Procedure-based Functional Decomposition for 5G Core Network Functions

Endri Goshi*, Raffael Stahl*, Mu He†, Rastin Pries†, Wolfgang Kellerer*
*Technical University of Munich, †Nokia,
Munich, Germany
E-Mail:*{endri.goshi, raffael.stahl, wolfgang.kellerer}@tum.de, †{mu.he, rastin.pries}@nokia.com

***Abstract*—The adoption of the cloud-native paradigm in mobile core networks has introduced increased traffic in the control-plane stemming from the inter-Network Function (NF) communication. In this paper, we evaluate a different functional (de)composition for the 5G Core NFs with the goal of achieving high performance and efficient orchestration while reducing the inter-NF traffic.**

I. Introduction

Borrowing the concepts from IT applications, the adoption of a microservice architecture and cloud-native deployment has led to the introduction of the Service-Based Architecture (SBA) for the 5th Generation (5G) mobile core networks in 3GPP Release 15 [1]. SBA allows the 5G Core (5GC) Network Functions (NFs) to be developed and orchestrated individually, thus providing high flexibility in these regards. To process the incoming control-plane (CP) traffic, the NFs rely on communication through standardized Service-Based Interfaces (SBIs).

From a CP procedure point of view, such functional decomposition leads to tight inter-NF dependencies and consequently the formation of service chains inside the 5GC, which we have previously investigated in [2].

Overall, the adoption of SBA leads to the following issues:

- *High volume of CP traffic* - This problem becomes more evident when deploying stateless NFs. In such 5GC deployments, additional communication with the *state database* is necessary for processing the incoming requests. As a result, this leads to increased procedure completion times [3].
- *More complex orchestration* - As the successful execution of CP procedures relies on inter-NF communication, it is more complex to orchestrate the 5GC while providing optimal performance. Thus, offloading individual NFs to the edge may not always yield the desired effects because of the requirement to communicate with the other NFs deployed in the central office.

Aiming to tackle the above issues, we propose a complete re-architecture of cloud-native 5GC using a functional (de)composition that is based on the CP procedure that they will serve. Authors in [4] apply a similar methodology but tailored only for LTE/4G and with limited scope as an offloading mechanism. Furthermore, we highlight the improvements in terms of easier orchestration and faster deployment and migration of procedure-based NFs.

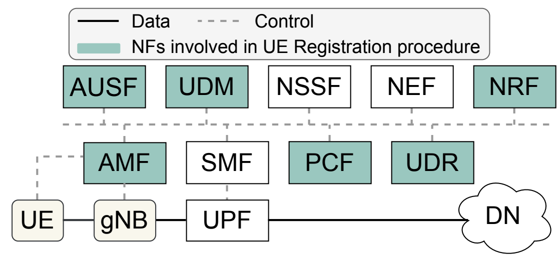


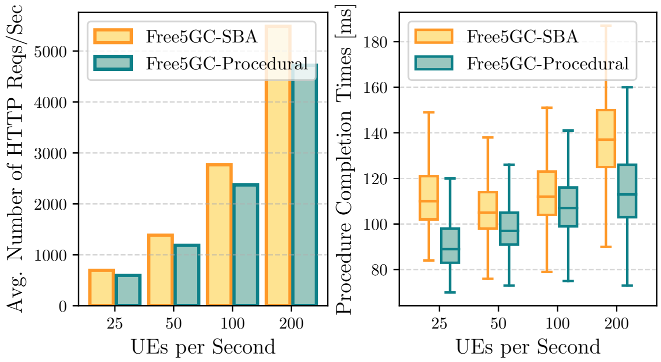
Fig. 1: 5G Core Service-Based Architecture. Highlighted in green are the NFs involved in the execution of the UE Registration procedure.

II. System Design and Implementation

The functional decomposition of NFs in the 5GC is a continuation of the approaches taken for the virtualization of legacy LTE networks. While new NFs are introduced, and further decomposition of the previous core entities is performed (e.g., AMF and SMF implement functionalities previously performed in the MME), the 5GC CP is designed such that it can serve user procedures only through communicating with other NFs. Therefore, this design inherently leads to higher volume of CP traffic and increased latency.

In contrast, our system takes a procedure-oriented approach. First, we identify the NFs that take part in the execution of different CP procedures, e.g., *UE Registration* as highlighted in Fig. 1. Next, we identify the function blocks within these NFs that are triggered during the execution of the given procedure. These blocks are then implemented in a single NF (e.g., *RegistrationNF*) that will solely serve a given procedure. An exception here is UDR which functions as a database abstraction and would still remain as a separate NF. In this way the procedure execution can be handled almost entirely within one NF, greatly reducing inter-NF communication.

For the implementation of our procedure-based 5GC, we consider *Free5GC* [5] which is an open-source mobile core implementation conforming to 3GPP Release 15. Based on its source code, we have implemented the procedure-based *RegistrationNF*. The functionality of this NF consists of functional blocks previously implemented in AMF, AUSF and PCF. AMF is the entry point of traffic coming from the radio access network to the core. Therefore, we use it as a base NF and extend it with some functional blocks of AUSF and PCF.



(a) (b)

Fig. 2: Comparison of SBA and Procedural Free5GC in terms of *(a)* number of HTTP requests/second and *(b)* PCTs during the execution of *UE Registration* procedure.

The Registration NF retains the NGAP/NAS processing capabilities of the AMF. This is necessary to be able to decode incoming traffic and correctly dispatch it to the corresponding handling functions. To identify the blocks in AUSF and PCF that are called during the registration procedure, we look at the generated HTTP requests and their destination SBIs. Next, we migrate the *callback* functions from these NFs and incorporate them into the RegistrationNF. To illustrate this with an example, the AUSF's `/nausf-auth/` endpoint is now part of RegistrationNF.

Moreover, the RegistrationNF's context consists of a unified context that was previously held in each of the aforementioned NFs. In our implementation, we were careful to avoid duplicated information e.g., regarding *UE Identity*, and remove redundant information such as context about AUSF and PCF which are held in AMF in the case of SBA. Lastly, we implement a serialization mechanism that would allow us to easily store this context in a *state database* (UDSF), making the adoption of the stateless paradigm feasible.

III. Evaluation and Results

To evaluate our design and compare it with the standardized SBA, we consider a private cloud environment orchestrated with Kubernetes [6]. Our cluster consists of 9 bare-metal nodes interconnected through a 1G switch and each hosts a single 5GC NF. In addition, we have developed a *gNB and UE Emulator* which generates input traffic to the 5GC by initiating new CP procedures for an arbitrary number of UEs/second.

With regards to the UE Registration procedure, we consider four scenarios with $[25, 50, 100, 200]$ new UEs every second. Each of the measurements runs for 100 seconds and the deployment is restarted after each run. Our evaluation of the architecture with respect to the UE Registration procedure is based on the following two key performance indicators (KPIs):

- *Number of HTTP requests/second* - corresponds to the total HTTP requests that are generated inside 5GC as a result of the inter-NF communication. It is calculated as the sum of the requests received at each of the exposed NFs' endpoints.
- *Procedure Completion Time (PCT)* - refers to the total time needed to successfully complete the CP procedure and is calculated on the Emulator side from the moment the first request is sent until the `RegistrationComplete` message is received.

In Fig. 2 (a) we compare the average number of HTTP requests/second that are generated in 5GC for the different configurations of UEs/second. As this KPI is directly correlated to the number of UEs/second, we observe a 17.5% reduction of the CP traffic for all the scenarios. Integrating UDM into the RegistrationNF is expected to yield additional improvement. Due to its functionality, in case of UE Registration UDM is responsible for 28.5% of the traffic generated in 5GC SBA.

In terms of PCTs, the adoption of the procedure-based architecture results in an improvement as well. Our evaluations show that the integration of AMF, AUSF and PCF in the RegistrationNF can result in mean PCTs reduction of up to 17% compared to the SBA deployment as shown in Fig 2 (b).

A procedure-based architecture significantly reduces the number of NFs that take part in a CP procedure. Therefore, it reduces the complexity of orchestrating and scaling the 5GC. As a plus, it improves the initialization time because all the necessary processing logic is contained in a single NF in comparison to SBA where all the NFs need to be initialized before processing can take place.

IV. Conclusions and Future Work

In this work, we propose a complete re-architecture of 5G and beyond mobile core networks with procedure-based functional (de)composition. Our initial evaluations show that the adoption of such architecture, even to a small degree, can improve the performance.

As future work, we first plan to extend the procedure-based implementation for additional procedures such as *PDU Session Establishment*, *UE Deregistration*, etc., to then perform system-wide comparisons between the two architectures. Furthermore, as cloud-native deployments remain at the core of 5GC orchestration, it would be valuable to compare stateless versions of service- and procedure-based architectures. Lastly, we argue that our proposed architecture would benefit the orchestration and as such we want to quantify its improvements in terms of initialization and migration time.

References

[1] 3GPP, "3GPP TS 23.501 - System architecture for the 5G System (Rel 15)," 2021.

[2] E. Goshi, M. Jarschel, R. Pries, M. He, and W. Kellerer, "Investigating inter-nf dependencies in cloud-native 5g core networks," in *2021 17th International Conference on Network and Service Management (CNSM)*, 2021, pp. 370–374.

[3] U. Kulkarni, A. Sheoran, and S. Fahmy, "The Cost of Stateless Network Functions in 5G," *ANCS*, 2021.

[4] M. T. Raza, D. Kim, K.-H. Kim, S. Lu, and M. Gerla, "Rethinking LTE network functions virtualization," in *2017 IEEE 25th International Conference on Network Protocols (ICNP)*, 2017, pp. 1–10.

[5] "Free5GC," https://www.free5gc.org/, 2022.

[6] "Kubernetes," https://kubernetes.io/, 2022.