COINRG
Internet-Draft
Intended status: Informational
Expires: September 10, 2020

I. Kunze
K. Wehrle
RWTH Aachen University

March 09, 2020

Industrial Use Cases for In-Network Computing
draft-kunze-coin-industrial-use-cases-02

Abstract

Cyber-physical systems and the Industrial Internet of Things are characterized by diverse sets of requirements which can hardly be satisfied using standard networking technology. One example are latency-critical computations which become increasingly complex and are consequently outsourced to more powerful cloud platforms for feasibility reasons. The intrinsic physical propagation delay to these remote sites can already be too high for given requirements. The challenge is to develop techniques that bring together these requirements. Utilizing available computational capabilities within the network for in-network computing concepts can be a solution to this challenge. This document discusses selected industrial use cases to demonstrate how in-network computing concepts can be applied to the industrial domain and to point out essential requirements of industrial applications.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at https://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on September 10, 2020.

Copyright Notice

Copyright (c) 2020 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (https://trustee.ietf.org/license-info) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction

The Internet is based on a best-effort network that provides limited guarantees regarding the timely and successful transmission of packets. This design-choice is suitable for general Internet-based applications, but specialized industrial applications demand a number of strict performance guarantees, e.g., regarding real-time capabilities, which cannot be provided over regular best-effort networks.

Enhancements to the standard Ethernet such as Time-Sensitive-Networking [TSN] try to achieve the requirements on the link layer by statically reserving shares of the bandwidth. These concepts are well-suited for industrial settings with well understood communication patterns where the communication paths are encapsulated at the factory sites. In the Industrial Internet of Things (IIoT), however, more and more parts of the industrial production domain are interconnected. This increases the complexity of the industrial

networks, makes them more dynamic, and creates more diverse sets of requirements. Furthermore, process control is imagined to be exercised from remote clouds for feasibility reasons which is why solutions on the link layer alone are not sufficient in these scenarios.

Common components of the IIoT can be divided into three categories as illustrated in Figure 1. Following [I-D.mcbride-edge-data-discovery-overview], EDGE DEVICES, such as sensors and actuators, constitute the boundary between the physical and digital world. They communicate the current state of the physical world to the digital world by transmitting sensor data or let the digital world interact with the physical world by executing actions after receiving (simple) control information. The processing of the sensor data and the creation of the control information is done on COMPUTING DEVICES. They range from small-powered controllers close to the EDGE DEVICES, to more powerful edge or remote clouds in larger distances. The connection between the EDGE and COMPUTING DEVICES is established by NETWORKING DEVICES. In the industrial domain, they range from standard devices, e.g., typical Ethernet switches, which can interconnect all Ethernet-capable hosts, to proprietary equipment with proprietary protocols only supporting hosts of specific vendors.

The challenge is to develop concepts that can include off-premise entities (such as distant cloud platforms) as well as proprietary

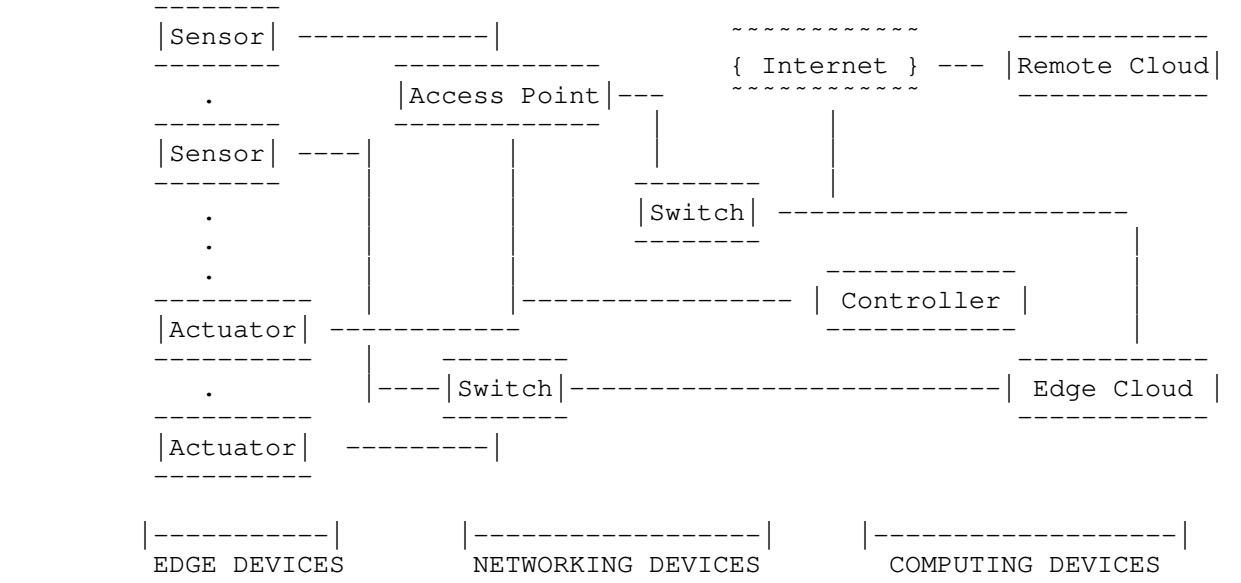

Figure 1: Industrial networks show a high level of heterogeneity.

hosts into the communication and still satisfy the performance requirements of modern industrial networks. The in-network computing paradigm presents a promising starting point because (pre-)processing data within the network can speed up the communication, e.g., by reducing the amount of transmitted data and thus congestion. Flexibly distributing the computation tasks across the network helps to manage dynamic changes. Specifying general requirements for the different application scenarios is difficult due to the mentioned diversity. This draft characterizes and analyzes three distinct scenarios to showcase potential requirements for the industrial production domain and to illustrate how in-network computations can be helpful.

2. In-Network Control / Time-sensitive applications

The control of physical processes and components of a production line is essential for the growing automation of production and ideally allows for a consistent quality level. Traditionally, the control has been exercised by control software running on programmable logic controllers (PLCs) located directly next to the controlled process or component. This approach is best-suited for settings with a simple model that is focussed on a single or few controlled components.

Modern production lines and shop floors are characterized by an increasing amount of involved devices and sensors, a growing level of dependency between the different components, and more complex control models. A centralized control is desirable to manage the large amount of available information which often has to be pre-processed or aggregated with other information before it can be used. PLCs are not designed for this array of tasks and computations could theoretically be moved to more powerful devices. These devices are no longer close to the controlled objects and induce additional latency.

It is worthwhile to investigate whether the outsourcing of control functionality to distant computation platforms is viable because these platforms have a high level of flexibility and scalability. In the following, we describe the requirements and characteristics of the control setting in more detail.

2.1. Characterization and Requirements

A control process consists of two main components as illustrated in Figure 2: a system under control and a controller. In feedback control, the current state of the system is monitored, e.g., using sensors and the controller influences the system based on the difference between the current and the reference state to keep it close to this reference state.

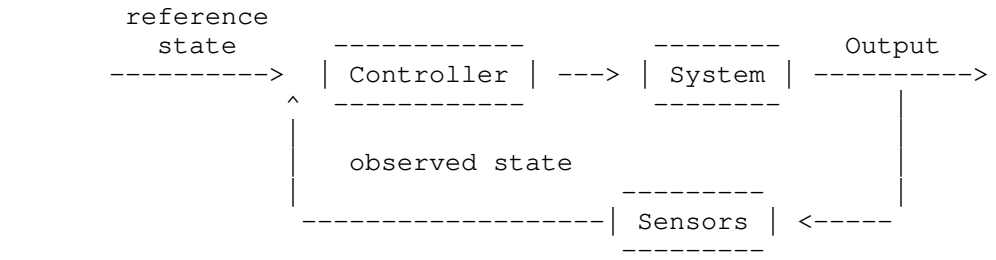

Figure 2: Simple feedback control model.

Apart from the control model, the quality of the control primarily depends on the timely reception of the sensor feedback, because the controller can only react if it is notified of changes in the system state. Depending on the dynamics of the controlled system, the control can be subject to tight latency constraints, often in the single-digit millisecond range. While low latencies are essential, there is an even greater need for stable and deterministic levels of latency, because controllers can generally cope with different levels of latency, if they are designed for them, but they are significantly challenged by dynamically changing or unstable latencies. The unpredictable latency of the Internet exemplifies this problem if off-premise cloud platforms are included.

The main requirements for the industrial control scenario are low and stable latencies to ensure that processes can work continuously and that no machines are damaged.

2.1.1. Approaches

Control models, in general, can become involved but there is a variety of control algorithms that are composed of simple computations such as matrix multiplication. As these are supported by programmable network devices, it is possibile to compose simplified approximations of the more complex algorithms and deploy them in the network. While the simplified versions induce a more inaccurate control, they allow for a quicker response and might be sufficient to operate a basic tight control loop while the overall control can still be exercised from the cloud. The problem, however, is that networking devices typically only allow for integer precision computation while floating-point precision is needed by most control algorithms. Early approaches like [RUETH] have already shown the general applicability of such ideas, but there are still a lot of open research questions not limited to the following:

- How can one derive the simplified versions of the overall controller?

    * How complex can they become?

    * How can one take the limited computational precision of networking devices into account when making them?

  o How does one distribute the simplified versions in the network?

  o How does the overall controller interact with the simplified versions?

3. Large Volume Applications/ Traffic Filtering

In the IIoT, processes and machines can be monitored more effectively resulting in more available information. This data can be used to deploy machine learning (ML) techniques and consequently help to find previously unknown correlations between different components of the production which in turn helps to improve the overall production system. Newly gained knowledge can be shared between different sites of the same company or even between different companies.

Traditional company infrastructure is neither equipped for the management and storage of such large amounts of data nor for the computationally expensive training of ML approaches. Similar to the considerations in Section 2, off-premise cloud platforms offer cost-effective solutions with a high degree of flexibility and scalability. While the unpredictable latency of the Internet is only a subordinate problem for this use case, moving all data to off-premise locations primarily poses infrastructural challenges which are presented in more detail in the following.

3.1. Characterization and Requirements

Processes in the industrial domain are monitored by distributed sensors which range from simple binary (e.g., light barriers) to sophisticated sensors measuring the system with varying degrees of resolution. Sensors can further serve different purposes, as some might be used for time-critical process control while others are only used as redundant fallback platforms. Overall, there is a high level of heterogeneity which makes managing the sensor output a challenging task.

Depending on the deployed sensors and the complexity of the observed system, the resulting overall data volume can easily be in the range of several Gbit/s [GLEBKE]. Using off-premise clouds for managing the data requires uploading or streaming the growing volume of sensor data using the companies' Internet access which is typically limited to a few hundred of Mbit/s. While large networking companies can simply upgrade their infrastructure, most industrial companies rely

on traditional ISPs for their Internet access. Higher access speeds are hence tied to higher costs and, above all, subject to the supply of the ISPs and consequently not always available. A major challenge is thus to devise a methodology that is able to handle such amounts of data over limited access links.

Another aspect is that business data leaving the premise and control of the company further comes with security concerns, as sensitive information or valuable business secrets might be contained in it. Typical security measures such as encrypting the data make in-network computing techniques hardly applicable as they typically work on unencrypted data. Adding security to in-network computing approaches, either by adding functionality for handling encrypted data or devising general security measures, is thus an auspicious field for research which we describe in more detail in Section 5.

3.2. Approaches

There are at least two concepts which might be suitable for reducing the amount of transmitted data in a meaningful way:

1. filtering out redundant or unnecessary data

2. aggregating data by applying pre-processing steps within the network

Both concepts require detailed knowledge about the monitoring infrastructure at the factories and the purpose of the transmitted data.

3.2.1. Traffic Filters

Sensors are often set up redundantly, i.e., part of the collected data might also be redundant. Moreover, they are often hard to configure or not configurable at all which is why their resolution or sampling frequency is often larger than required. Consequently, it is likely that more data is transmitted than is needed or desired. A trivial idea for reducing the amount of data is thus to filter out redundant or undesired data before it leaves the premise using simple traffic filters that are deployed in the on-premise network. There are different approaches to how this topic can be tackled. A first step would be to scale down the available sensor data to the data rate that is needed. For example, if a sensor transmits with a frequency of 5 kHz, but the control entity only needs 1 kHz, only every fifth packet containing sensor data is let through. Alternatively, sensor data could be filtered down to a lower frequency while the sensor value is in an uninteresting range, but let through with higher resolution once the sensor value range

becomes interesting. It is important that end-hosts are informed about the filtering so that they can distinguish between data loss and data filtered out on purpose.

In this context, the following research questions can be of interest:

- How can traffic filters be designed?
- How can traffic filters be coordinated and deployed?
- How can traffic filters be changed dynamically?
- How can traffic filtering be signaled to the end-hosts?

3.2.2. In-Network (Pre-)Processing

There are manifold computations that can be performed on the sensor data in the cloud. Some of them are very complex or need the complete sensor data during the computation, but there are also simpler operations which can be done on subsets of the overall dataset or earlier on the communication path as soon as all data is available. One example is finding the maximum of all sensor values which can either be done iteratively on each intermediate hop or at the first hop, where all data is available.

Using expert knowledge about the exact computation steps and the concrete transmission path of the sensor data, simple computation steps can be deployed in the on-premise network to reduce the overall data volume and potentially speed up the processing time in the cloud.

Related work has already shown that in-network aggregation can help to improve the performance of distributed ML applications [SAPIO]. Investigating the applicability of stream data processing techniques to programmable networking devices is also interesting, because sensor data is usually streamed. In this context, the following research questions can be of interest:

- Which (pre-)processing steps can be deployed in the network?
  - How complex can they become?
- How can applications incorporate the (pre-)processing steps?
- How can the programming of the techniques be streamlined?

4. Industrial Safety (Dead Man's Switch)

Despite increasing automation in production processes, human workers are still often necessary. Consequently, safety measures have a high priority to ensure that no human life is endangered. In traditional factories, the regions of contact between humans and machines are well-defined and interactions are simple. Simple safety measures like emergency switches at the working positions are enough to provide a decent level of safety.

Modern factories are characterized by increasingly dynamic and complex environments with new interaction scenarios between humans and robots. Robots can either directly assist humans or perform tasks autonomously. The intersect between the human working area and the robots grows and it is harder for human workers to fully observe the complete environment.

Additional safety measures are essential to prevent accidents and support humans in observing the environment. The increased availability of sensor data and the detailed monitoring of the factories can help to build additional safety measures if the corresponding data is collected early at the correct position.

4.1. Characterization and Requirements

Industrial safety measures are typically hardware solutions because they have to pass rigorous testing before they are certified and deployment-ready. Standard measures include safety switches, which need to be triggered manually, and light barriers. Additionally, the working area can be explicitly divided into 'contact' and 'safe' areas, indicating when workers have to watch out for interactions with machinery.

These measures are static solutions, potentially relying on specialized hardware, and are challenged by the increased dynamics of modern factories where the factory configuration can be changed on demand. Software solutions offer higher flexibility as they can dynamically respect new information gathered by the sensor systems. Depending on the corresponding occupational safety laws, the software has to satisfy stringent requirements which cannot be satisfied by regular best-effort networks.

4.1.1. Approaches

Software-based solutions can take advantage of the large amount of available sensor data. Different safety indicators within the production hall can be combined within the network so that programmable networking devices can give early responses if a

potential safety breach is detected. A rather simple possibility could be to track the positions of human workers and robots. Whenever a robot gets too close to a human in a non-working area or if a human enters a defined safety zone, robots are stopped to prevent injuries. More advanced concepts could also include image data or combine arbitrary sensor data.

In this context, the following research questions can be of interest:

- How can the software give guaranteed safety over best-effort networks?
- Which sensor information can be combined and how?

5. Security Considerations

Current in-network computing approaches typically work on unencrypted plain text data because today's networking devices usually do not have crypto capabilities. As is already mentioned in Section 3.1, this above all poses problems when business data, potentially containing business secrets, is streamed into remote computing facilities and consequently leaves the control of the company. Insecure on-premise communication within the company and on the shop-floor is also a problem as machines could be intruded from the outside. It is thus crucial to deploy security and authentication functionality on on-premise and outgoing communication although this might interfere with in-network computing approaches. Ways to implement and combine security measures with in-network computing are described in more detail in [I-D.fink-coin-sec-priv].

6. IANA Considerations

N/A

7. Conclusion

In-network computing concepts have the potential to improve industrial applications. There are at least three scenarios for which in-network processing can be beneficial, each having a unique set of requirements.

In the control scenario, tight latency constraints in the single digit millisecond range have to be satisfied despite the use of cloud platforms and the corresponding unstable latency of the Internet.

In a second scenario, large amounts of data have to be transmitted to cloud platforms for further evaluation. One important task here is to reduce the amount of data that needs to be transmitted as the

available Internet access speed is most likely non-sufficient. Apart from that, security measures have to be implemented as business data is transmitted to the Internet.

Regarding safety, software-based measures often lack the required guarantees and do not withstand the testing for certification. In-network processing with its potential for early responses can be a solution by combining different sensor outputs early and acting quickly.

8. Informative References

[GLEBKE] Glebke, R., "A Case for Integrated Data Processing in Large-Scale Cyber-Physical Systems", DOI: 10125/60162, in HICSS, January 2019.

[I-D.fink-coin-sec-priv] Fink, I. and K. Wehrle, "Enhancing Security and Privacy with In-Network Computing", draft-fink-coin-sec-priv-00 (work in progress), March 2020.

[I-D.mcbride-edge-data-discovery-overview] McBride, M., Kutscher, D., Schooler, E., and C. Bernardos, "Edge Data Discovery for COIN", draft-mcbride-edge-data-discovery-overview-03 (work in progress), January 2020.

[RUETH] Rueth, J., "Towards In-Network Industrial Feedback Control", DOI: 10.1145/3229591.3229592, in ACM SIGCOMM NetCompute, August 2018.

[SAPIO] Sapio, A., "Scaling Distributed Machine Learning with In-Network Aggregation", 2019, <https://arxiv.org/abs/1903.06701>.

[TSN] "Time-Sensitive Networking (TSN) Task Group", 2019, <https://1.ieee802.org/tsn/>.

Authors' Addresses

Ike Kunze
RWTH Aachen University
Ahornstr. 55
Aachen D-50274
Germany

Phone: +49-241-80-21422
Email: kunze@comsys.rwth-aachen.de

Klaus Wehrle
RWTH Aachen University
Ahornstr. 55
Aachen D-50274
Germany

Phone: +49-241-80-21401
Email: wehrle@comsys.rwth-aachen.de