An Intelligent User Plane to Support In-Network Computing in 6G Networks

Susanna Schwarzmann*, Riccardo Trivisonno *, Stanislav Lange †, Tugce Erkilic Civelek ‡,
Daniel Corujo §, Riccardo Guerzoni ‡, Thomas Zinner †, Toktam Mahmoodi ¶
* Huawei Technologies Munich, Germany
†NTNU Norway, Department of Information Security and Communication Technology
‡ DOCOMO Communications Laboratories Europe
§ University of Aveiro and Instituto de Telecomunicações
¶ Kings College London, Centre for Telecommunications Research

***Abstract*—Driven by the development of programmable networking hardware, In-network Computing (INC) has gained a considerable amount of attention in recent years. However, INC has so far barely been studied in the context of mobile networks, despite the vast advantages, such as latency or traffic reduction, shown for fixed networks. Motivated by an Augmented Reality (AR) use-case, our work envisions an INC-enabled Intelligent User Plane (IUP) for 6G networks, which allows offloading computational tasks to UP entities having enhanced computational capabilities. The 6G IUP thus helps to keep mobile end-devices lighter and supports meeting the stringent delay requirements of AR applications. Besides elaborating on the involved prospects and challenges, we identify key enablers for realizing the proposed INC-enabled IUP. We show that embedding INC into the 6G system entails major changes in the architecture, as compared to the current 5G design.**

***Index Terms*—In-Network Computing, Intelligent User Plane, 6G, Mobile Networks**

I. Introduction

In order to meet the stringent delay requirements of modern applications, Mobile Edge Computing (MEC) has evolved as a promising concept widely deployed nowadays. However, even having the computational resources close to the edge will not be sufficient for emerging applications such as interactive, high resolution Virtual Reality gaming, which require a throughput of up to 2.3 Gbps and can tolerate a latency of only 10 ms [1]. The next evolutionary step, to bring the computations even closer to the user, is to *compute in the network*. In the context of mobile networks, this could mean a leap from a pure communication system – as with 5G – to a coalesced *communication and compute system* with the design of 6G.

The idea of shifting computation to the network is not novel as such, but has been proposed as active networking [2] already decades ago. However, it is the recent advances in programmable network devices (PNDs) and the development of programming languages like P4, that make INC a practically usable concept today. A vast amount of proof of concepts show how to execute applications on the fly, or how to reduce network traffic and increase energy-efficiency using INC. Thereby, most works quantify performance improvements by means of an integration of INC in static scenarios, but barely consider the requirements and implications for networking eco-systems as a whole. In this respect, the work in [3] gives a more holistic picture of an INC embedding, taking into account architectural aspects, data management challenges, as well as possible impacts on applications. While the discussions clearly highlight the need for rethinking the way we design networks so to allow an INC integration, dedicated solution outlines are still scarce.

This paper aims towards closing this gap by elaborating on an embedding of the INC concept into 6G mobile networks. More specifically, we present the idea of an INC-enabled Intelligent User Plane (IUP), which allows offloading application-specific tasks to User Plane (UP) entities. In this context, we present key enablers for its realization and elaborate on the necessary enhancements with respect to the mobile network architecture. We show that integrating the INC concept involves major modifications, as compared to current 5G system design considerations, which can only be implemented at a generation shift towards 6G. Our key contributions can be summarized as follows: (1) We elaborate on the prospects of INC and unsolved challenges, with a specific focus on mobile networks; (2) By means of an AR-related use-case, we motivate the need for INC in the mobile context; (3) We describe our idea of the IUP along with its key enablers; (4) We provide at a first glance the necessary architectural enhancements, as compared to the current 3GPP architecture for 5G systems.

The remainder of the paper is organized as follows. Section II outlines related works, followed by a discussion on the INC propsects and challenges in Section III. In Section IV, we motivate the IUP and present the key enablers necessary for its realization. Section V focuses on the architectural enhancements as compared to 5G systems and Section VI concludes the paper.

II. Related Work

Recent research proposes the usage of INC for a multitude of different tasks. NetCache [4] describes an approach for in-network caching, leveraged by a packet-processing pipeline. DAIET [5] is an in-network solution for hand-written digit identification using Machine Learning (ML), distributed on several machines within a data center. The system detects overlapping information exchanged between the machines, allow-

ing to reduce the network load by aggregating such duplicate data. Similarly, the architecture proposed as SwitchAgg [6] performs traffic aggregation at line rate using a payload analyzer and multiple processing elements. Besides shifting such key networking tasks into the network, several works leverage the INC concept for offloading arbitrary application-level logics to the network. The authors of [7] explore to which extent computer vision functions can be offloaded to PNDs. More specifically, the authors present a proof of concept for performing the discrete convolution on an image recorded by a car, so to enable in-network edge-detection.

While all the above-mentioned approaches focus on INC using programmable network devices, [8] proposes an INC model for 6G, where Network Functions (NF) are integrated into a general computing platform, instead of delegating computing tasks to the network devices. That is, network nodes are equipped with powerful computing capabilities, being able to carry out application-specific computations on top of their typical forwarding tasks. The ultimate goal of INC in [8] is to reduce the amount of traffic processed at remote data centers, and thus increasing the energy-efficiency. Motivated by the stringent requirements of current trends and applications, such as industrial automation and AR, the work in [9] highlights the need for a shift towards more intelligence and openness with 6G. The authors argue that this can only be achieved by means of a convergence of communication, computing, and caching towards a service-aware 6G. In this respect, the work presents key enabling technologies, relating to spectrum management, radio channel construction, delay-aware transmission, wireless distributed computing, and network self-evolution.

III. INC - Prospects and Challenges

The following section first elaborates on the key prospects of INC, giving examples from existing works. Afterwards, we outline a set of involved challenges to be addressed.

A. Prospects

Reduction of traffic volume: Especially if a task involves large amounts of data, e.g., video streaming, offloading it to close network entities, instead of to a MEC server, reduces the overall traffic volume, and thus alleviates congestion. On the one hand, this is due to the reduced number of hops along the path. On the other hand, INC can reduce the traffic by analyzing the data and only forwarding those packets that are relevant to the application, while the others are aggregated or dropped. For example, the proposals in [5], [6] detect and prevent the transmission of duplicate information in data center networks, while [10] assesses the relevance of monitoring information using ML for an industrial fine blanking use-case.
Latency reduction: INC can reduce the latency due to the reduced overhead of multi-hop paths, resulting from the inclusion of intermediary compute servers. Instead of sending the data to a dedicated server, INC modifies the data packets from sender to receiver in the network node at line rate, as shown in [11] on the example of a coordinate transform task. Further, application-level latency can be reduced by an active and shortened response by the network entity. In [12], a switch analyzes the traversed packets to monitor the position of a robot which is collaborating with a human. If it detects a critical position, the switch can autonomously send an emergency stop signal to the robot, thus reducing the risk of human injuries. In [13], a programmable switch parses the packets using a match-action pipeline. If a DNS request is detected and the switch has the DNS entry cached, it can directly respond, thus drastically reducing the latency as compared to the queried server being the one to respond.
Support of simple end devices: INC supports the design of light-weight devices by allowing to offload their tasks to close network elements, as later described in Section IV-B.
Increased energy-efficiency: Offloading application tasks to the network can be beneficial in terms of energy-efficiency in general, not only for the (lightweight) end devices. For example, data centers can benefit from a reduced energy consumption by means of intelligent data processing along the transmission path, as shown in [8].

B. Challenges

Traffic encryption: Nowadays, most Internet traffic is encrypted, making INC impracticable outside of vertical networks, where the network is a non-trusted party. A possible solution to the problem can be homomorphic encryption (HE) [14], a technique allowing to compute on encrypted traffic. However, despite the currently ongoing vast research efforts towards making HE more efficient, e.g., by means of advanced algorithms and hardware acceleration [15], [16], HE is still too slow for practically usage.
Inter-operability with existing protocols: The usage of unreliable protocols, such as UDP, is limited for INC. If packets are dropped that contain computation results, the computation has been carried out with no purpose. On the other hand, using TCP, which adapts its sending rate according to the RTT, can lead to problems due to the RTT increase resulting from additional computation time. Further examination on how far existing protocols can support INC is needed, but it is expected that dedicated protocols will be necessary [17].
Trust issues and ensuring the correctness of the computation: It requires trust of the user and of the application provider, that the network carries out the computation as intended [18]. It further needs to be ensured that the conducted computations are correct. Thereby, verifiable computing [19] might be a concept to be exploited in conjunction with INC.
Interoperability with QoS: The UP entities in 5G are responsible for fulfilling a flow's QoS requirements while routing its traffic through the network. That is, e.g., to ensure a certain guaranteed bitrate or to keep a specific delay budget. When introducing INC to the 6G UP, it needs to be clarified how to ensure both, i.e., the QoS of the flow, as well as the correct computation. Ensuring QoS will be more challenging, due to the additional latency for computing on the flow's packets.
Disruption of the end-to-end principle: The end-to-end principle generally states that information pushed on the sending side of the connection should be received without modification

at the receiver side. Intermediary nodes only act as the connector between the explicitly addressed communication end points. This pure end-to-end communication might no longer be suitable with INC [18]. Besides payload modification with INC, several proposals [12], [13] allow the network entity to actively respond, although it is not directly addressed by the sender. The concept of having one dedicated sender and one dedicated receiver, as well as existing transport layer solutions will need to be re-considered [20].

UP path setup: The complexity of determining an appropriate UP path is increased when additional constraints – relating to the computations – come into play. That is, besides factors such as the location and link properties, we need to consider the available computational resources and which tasks are supported by the respective UP nodes. Solving the problem of an optimal embedding is not straightforward.

Dynamic and optimized computation allocation: Distributing computations among various instances (UE, MEC server, UP entities), requires sophisticated mechanisms to determine an optimized allocation of the compute tasks, given a current set of dynamic conditions. For example, in [21], a multi-criteria edge-computing-enabled live service migration procedure is optimized considering different types of migration costs and benefits.

User Mobility: An important aspect is to ensure that the computation is "moved" with the user. Depending on the magnitude of the UE's movement, a couple of entities need to be re-selected, e.g., Access Node (AN), User Plane Functions (UPFs), Access and Mobility Function (AMF), or Session Management Function (SMF). The re-selection becomes more difficult when computing on the flows, due to (1) the increased complexity of the UP path setup and (2) possible packet dependencies and states. That is, certain computations may require a batch of packets and re-selections need to be coordinated with the specific computation's requirements and current state.

Dependability: Finally, the proposed communication and compute system gives rise to challenges related to dependability. In contrast to current mechanisms that typically only consider static configurations, more sophisticated approaches will be necessary. Particularly, operational states will need to be taken into consideration when dealing with failures in order to maintain correct operation.

IV. Envisioned IUP for 6G Networks

In the following, we introduce our idea of the 6G IUP. Thereby, we motivate the concept by means of an AR use-case and describe its necessary enablers.

A. Definition of INC for the IUP

For the envisioned IUP, we refer to INC in a wider scope [22], i.e., considering different definitions of the concept. Firstly, as offloading arbitrary tasks to the network, similar as in [23]: *In-network computing refers to the execution of programs typically running on end hosts within network elements. It focuses on computing in the network, using devices that already exist within the network and are already used to forward the traffic*. Secondly, as processing packets at line speed, in a pipelined manner [24]: *Application-specific functions that can run in programmable network hardware at line rate, offering orders of magnitude higher throughput and lower latency than can be achieved by a traditional server*. Accordingly, the computations carried out in the IUP can range from very simple tasks, only requiring to keep small states, such as a deadband-based packet reduction [25], as well as more complex ones, such as ML-based object detection.

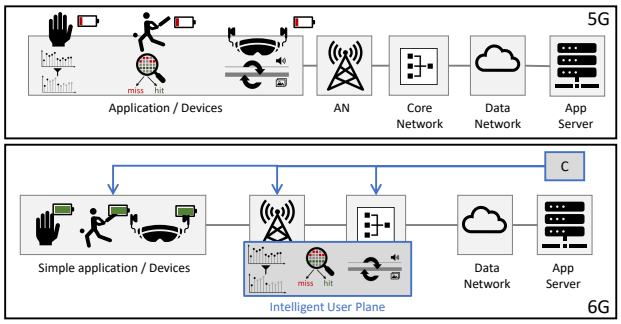

Fig. 1. Vision of the IUP for 6G, supporting lightweight devices through offloading application-specific tasks to the UP entities.

B. Motivation: Mobile AR Gaming

Along several research associations, INC is seen as a key enabler for meeting the requirements of future immersive media, such as VR and AR [26]. We also motivate the IUP by means of a mobile AR gaming use-case, showing that it can (i) simplify the mobile end-devices and (ii) reduce the latency as compared to, e.g., MEC-based solutions. The upper part of Figure 1 shows a scenario in the 5G system, where several devices are involved in a game. Via the AN and the core network (CN) UPF, the devices establish a connection to the Data Network (DN), and thus to the remote application server. We assume that each device has a dedicated connection to the AN, i.e., act as independent user equipment (UEs). Each device performs different game-relevant tasks: The glove sends haptic information captured by its sensors, the racket detects if the ball was hit or missed, and the AR glasses render video objects and detect objects in the real environment. The problem with these (complex) computations being carried out at the mobile devices is the clash with their vital design requirements and characteristics: No external power supply and high wearing comfort. To achieve a high wearing comfort, the devices should be equipped with small batteries and should not become to warm, even when wearing them for a long time. However, a high computational load can heat up the device and lead to fast drainage of the small batteries. Indeed, the issues of battery lifetime and overheating are still today challenging the development of AR glasses and have lead to delays in commercial releases of end-consumer grade equipment.

The lower part of Figure 1 depicts the same scenario with the envisioned 6G IUP. The devices can be kept simple, as tasks are offloaded to the 6G UP entities, i.e., AN and UPF. A controlling entity (C) programs end-devices and involved UP entities for INC usage and allocates the compute tasks among

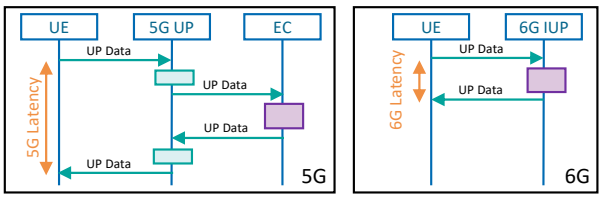

Fig. 2. Reduced latency with the 6G IUP as compared to a 5G solution leveraging edge computing.

them. This entity provides the required level of interaction between application, devices, and the 6G system. It can, e.g., be implemented by means of an Application Function (AF), a new Control Plane (CP) NF introduced with 5G systems.

Other concepts, like mobile code offloading, as used for example within the MobiCOP framework [27], have also shown a great potential for reducing mobile devices' energy consumption by migrating processor-intensive tasks to resource-rich surrogates. Such solutions, however, are prone to exceed the stringent delay requirements of novel use-cases, such as mobile AR gaming. Figure 2 highlights the need for bringing computations into the network, when aiming at both, energy-efficiency and low latency. The left part shows a 5G scenario, where the UP data is sent via the 5G UP (both AN and CN) to an edge computing (EC) server, where it is processed and sent back to the UE via the 5G UP. Having the proposed IUP in 6G networks allows to process the UP data directly within the mobile system, thus drastically reducing the latency as compared to the EC solution in a 5G system.

C. Envisioned Enablers

In the following, we describe four key enablers for realizing the envisioned IUP for 6G networks.

1) ***CUPE:*** *Computation-enabled User Plane Entity:* Compared to state-of-the-art UP entities (i.e. AN and UPF), dedicated to communication only, the Computation-enabled User Plane Entities (CUPEs) are significantly enhanced in terms of computational resources, i.e., CPU, GPU, RAM, and storage. We envision two realizations of a CUPE: Firstly, a general-purpose CUPE, potentially running on virtualized infrastructure, so as to scale it dynamically to the current needs, e.g., number of active flows or complexity of the conducted computations. This realization has the benefit that it can carry out any arbitrary tasks, but it is potentially slow as computations are done in software. Secondly, a specific-purpose CUPE, potentially with dedicated hardware-acceleration, with the key benefit of being able to process packets at line-rate, which however comes with the limitation of being applicable only for very particular tasks.

2) ***CS:*** *Compute Service:* With the term Compute Service, we refer to computations carried out in the CUPEs. That is, a CS unites all computational instructions and precisely describes the actions to be executed on a flow's packets. We envision two types of such compute services: (1) Predefined CS: Those represent generic computations, which can be useful for a wide range of applications. As they are foreseen to be frequently used, all – or at least a large set of – CUPEs would support these compute services off the shelf. (2) Customized CS: Those represent highly specific computations, e.g., well-tailored to the particular needs of a given application. While such customized CSs offer a high degree of flexibility to the application, they cannot be supported by a large set of CUPEs per se. The deployment of a customized CS at the CUPEs in charge, i.e., those connecting the application flow's end points, can be initiated via the AF. An application provider can communicate the desired CS, e.g., in the form of an execution script, to the 6G system's orchestration and management (OAM). If the request is accepted, the new CS can be deployed and the application's packets are processed in the network as specified by the application provider.

3) ***CC Flow:*** *Communication and Compute Flow:* We envision Communication and Compute Flows (CC Flows), that allow computation management in the 6G system on a per-flow level. They can be seen as an evolution of QoS Flows [28], an existing concept from 5G networks, allowing QoS management on a per-flow level. The key advancement introduced with CC Flows is that – besides treating the flows in the specified QoS-aware manner – the CUPEs carry out computations on CC Flows' packets, according to the compute service(s) associated to that flow. The packets' payload data may hence be modified along the way from sender to receiver. This is a key change compared to flows as known today.

4) ***CCCE:*** *Communication and Compute Control Entity:* This control entity is responsible for determining the appropriate CUPEs to use for a CC Flow, allocating the computations among the involved entities, and any further tasks related to the CC Flow setup. The CCCE is aware of the CSs supported by the different CUPEs, as well as their current load. The control entity is not necessarily a single entity, but can be realized in a distributed manner, such that logical tasks are distributed among different entities, i.e. CP NFs.

V. Architectural Enhancements

In comparison to current 5G systems, the envisioned IUP requires significant modifications to the mobile network UP and consequent updates of the CP, which can only be implemented at a generation shift towards 6G. Our key focus is on the novelties in the UP, while for the CP we will – for simplicity – assume that the general concepts (Service Based Architecture (SBA) in the CN) and functional splits of NFs will still hold with 6G. The goal is to show the general realization of the IUP and its enablers and to provide a comparison to existing procedures. The reference architecture is the 5G architecture as described by 3GPP for the access network [29] and the core network [28].

A. Architecture Baseline and 5G User Plane

Figure 3 shows an excerpt of the 3GPP architecture. We first refer to the 5G UP, illustrated as the gray box and focus on the CP and UP functions that are involved in setting up a communication flow. In a simplified view, the Access Node (AN) includes CU-CP (Centralized Unit Control Plane), which implements (i) the AN's CP functions and the interaction with

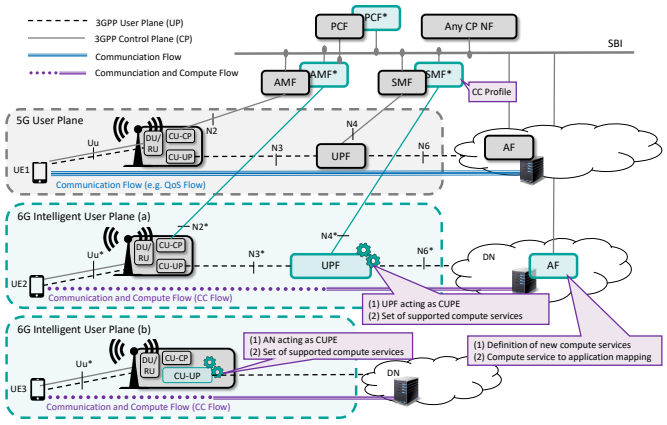

Fig. 3. 3GPP architecture excerpt: Pure 5G Communication Flows vs. envisioned 6G Communication and Compute Flows.

TABLE I
EVOLUTION OF CONCEPTS TOWARDS ENABLING 6G CC FLOWS

5G System		*6G System*	
QoS Flow	QoS-aware treatment of a flow's packets	CC Flow	Computations carried out on the flow's packets at CUPEs
QFI	Unique identifier of QoS Flow	CFI	Unique Identifier of CC Flow
5QI	Pointer to a set of QoS charact.	6CSI	Pointer to a (set of) CSs
QoS Profile	Describes the QoS parameters and specifications of the flow (e.g., guaranteed bitrate value)	CC Profile	Describes the CSs to use and further CS-specific parameters (e.g., thresholds)

the CN over the N2 interface, and (ii) CU-UP (Centralized Unit User Plane), which implements the AN's UP functions. The DU/RU (Distributed / Remote Unit) implements the upper and lower Physical Layer functions. For the CN CP, the relevant CP NFs interacting with each other via the SBI are: AMF, SMF, and PCF (Policy Control Function). The SMF interacts via the point-to-point N4 interface with UPF.

We briefly describe the setup of a 5G QoS Flow (for more detailed information, please refer to [30]), as we later explain the CC Flow setup in comparison to it, highlighting the necessary advancements. The QoS Flow setup procedure can either be initiated by (1) the UE (e.g., because an application in the UE needs to send traffic to an application server in a data network (DN)) or by (2) the 5G network. In both cases, the application layer can influence the QoS profile that the 5G system associates to the QoS Flow: an application server can act as an AF to request via SBI interaction with the 5G CN CP a certain QoS profile for application traffic to/from a specific UE. The AF's influence on the QoS profile can be performed when the UE has already established a Protocol Data Unit (PDU) session or before the establishment. The PCF considers the QoS requirements of the AF when determining the Policy and Charging (PCC) rules associated to a UE. Based on the PCC rules received from the PCF, the enforcement of the QoS profile is performed by the SMF, which coordinates the QoS Flow establishment among all involved UP entities, i.e., UE, AN, and UPF(s). Within the 5G system, this QoS framework allows to meet the application's QoS requirements. Yet, does it not allow to modify the flow's packets, thus, providing pure communication capability.

TABLE II
6G NF ENHANCEMENTS COMPARED TO EXISTING 5G NFS

NF	*Relevant tasks in 5G system*	*Additional tasks for CC Flow support*
AF	Influence the QoS profile assigned by the 5GS to application traffic to/from a UE	Initiate CC Flow Setup, define mapping between applications and CSs, request deployment of new, customized CSs
SMF	UPF selection considering UPF locations, link properties, supported DNs, Network Slices, and features like service chaining	Supported set of CSs, currently available computational resources / load
UPF/ AN	Guaranteeing a flow's QoS requirements, traffic forwarding	Compute on a flow's packets, additional reporting (e.g. computational load)

B. Architecture Impact for the 6G Intelligent User Plane

When describing the setup of CC Flows in the 6G IUP, we keep the naming of existing 5G CP NFs and interfaces, but indicate by means of the asterisk (*) that they are subject of change. At this stage of our research, the key focus is on the UP enhancements. While major modifications on the CP and their NFs may be necessary, we only refer to their key enhancements necessary to realize the proposed enablers in this work. Table I and Table II give a brief outline on the evolution of existing 5G concepts and the enhancements of existing 5G NFs, so as to enable the envisioned CC Flows for the IUP for 6G networks.

The turquoise parts in Figure 3 illustrate two versions for the 6G IUP, connecting UEs to the DN: Firstly, option (a), where the INC feature is available at the UPF (UPF acting as CUPE). Similar as with 5G QoS Flows, the AF can influence the setup of a CC Flow. However, offloading application tasks to the network requires a higher degree of interaction between application and network. The AF consequently needs additional capabilities. That is, defining the compute service(s) to be used for a set of UEs (e.g., depending on their computational capability or location) when running a specific application. Furthermore, the AF can send requests to the network, for the deployment of a new, customized CS to (a set of) UP entities.

Analogous to the enforcement of the QoS profile in 5G, the PCF* determines a CC Profile (either based on pre-configured PCC rules or application-initiated via AF influence), which is propagated to the SMF*. The SMF*, in turn, derives from the CC Profile all relevant information for the CUPEs, indicating how to treat the flow's packets. The CC Profile includes a 6G Compute Service Identifier (6CSI), pointing to the (set of) CS to use, analogues to the concept of 5G QoS identifier (5QI), which are pointers to a set of QoS characteristics. The CC Profile can contain additional CS-specific computational parameters, e.g. a just-noticeable difference threshold for deadband-based encoding techniques.

For determining the appropriate UP components in the envisioned communication and compute architecture for 6G, the SMF* needs to consider several additional factors, as compared to 5G systems. That is, the availability of compute resources and the set of supported CSs at the different CUPEs. After determining the UP path, the SMF* distributes the configuration derived from the CC Profile to all involved UP entities, i.e. UE, AN, CUPEs. The established CC Flow between DN and UE2 is uniquely identified by means of its CC Flow Identifier (CFI), analogues to the 5G QFI which

identifies QoS Flows. Please note that the packets of the CC Flow are modified by the CUPE. In the outlined scenario, AF, PCF*, and SMF* would – in a distributed manner - act as the Communication and Compute Control Entity (CCCE).

Secondly, option (b), where an INC-enabled AN (AN acting as CUPE) performs traffic offloading to a local access to the DN. This option would require – in addition to the INC functionality – merging/co-locating the AN with some networking functionalities implemented by the UPF in 5G (uplink/downlink QoS enforcement, downlink traffic notification, traffic forwarding, etc), to allow operating on higher, i.e. PDU, layer packets at the AN. This would mean a break with the functional split between access and core network – one of the key design principles of 5G systems [31] – but would enable capillary traffic offload to the DN, with enhancements in terms of utilization of the underlying transport network and latency reduction. Use cases that require high throughput and latency everywhere in wide areas would benefit from deployments based on option (b). We omit the CP and interfaces involvement for this implementation option, as the mapping of 5G NFs to the respective 6G counterparts is not as straightforward as in the previous case. A third scenario, where INC functionalities are realized by both AN and UPF, is also possible, but its usage requires further study.

VI. Conclusion

In-network computing is an evolving concept which brings a vast number of benefits to the networking domain. As shown in this work, it promises to be especially beneficial in the context of mobile networks, as it can help to meet the stringent delay requirements of emerging applications such as AR, and supports the design of lightweight end user equipment. In this scope, we presented our vision of an Intelligent User Plane, which embeds the INC concept in 6G networks. We elaborated on the key enablers and showed that their realization entails major modifications to the mobile networking architecture, as compared to the current 5G design. Our next steps will focus on performance evaluations to quantify the impact of the INC-enabled IUP. Furthermore, we plan to elaborate deeper on how the open challenges can be addressed.

Acknowledgment

This work is a joint contribution within Working Item 207 on "Intelligent User Plane and In-Network Computing" of the one6G association.

References

[1] S. Mangiante, G. Klas *et al.*, "Vr is on the edge: How to deliver 360 videos in mobile networks," in *Proceedings of the Workshop on Virtual Reality and Augmented Reality Network*, 2017, pp. 30–35.

[2] D. L. Tennenhouse, J. M. Smith *et al.*, "A survey of active network research," *IEEE communications Magazine*, vol. 35, no. 1.

[3] M.-J. Montpetit, "The network as a computer board: Architecture concepts for in-network computing in the 6G era," in *1st International Conference on 6G Networking*. IEEE, 2022, pp. 1–5.

[4] X. Jin, X. Li *et al.*, "NetCache: Balancing key-value stores with fast in-network caching," in *Proceedings of the 26th Symposium on Operating Systems Principles*, 2017, pp. 121–136.

[5] A. Sapio, I. Abdelaziz *et al.*, "In-network computation is a dumb idea whose time has come," in *Proceedings of the 16th ACM Workshop on Hot Topics in Networks*, 2017, pp. 150–156.

[6] F. Yang, Z. Wang *et al.*, "SwitchAgg: A further step towards in-network computation," *arXiv preprint arXiv:1904.04024*, 2019.

[7] R. Glebke, J. Krude *et al.*, "Towards executing computer vision functionality on programmable network devices," in *1st ACM CoNEXT Workshop on Emerging in-Network Computing Paradigms*, 2019, pp. 15–20.

[8] N. Hu, Z. Tian *et al.*, "An energy-efficient in-network computing paradigm for 6G," *IEEE Transactions on Green Communications and Networking*, vol. 5, no. 4, pp. 1722–1733, 2021.

[9] Y. Zhou, L. Liu *et al.*, "Service-aware 6G: An intelligent and open network based on the convergence of communication, computing and caching," *Digital Communications and Networks*, vol. 6, no. 3.

[10] I. Kunze, P. Niemietz *et al.*, "Detecting out-of-control sensor signals in sheet metal forming using in-network computing," in *International Symposium on Industrial Electronics*. IEEE, 2021, pp. 1–6.

[11] I. Kunze, R. Glebke *et al.*, "Investigating the applicability of in-network computing to industrial scenarios," in *4th International Conference on Industrial Cyber-Physical Systems (ICPS)*. IEEE, 2021, pp. 334–340.

[12] F. E. R. Cesen, L. Csikor *et al.*, "Towards low latency industrial robot control in programmable data planes," in *6th Conference on Network Softwarization (NetSoft)*. IEEE, 2020, pp. 165–169.

[13] J. Woodruff, M. Ramanujam, and N. Zilberman, "P4DNS: In-network DNS," in *ACM/IEEE Symposium on Architectures for Networking and Communications Systems (ANCS)*. IEEE, 2019, pp. 1–6.

[14] X. Yi, R. Paulet, and E. Bertino, "Homomorphic encryption," in *Homomorphic encryption and applications*. Springer, 2014, pp. 27–46.

[15] N. Samardzic, A. Feldmann *et al.*, "F1: A fast and programmable accelerator for fully homomorphic encryption," in *54th Annual IEEE/ACM International Symposium on Microarchitecture*, 2021, pp. 238–252.

[16] B. Reagen, W.-S. Choi *et al.*, "Cheetah: Optimizing and accelerating homomorphic encryption for private inference," in *Intl. Symposium on High-Performance Computer Architecture*. IEEE, 2021, pp. 26–39.

[17] B. E. Stephens, D. Grassi *et al.*, "TCP is harmful to in-network computing: Designing a message transport protocol (mtp)," in *Proceedings of the 20th ACM Workshop on Hot Topics in Networks*, 2021, pp. 61–68.

[18] I. Kunze, "Transport protocols in the age of in-network computing," https://blog.apnic.net/2020/11/04/transport-protocols-in-the-age-of-in-network-computing/, accessed: 08.07.2022.

[19] R. Gennaro, C. Gentry, and B. Parno, "Non-interactive verifiable computing: Outsourcing computation to untrusted workers," in *Annual Cryptology Conference*. Springer, 2010, pp. 465–482.

[20] I. Kunze, D. Trossen, and K. Wehrle, "Evolving the end-to-end transport layer in times of emerging computing in the network," in *Proceedings of the 1st Workshop on New IP and Beyond*, 2022.

[21] A. Radwan, H. R. Chi *et al.*, "Multi-criteria modeled live service migration for heterogeneous edge computing," in *IEEE Global Communications Conference*, December 2022, pp. –.

[22] S. Kianpisheh and T. Taleb, "A survey on in-network computing: Programmable data plane and technology specific applications," *IEEE Communications Surveys & Tutorials*, 2022.

[23] N. Zilberman, "In-network computing," https://www.sigarch.org/in-network-computing-draft/, accessed: 23.10.2022.

[24] D. R. Ports and J. Nelson, "When should the network be the computer?" in *Workshop on Hot Topics in Operating Systems*, 2019, pp. 209–215.

[25] B. Gulecyüz, L. Oppici *et al.*, "Learning-adaptive deadband sampling for teleoperation-based skill transfer over the tactile internet," in *Intl. Symposium on Wireless Communication Systems*. IEEE, 2021, pp. 1–6.

[26] M.-J. Montpetit, "In Network Computing Enablers for Extended Reality," Internet Engineering Task Force, Internet-Draft draft-montpetit-coin-xr-03, Jul. 2019, work in Progress. [Online]. Available: https://datatracker.ietf.org/doc/draft-montpetit-coin-xr/03/

[27] J. I. Benedetto, G. Valenzuela *et al.*, "MobiCOP: a scalable and reliable mobile code offloading solution," *Wireless Communications and Mobile Computing*, 2018.

[28] 3GPP, "System architecture for the 5G System," 3rd Generation Partnership Project, Technical Specification 23.501, 09, version 17.6.0.

[29] ——, "NG-RAN; Architecture description," 3rd Generation Partnership Project , Technical Specification (TS) 38.401, 09, version 17.2.0.

[30] ——, "Procedures for the 5G System," 3rd Generation Partnership Project, Technical Specification (TS) 23.502, 09, version 17.6.0.

[31] ——, "Study on Architecture for Next Generation System," 3rd Generation Partnership Project, Technical Report 23.799, version 14.0.0.