more centralized resources. It is hence both amenable to a wide variety of use-cases and inherently scalable. As such, we can realize integrated data processing in modern manufacturing systems and thus pave the way towards large-scale cyber physical information systems.

7. Acknowledgments

The authors would like to thank the German Research Foundation (DFG) for the kind support within the Cluster of Excellence “Integrative Production Technology for High-Wage Countries” and the Priority Programme 1914 (“Cyber-Physical Networking”).

8. References

- [1] A. W. Colombo et al., “Industrial Cyberphysical Systems: A Backbone of the Fourth Industrial Revolution,” *IEEE IEM*, vol. 11, no. 1, 2017.
- [2] S. Jeschke et al., “Industrial Internet of Things and Cyber Manufacturing Systems,” in *Industrial Internet of Things: Cybermanufacturing Systems*, Springer, 2017.
- [3] K. Liere-Netheler et al., “Drivers of Digital Transformation in Manufacturing,” in *HICSS*, 2018.
- [4] L. Monostori et al., “Cyber-physical systems in manufacturing,” *CIRP Annals*, vol. 65, no. 2, 2016.
- [5] J. Lee et al., “Industrial big data analytics and cyber-physical systems for future maintenance & service innovation,” *Procedia CIRP*, vol. 38, 2015.
- [6] G. Schuh et al., “Promoting Work-based Learning through INDUSTRY 4.0,” *Procedia CIRP*, vol. 32, 2015.
- [7] L. Spendla et al., “Concept of Predictive Maintenance of Production Systems in Accordance with Industry 4.0,” in *IEEE SAMI*, Jan 2017.
- [8] M. Henze et al., “A Comprehensive Approach to Privacy in the Cloud-based Internet of Things,” *Future Generation Computer Systems*, vol. 56, 2016.
- [9] L. Furtado et al., “Value Creation in Big Data Scenarios: A Literature Survey,” *Journal of Industrial Integration and Management*, vol. 02, no. 01, 2017.
- [10] S. Yu et al., “Networking for Big Data: A Survey,” *IEEE Comm. Surv. & Tuts.*, vol. 19, no. 1, 2017.
- [11] J. Manyika et al., “Big data: The next frontier for innovation, competition, and productivity.” McKinsey Global Institute, 2011.

- [12] V. Mayer-Schönberger and K. Cukier, *Big Data – A Revolution that will Transform How We Live, Work and Think*. Houghton Mifflin Harcourt, 2013.
- [13] A. Gandomi and M. Haider, “Beyond the hype: Big data concepts, methods, and analytics,” *Int. J. Inform. Mgmt.*, vol. 35, no. 2, 2015.
- [14] T. Sauter, “The Three Generations of Field-Level Networks – Evolution and Compatibility Issues,” *IEEE Trans. Ind. Electr.*, vol. 57, no. 11, 2010.
- [15] J. Hiller et al., “Secure Low Latency Communication for Constrained Industrial IoT Scenarios,” in *IEEE LCN*, 2018.
- [16] F. Klocke and A. Kuchle, *Manufacturing Processes*. Springer, 2009.
- [17] R. Schmidt et al., *Cold Forming and Fineblanking – A handbook on cold processing steel material properties component design*. Carl Hanser, 2007.
- [18] H. Voigts et al., “Dependencies of the die-roll height during fine blanking of case hardening steel 16mncr5 without v-ring using a nesting strategy,” *Int. J. Adv. Mfg. Tech.*, vol. 95, no. 5-8, 2018.
- [19] T. Lee et al., “Application of the Finite-Element Deformation Method in the Fine Blanking Process,” *J. Mat. Proc. Tech.*, vol. 63, no. 1, 1997.
- [20] J. Stanke et al., “Setup of a Parameterized FE Model for the Die Roll Prediction in Fine Blanking using Artificial Neural Networks,” *JPCS*, vol. 896, no. 1, 2017.
- [21] J. D. Kim et al., “A Study on the Effect of V-Ring Position on the Die Roll Height in Fine Blanking for Special Automobile Seat Recliner Gear,” in *ICMST*, 2011.
- [22] T. Kwak et al., “Finite element analysis on the effect of die clearance on shear planes in fine blanking,” *Journal of Materials Processing Technology*, vol. 130-131, 2002.
- [23] F. Birzer, *Forming and Fineblanking: Cost-effective manufacture of accurate sheet metal parts*. Verlag Moderne Industrie, 1997.
- [24] F. Van-Den-Berg et al., “In-line Characterisation of Microstructure and Mechanical Properties in the Manufacturing of Steel Strip for the Purpose of Product Uniformity Con,” in *WCNDT*, 2016.
- [25] N. McKeown et al., “OpenFlow: Enabling Innovation in Campus Networks,” *ACM SIGCOMM CCR*, vol. 38, no. 2, 2008.
- [26] Y. Zhang et al., “Outlier Detection Techniques for Wireless Sensor Networks: A Survey,” *IEEE Comm. Surv. & Tuts.*, vol. 12, no. 2, 2010.
- [27] P. Bosshart et al., “P4: Programming Protocol-Independent Packet Processors,” *ACM SIGCOMM CCR*, vol. 44, no. 3, 2014.
- [28] Praveen Bhagwatula, Cisco Systems, “Introduction to P4 and P4Runtime.” <https://xrdocs.io/cloud-scale-networking/blogs/2018-03-08-introduction-to-p4-and-p4runtime/>. Last accessed 2018-08-27.
- [29] *IEC 61499-1 – Function blocks – Part 1: Architecture*. International Electrotechnical Commission, 2012.
- [30] K. Thramboulidis, “IEC 61499: Back to the well proven practice of IEC 61131?,” in *IEEE ETFA*, 2012.
- [31] Raspberry Pi Foundation, “Raspberry Pi.” <https://www.raspberrypi.org/>. Last accessed 2018-08-27.
- [32] T. Marian et al., “NetSlices: Scalable Multi-core Packet Processing in User-space,” in *ACM/IEEE ANCS*, 2012.
- [33] S. Han et al., “PacketShader: A GPU-accelerated Software Router,” in *ACM SIGCOMM*, 2010.
- [34] F. Schmidt et al., “Santa: Faster Packet Delivery for Commonly Wished Replies,” in *ACM SIGCOMM*, 2015.
- [35] L. Rizzo, “Netmap: A Novel Framework for Fast Packet I/O,” in *USENIX ATC*, 2012.
- [36] S. McCanne and V. Jacobson, “The BSD Packet Filter: A New Architecture for User-level Packet Capture,” in *USENIX Winter*, 1993.
- [37] J. RÜth et al., “Demo: Towards In-Network Processing for Low-Latency Industrial Control,” in *IEEE INFOCOM*, 2018.
- [38] J. RÜth et al., “Towards In-Network Industrial Feedback Control,” in *ACM SIGCOMM NetCompute*, 2018.
- [39] F. Rath et al., “SymPerf: Predicting Network Function Performance,” in *ACM SIGCOMM Posters*, 2017.
- [40] S. Jouet et al., “Arbitrary packet matching in OpenFlow,” in *IEEE HPSR*, July 2015.
- [41] D. Schäfer et al., “Tasklets: ”Better than Best-Effort” Computing,” in *IEEE ICCCN*, 2016.
- [42] L. Lessard and S. Lall, “Quadratic Invariance is Necessary and Sufficient for Convexity,” in *ACC*, 2011.
- [43] M. Henze et al., “A Trust Point-based Security Architecture for Sensor Data in the Cloud,” in *Trusted Cloud Computing*, Springer, 2014.
- [44] M. Henze et al., “Network Security and Privacy for Cyber-Physical Systems,” in *Security and Privacy in Cyber-Physical Systems: Foundations, Principles and Applications*, Wiley-IEEE Press, 2017.
- [45] M. Stonebraker et al., “The 8 Requirements of Real-Time Stream Processing,” *ACM SIGMOD Record*, vol. 34, no. 4, 2005.
- [46] W. Shi et al., “Edge Computing: Vision and Challenges,” *IEEE IoT*, vol. 3, no. 5, 2016.
- [47] H. Zhang et al., “Intelligent Workload Factoring for a Hybrid Cloud Computing Model,” in *IEEE SERVICES*, 2009.
- [48] N. Marz and J. Warren, *Big Data: Principles and best practices of scalable real-time data systems*. Manning, 2015.
- [49] R. Ranjan, “Streaming Big Data Processing in Datacenter Clouds,” *IEEE Cloud Computing*, vol. 1, no. 1, 2014.
- [50] R. P. Padhy et al., “RDBMS to NoSQL: Reviewing Some Next-Generation Non-Relational Database’s,” *Int. J. Adv. Eng. Sc. and Tech.*, vol. 11, no. 1, 2011.
- [51] M. Henze et al., “Complying with Data Handling Requirements in Cloud Storage Systems.” arXiv preprint arXiv:1806.11448 [cs.NI], 2018.
- [52] S. Gilbert and N. Lynch, “Brewer’s Conjecture and the Feasibility of Consistent, Available, Partition-Tolerant Web Services,” *ACM SIGACT News*, vol. 33, no. 2, 2002.
- [53] S. Gilbert and N. Lynch, “Perspectives on the CAP Theorem,” *Computer*, vol. 45, no. 2, 2012.
- [54] S. Landset et al., “A survey of open source tools for machine learning with big data in the Hadoop ecosystem,” *J. Big Data*, vol. 2, no. 1, 2015.