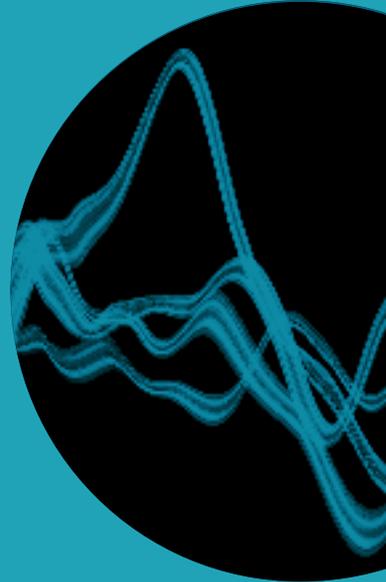
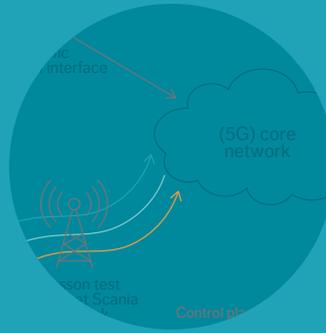
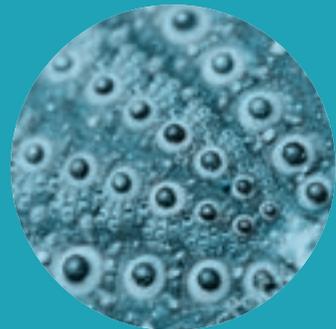


# Review

ERICSSON  
TECHNOLOGY



## 5G NETWORK PROGRAMMABILITY



# 5G network programmability

## FOR MISSION-CRITICAL APPLICATIONS

5G will make it possible for mobile network operators to support enterprises in a wide range of industry segments by providing cellular connectivity to mission-critical applications. The ability to expose policy control to enterprise verticals will create new business opportunities for mobile network operators by enabling a new value chain through the integration of telecom with other industries.

---

RAFIA INAM,  
ATHANASIOS  
KARAPANTELAKIS,  
LEONID MOKRUSHIN,  
ELENA FERSMAN

---

**A key differentiator of 5G systems from previous generations will be a higher degree of programmability. Instead of a one-size-fits-all mobile broadband service, 5G will provide the flexibility to tailor QoS to connectivity services to meet the demands of enterprise customers. This enables a new range of mission-critical use cases, such as those involving connected cars, manufacturing robots, remote surgery equipment, precision agriculture equipment, and so on.**

■ Network programmability can support rapid deployment of new use cases by combining cloud-based services with mobile network infrastructure and taking advantage of new levels of flexibility. Further, network programmability will enable a greater number of enterprise customers to use such services, and consumers will benefit from

a unique and personalized experience.

A number of use cases in mission-critical scenarios can benefit from QoS programmability because a cellular network's connectivity requirements – including latency, throughput, service lifetime and cost – vary widely across different use cases. To support them all, we have developed an application programming interface (API) that allows third parties to specify and request network QoS. We have also demonstrated the usefulness of this API on a test mobile network using a transport-related use case.

As part of this use case, we have been collaborating with commercial vehicle manufacturer Scania to develop the QoS requirements for teleoperation. Teleoperation is the remote operation of an autonomous vehicle by a human operator in cases where the vehicle encounters a situation that the autonomous system cannot overcome by itself (a road obstacle or malfunction, for example).

### Drivers of network programmability

The key drivers behind the creation of a programmable network are the need to accelerate time to market, and the desire to reduce operational costs and take advantage of the business opportunities presented by a new mission-critical service market. In a programmable network, traditional network functions requiring specialized hardware are replaced with software functions hosted on commercial off-the-shelf infrastructure. Technologies such as software-defined networking and network functions virtualization are essential to cutting operational and capital costs in mobile networks.

Cloud-based services and applications are enablers for programmability. Service provisioning in the cloud and managed access to the provisioned services and applications are important. This requires collaboration between telecom and other industries (IT application and content providers, and automotive original equipment manufacturers, for example). One way to simplify and accelerate the deployment of services and applications from industry verticals is the automatic translation of industrial requirements to service requirements, and then on to resource-level requirements (in other words, network requirements). Network slicing provides a dedicated, virtualized mobile network containing a set of network resources, and provides guaranteed QoS. The network slices are not only beneficial but also critical to support many applications in vertical industries.

New network communication services can also be provisioned programmatically; that is,

by using a software service orchestration function instead of manual provisioning by engineers.

As orchestration will also be used for provisioning connectivity services to mission-critical applications, mobile networks need to support QoS programmability.

### Mission-critical Intelligent Transportation System use cases

5G will support a diverse range of use cases in different industry sectors, each putting its own QoS requirements on the mobile network [1]. It is possible to use network programmability to realize mission-critical use cases with QoS requirements by creating highly specialized services tailored to industrial needs and preferences.

Features like lower latency (reaction times that are five times faster), higher throughput (10 to 100 times higher data rate) and an enormous increase in the number of connected devices (10 to 100 times more) can support the large-scale use of massive machine-type communication (mMTC) and mission-critical MTC (MC-MTC) use cases for the first time. Further, a dedicated network slice would meet the specific requirements of each use case.

In mMTC use cases, a large number of sensors and actuators are connected using a short-range radio (capillary network) to a base station (eNodeB) using a low protocol overhead to save the battery life of the devices.

This requires a network slice with broad coverage, small data volumes from massive numbers of devices. MC-MTC use cases emphasize lower latency (down to a level of milliseconds),

#### Terms and abbreviations

**AAR** – Authentication Authorization Request | **AF** – application function | **API** – application programming interface  
**eNB** – eNodeB | **EPC** – Evolved Packet Core | **EPS** – Evolved Packet System | **HSS** – Home Subscriber Server |  
**ITS** – Intelligent Transportation System | **MME** – Mobility Management Entity | **mMTC** – massive machine-type  
communication | **MTC** – machine-type communication | **PCRF** – policy and charging rules function | **PDN** – packet  
data network | **PGW** – PDN gateway | **QCI** – QoS class identifier | **Rx** – radio receiver | **SAPC** – Service-Aware  
Policy Controller | **SGW** – service gateway | **UDP** – User Datagram Protocol | **UE** – user equipment

## WE ENVISION REALIZING THESE USE CASES WITH A FLEXIBLE NETWORK PROGRAMMABILITY TECHNIQUE

robust transmission and multilevel diversity due to their mission-critical nature and, consequently, need a network slice of very low latency, high reliability and availability (packet loss down to 10<sup>-9</sup>). This is possible by creating a slice of very high priority. We envision realizing these use cases with a flexible network programmability technique.

The current focus of our research is within the Intelligent Transportation System (ITS) domain and includes a few 5G use cases in mMTC and MC-MTC, including transportation and logistics, autonomous cars and teleoperated vehicles.

### Transportation and logistics

The lower latency and high throughput of 5G will support multiple use cases related to connected cars, transportation and retail logistics that consist of fleets of connected/driverless vehicles transporting people and goods. The key network requirements for mission-critical automotive driving are high throughput and low latency up to 100ms. Failure is not an option in these cases. There are also many potential sub-use cases. For example, a journey from A to B in a driverless vehicle could involve vehicle-to-vehicle connections, connections between vehicles and street infrastructure for traffic management, and high-speed reliable connectivity to support cloud applications.

### Autonomous vehicles

The vision of fully autonomous vehicles aims to reduce the risks associated with human error. A system to achieve this vision would need to connect the cars and the road infrastructure with 1ms latency in all areas (100 percent coverage). Unfortunately, 1ms latency is currently not possible

in mobile networks. However, the bandwidth requirements to make this possible are not excessive, as only vehicle control data needs to be communicated. This capability is expected in 5G.

### Teleoperation of vehicles

The ability to control a self-driving vehicle from a distance is an important use case that is needed in public transportation when an onboard, autonomous system faces a difficult situation, such as a traffic accident, an unexpected demonstration, unscheduled roadworks or flooding. These scenarios require the planning of an alternate route, and an operator needs to drive the vehicle remotely for a short time. Another case could be a mechanical malfunction or an injury on a bus that requires remote intervention to mitigate the risk of danger to others. Network requirements for remote monitoring and control include broad coverage, high data throughput and low latency to enable continuous video streaming and the ability to send commands between a remote operations center and a vehicle [2].

### Why an API?

To guarantee QoS for the three ITS cases described above (and mission-critical use cases in general) mobile network operators typically go through manual network planning and configurations. Examples include configuring manual data routes via different routers, configuring Differentiated Services and allocating dedicated spectrum ranges to each use case. However, doing this is costly because it requires the configuration and deployment of network equipment. Nor is it particularly feasible, as this kind of configuration deployment cannot be done merely in parts of the transport network (such as backhaul). If, on the other hand, the resources were virtualized and there was software that could set up these routes over the same physical network link, both of the limiting factors would be eliminated: the cost of configuration and the deployment of multiple routes. As a result, it would be both financially and technically feasible to support these use cases concurrently.

It is clear that operators will not be able to

support the volume and diversity of use cases with the current network management approach. They need a different means of managing the network to stay competitive. In our view, developing an API is the logical first step toward exposing a programmable network to the industry verticals. This approach will result in a solution that is more responsive than rigid commercial offerings, such as preconfigured subscription packages.

### Architecture of the Ericsson-Scania project

Teleoperating a bus requires data from sensors on the bus, including a video feed from a camera at the front that is streamed to a remote operations center over LTE radio access with an evolved 5G core network. The commands to drive the bus are sent from the center to the bus using Scania's command system.

*Figure 1* illustrates the data streams that need to be prioritized to meet QoS demands: sensor data, the video feed originating from the vehicle user equipment (UE) and the commands to remotely drive the bus. Sending these data streams over low-priority data traffic (like infotainment) is a critical requirement. We used QoS class identifier (QCI) bearers, as detailed in the corresponding 3GPP standard, to enforce this prioritization. We assigned QCI class 5 and 2 to video and sensor data respectively and lowest-priority QCI class 9 to infotainment. In our lab environment, we have confirmed that the high-priority streams (QCI 2 and 5) can be kept regardless of the amount of low-priority background data traffic in the network [3, 4]. The next step will be to test our testbed setup for the prioritized video stream in the presence of the network load due to infotainment-type background traffic.

### How it works

A cloud-hosted application function (AF) dynamically sets up virtual connections between vehicles and the 5G Evolved Packet Core (EPC) network, with specific QoS attributes, such as designated latency levels and guaranteed throughput.

*Figure 2* illustrates the architecture of the system on which we have implemented the API. In addition to deploying a standard EPC and LTE band-40 RAN, an AF is deployed on an OpenStack-managed cloud. This application functionality allows third parties to set up QoS for their UEs through an API.

The AF consists of the following components: a knowledge base module, an API endpoint module and a transformer.

### Knowledge base module

This module maps domain-specific concepts to the generic concepts. The knowledge base is implemented as a graph database, has a schema of general concepts and can be extended with additional domain concept documents that instantiate the general concept schema. The schema includes a basic vocabulary of general concepts that model QoS requests. These concepts can be instantiated in domain concepts for a specific enterprise. In our case, the enterprise is automotive.

Within the knowledge base module, an "agent" is a string that is semantically related to the mobile device for which QoS is requested. In our case, the agent is instantiated with the "vehicle" domain concept. QoS class identifiers (QCIs) are indicators of network QoS for a given agent. The QCI concept was introduced in 3GPP TS 23.203 Release 8, with additional classes being introduced in Release 12 and Release 14.

Every QCI class has an integer identifier, for example QCI1 or QCI2, and is mapped to a set of QoS metrics such as an indicator of priority of data traffic, an upper ceiling for network latency and, in some cases, guaranteed bit rate. In our case, QCIs are instantiated with domain concepts for real-time vehicle traffic domain concepts. For example, QCI3 is instantiated as "vehicle\_control\_traffic" and QCI4 is instantiated as "vehicle\_video\_traffic." For users browsing their mobile devices in the vehicles, we instantiate a low-priority class QCI9 as "vehicle\_web\_browsing."

Data traffic descriptors are generic contents that can configure each QCI. The configuration

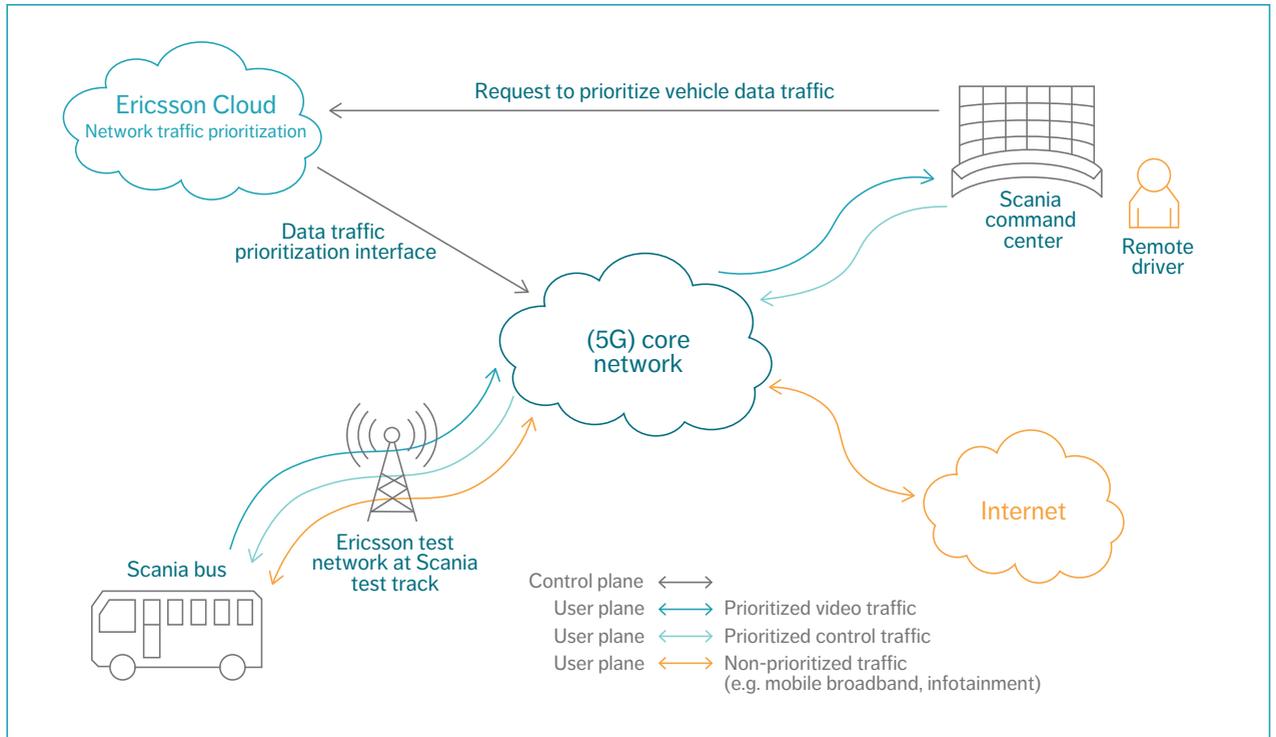


Figure 1 Prioritized data streams to meet QoS demands

pertains to a characterization of the traffic in terms of required throughput for both “uplink” and “downlink,” the former being data traffic transmitted from the agent and the latter being the opposite. Optionally, descriptors may also contain the type of data packets exchanged (for example, UDP/IP or TCP/IP), as well as potentially the port or port range. For example, in the case of “vehicle\_control\_traffic,” the data traffic descriptor identifies an uplink bandwidth of 1Mbps and a downlink bandwidth of 1Kbps.

A combination of agents, QCIs and their associated data traffic descriptors are stored in the knowledge base as a domain concept document. Every use case has its own domain

concept document, while each specific enterprise has more than one document. For example, in our case, there is an “automotive/teleoperation” document. However, other documents for automotive can also exist, such as “automotive/autonomous drive” or “automotive/remote fleet management.” Because the data is stored as linked data, concepts from one domain concept document can be reused in another.

#### API endpoint module

This module composes API specifications from every domain concept document in the knowledge base. This API specification is RESTful, uses symmetric encryption (HTTPS) and can be called

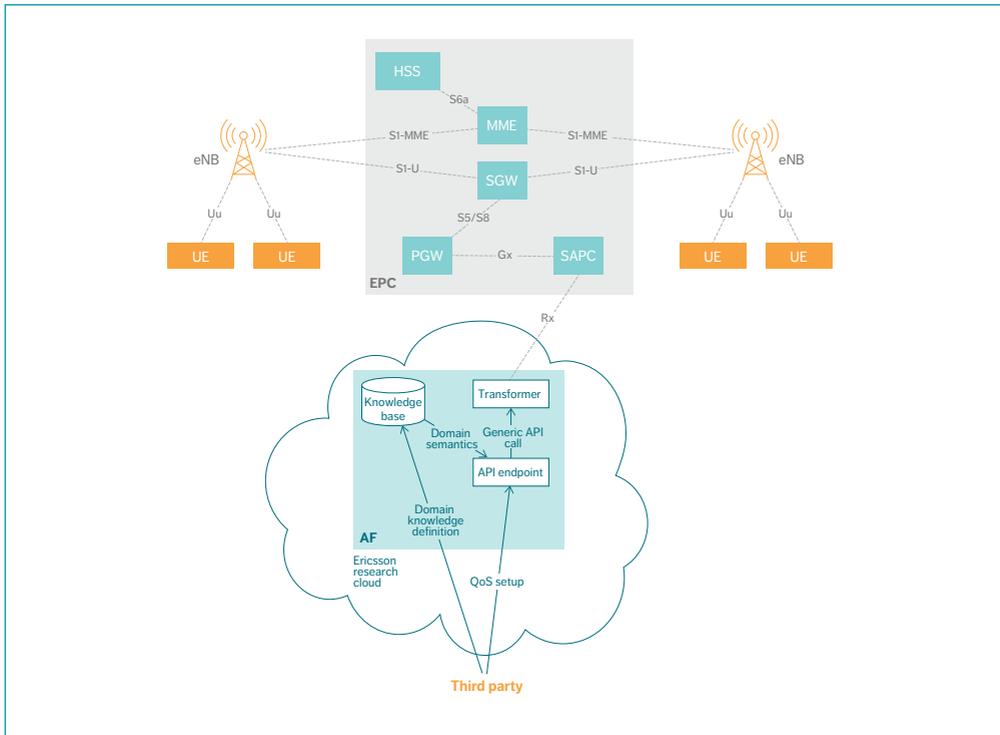


Figure 2 Architecture for programmable QoS in existing an LTE EPC network

from any third party. These API calls get translated into generic concept calls that are subsequently sent to the transformer module. Note that, in addition to an API call for setup of specialized QoS, there is another API call for teardown of this QoS. For example, when a vehicle is decommissioned or does not need to be teleoperated, there can be a call to tear down the QoS tunnel, so network resources can be allocated to UEs in other vehicles or devices. [Figure 3](#) provides an overview of domain-specific and generic requests.

#### Transformer module

The transformer module translates generic requests for QoS to Rx AAR requests, as these requests are

specified in 3GPP TS 29.214. The Rx requests are sent directly to the PCRF node in order to set up the “EPS bearer” (in other words, the data tunnel with the requested QoS). As is the case with the endpoint module, the transformer module can also translate a teardown request to an Rx request to revert to the lowest-priority default bearer (in most cases, QCI9).

CONCEPTS FROM  
ONE DOMAIN CONCEPT  
DOCUMENT CAN BE REUSED  
IN ANOTHER

Domain specific request	Generic request	Description of the request
GET /vehicle	GET /agent	Retrieve QoS information for all UEs
GET /vehicle/<IP>	GET /agent/<IP>	Retrieve QoS information for one UE, based on its IP address
GET /qos	GET /qos	Retrieve QoS for all UEs
GET /qos/vehicle_control_traffic	GET /qos/QCI3	Retrieve all UEs with QCI3 bearer setup
POST /qos/ <pre>{   "source_IP":&lt;src_IP&gt;,   "source_port":&lt;src_port&gt;,   "destination_IP":&lt;dst_IP&gt;,   "destination_port":&lt;dst_port&gt;,   "qos_class": "vehicle_control_traffic",   "protocol": "TCP",   "type": "vehicle_video_stream" }</pre>	POST /qos <pre>{   "source_IP":&lt;src_IP&gt;,   "source_port":&lt;src_port&gt;,   "destination_IP":&lt;dst_IP&gt;,   "destination_port":&lt;dst_port&gt;,   "qos_class": "QCI3",   "protocol": "TCP",   "max-requested-bandwidth-UL": "1024",   "max-requested-bandwidth-DL": "100" }</pre>	Set QoS for UE with IP src_IP and port src_port toward destination with IP dst_IP and port dst_port. Protocol in this example is TCP but it can also be UDP.
DELETE /vehicle <pre>{   "source_IP":&lt;src_IP&gt;,   "source_port":&lt;src_port&gt;,   "destination_IP":&lt;dst_IP&gt;,   "destination_port":&lt;dst_port&gt;,   "protocol": "TCP" }</pre>	DELETE /agent <pre>{   "source_IP":&lt;src_IP&gt;,   "source_port":&lt;src_port&gt;,   "destination_IP":&lt;dst_IP&gt;,   "destination_port":&lt;dst_port&gt;,   "protocol": "TCP" }</pre>	Remove QoS for UE with given source and destination IP and port

Figure 3 Overview of requests

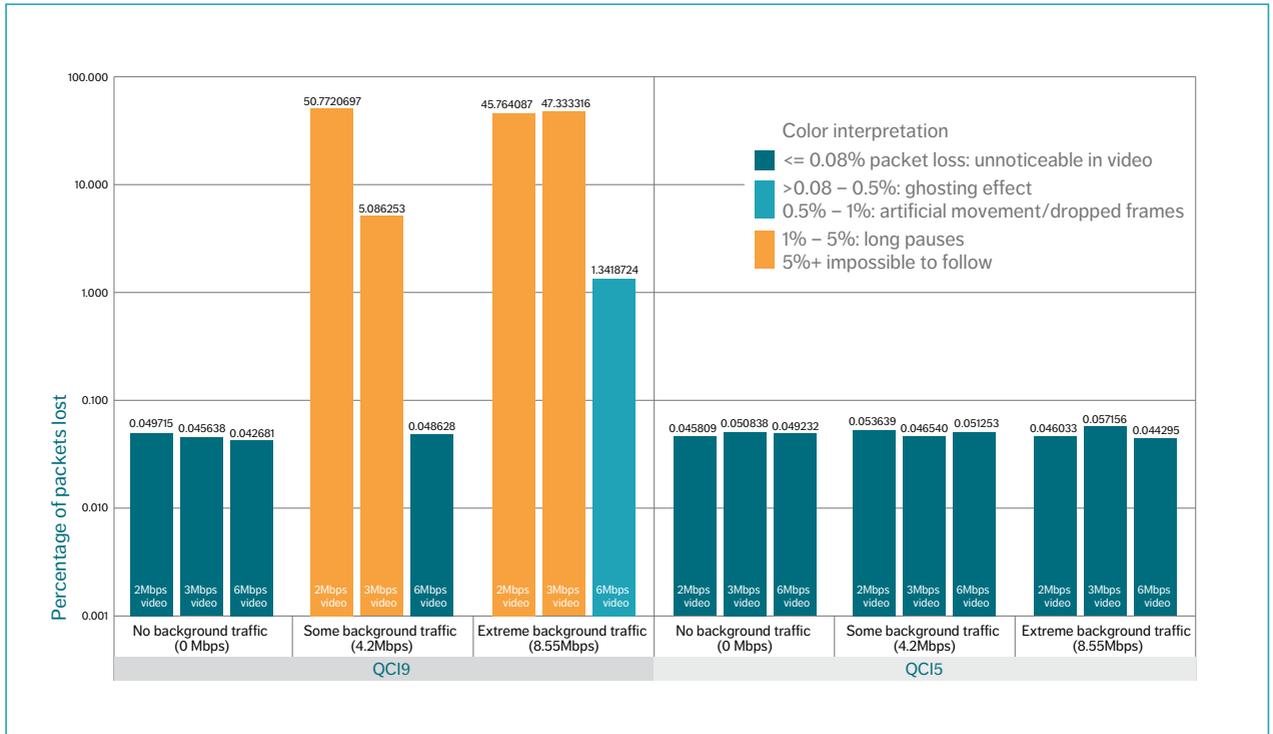


Figure 4 Packet loss (in percentage of total packets) in best effort (QCI9) and prioritized (QCI5) bearers

### Testbed results

To assess QoS, we performed experiments on the uplink prioritized video stream using QCI5 in the presence of the network load due to infotainment-type background traffic using QCI9. The total measured available bandwidth on the network was approximately 8.55Mbps. We tested several network load scenarios and measured the results against three background traffic conditions:

- » none (0Mbps)
- » some (4.2Mbps or 49 percent of the available bandwidth)
- » extreme (8.55Mbps or 100 percent of the available bandwidth)

We measured both throughput and one-way network delay under these traffic conditions. We also measured the ratio of packets lost versus packets sent to test the throughput quality of the network for three different qualities of video streams:

- » excellent (6Mbps or 70 percent of the available bandwidth)
- » good (3Mbps or 35 percent of the available bandwidth)
- » borderline drivable (2Mbps or 23 percent of the available bandwidth)

Borderline drivable is the minimum requirement to perform teleoperation. We obtained the packet

drop requirements from empirical observations during test driving. We took a total of 160 measurements for each experiment and plotted the graphs based on the respective average value.

Measurements from the 5G-network testbed show that resource prioritization can assure predefined QoS levels for mission-critical applications, regardless of background traffic. *Figure 4* illustrates guaranteed uplink packet loss for a critical application, in which the acceptable packet loss of less than or equal to 0.08 percent is unnoticeable in the video stream. This is true even with extreme background traffic when the system is congested – the critical traffic is still served with no performance degradation.

However, as *Figure 4* also illustrates, for the non-prioritized infotainment traffic (QCI9) the packet loss increases heavily with the increase in the background traffic, introducing long pauses and making teleoperation impossible even for a lower level of congestion.

When we measured the uplink delay, we found that it is preserved (remaining at less than 34ms) for the critical video traffic even when the system exhibits congestion. For the non-prioritized traffic, the delay reaches up to 600ms during congestion.

The next step is to develop the concept for a self-service portal where network customers could specify QoS requirements on their own terms; for example, to prioritize 4K video traffic for 40 buses in an urban scenario. The software would then translate this specification into instructions for network resource prioritization.

## Conclusion

5G attributes such as network slicing and low latency will soon make mission-critical use cases such as safe, autonomous public transport a reality. Automated network resource prioritization via a programmable API can support network QoS for diverse use cases with different connectivity requirements on the cellular network. By developing an API that allows a third party to request network resources and implementing it on a test mobile network, we have demonstrated how the technology works in an urban transport-related use case with Scania. The initial results show that throughput and latency are maintained for high-priority streams regardless of the network load. \*

## Further reading

- » **YouTube, Remote bus driving over 5G, November 2016**: <https://www.youtube.com/watch?v=IPyzGTD5FtM>
- » **Ericsson Research blog, 5G teleoperated vehicles for future public transport, June 8, 2017, Berggren, V; Fersman, E; Inam, R; Karapantelakis, A; Mokrushin, L; Schrammar, N; Vulgarakis, A; Wang, K**: <https://www.ericsson.com/research-blog/5g/5g-teleoperated-vehicles-future-public-transport/>
- » **Ericsson Mobility Report, June 2017**: <https://www.ericsson.com/assets/local/mobility-report/documents/2017/ericsson-mobility-report-june-2017.pdf>

**Rafia Inam**

◆ joined Ericsson Research in 2015. She works as a senior researcher in the area of machine intelligence and automation, and her research interests include 5G management, service modeling, virtualization of resources, reusability of real-time software and ITSs. Inam received her M.S. from Chalmers University of Technology in Gothenburg, Sweden, in 2010. She received her Licentiate and doctoral degrees from Mälardalen University in Västerås, Sweden in 2012



and 2014 respectively. Her paper, "Towards automated service-oriented lifecycle

management for 5G networks", won a best paper award in 2015.

**Athanasios Karapantelakis**

◆ joined Ericsson in 2007 and currently works as a master research engineer in the area of machine intelligence and automation. He holds a B.Sc. in computer science from the University of Crete in Greece and an M.Sc. and Licentiate of Engineering in communication systems from KTH Royal Institute of Technology in Stockholm, Sweden. His background is in software engineering.

**Leonid Mokrushin**

◆ is a senior specialist in the area of cognitive technologies. His current focus is on investigating what technological opportunities artificial intelligence may bring to Ericsson by creating and prototyping innovative concepts in the context of new industrial and telco use cases. He joined Ericsson Research in 2007 after postgraduate studies at Uppsala University in Sweden. He holds an M.Sc. in software engineering from Peter the Great St.



Petersburg Polytechnic University.

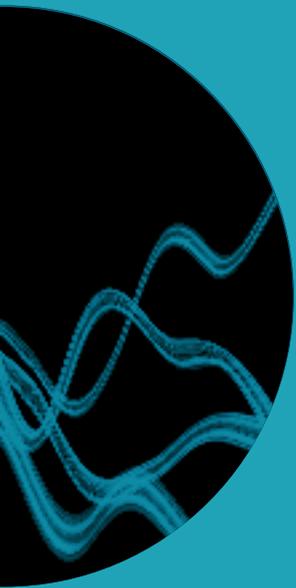
**Elena Fersman**

◆ is head of machine intelligence and automation at Ericsson Research and an adjunct professor in cyber-physical systems at KTH Royal Institute of Technology in Stockholm. She holds a Ph.D. in computer science from Uppsala University in Sweden and did post-doctoral research at École normale supérieure Paris-Saclay, France, before starting her industrial career. Her current research interests are in the areas of modeling and analysis and of software- and knowledge-intensive intelligent systems applied to 5G and IoT.

The authors would like to acknowledge the work of Keven (Q) Wang on this project during his stay at Ericsson.

**References**

1. IEEE, International Conference on Intelligent Transportation Systems (November 2016) – Feasibility Assessment to Realise Vehicle Teleoperation using Cellular Networks – Rafia Inam, Nicolas Schrammar, Keven Wang, Athanasios Karapantelakis, Leonid Mokrushin, Aneta Vulgarakis Feljan and Elena Fersman, available at: <http://ieeexplore.ieee.org/document/7795920/>
2. IEEE, International Conference on Future Internet of Things and Cloud (August 2016) – DevOps for IoT Applications Using Cellular Networks and Cloud – Athanasios Karapantelakis, Hongxin Liang, Keven Wang, Konstantinos Vandikas, Rafia Inam, Elena Fersman, Ignacio Mulas-Viela, Nicolas Seyvet and Vasileios Giannokostas, available at: <http://ieeexplore.ieee.org/document/7575883/>
3. Ericsson Mobility Report, Improving Public Transport with 5G, November 2015, available at: <https://www.ericsson.com/res/docs/2015/mobility-report/emr-nov-2015-improving-public-transport-with-5g.pdf>
4. IEEE, Conference on Emerging Technologies and Factory Automation (September 2015) – Towards automated service-oriented lifecycle management for 5G networks (Best Paper) – Rafia Inam, Athanasios Karapantelakis, Konstantinos Vandikas, Leonid Mokrushin, Aneta Vulgarakis Feljan, and Elena Fersman, available at: <http://ieeexplore.ieee.org/document/7301660/>



ISSN 0014-0171  
284 23-3306 | Uen

© Ericsson AB 2018 Ericsson  
SE-164 83 Stockholm, Sweden  
Phone: +46 10 719 0000