IEEE Open Journal of the Communications Society

Received 17 January 2025; revised 3 March 2025; accepted 20 April 2025. Date of publication 25 April 2025; date of current version 20 May 2025.

Digital Object Identifier 10.1109/OJCOMS.2025.3564538

Performance Analysis of 5G Positioning Procedures on Resource-Constrained Devices

NICO KALIS, CHRISTIAN HAUBELT, AND FRANK GOLATOWSKI (Member, IEEE)

Institute of Applied Microelectronics and Computer Engineering, University of Rostock, 18051 Rostock, Germany

CORRESPONDING AUTHOR: N. KALIS (e-mail: nico.kalis@uni-rostock.de)

This work was supported by the KliNet5G Project funded by the German Federal Ministry for Economic Affairs and Climate Action under Grant 01MC22003G.

ABSTRACT The fast positioning of targets in indoor environments poses a significant challenge on resource-constrained devices. Therefore, this article provides novel insights into fast positioning based on the fifth generation (5G) of mobile communication. Initially, a detailed introduction to 5G positioning focusing on a network-based approach that enables Enhanced Cell ID (E-CID) Location Information Transfer is given. To highlight this positioning approach, this article proposes an implementation of corresponding positioning procedures that are based on the New Radio Positioning Protocol A (NRPPa). Additionally, the performance of these positioning procedures is also investigated in terms of their time behavior using statistical methods. More specifically, based on a 5G positioning system, which is executed on two Raspberry Pi 5 single board computers, the Round Trip Time as well as the Periodical Time Interval metrics are examined depending on different parameters, such as the task priority, the thread pool size and the number of parallel executed positioning procedures. The results show that a resource-constrained Raspberry Pi 5 in combination with the proposed implementation is capable to handle the maximum of 256 parallel positioning procedures. In this case, however, the Round Trip Time increases by a factor of up to 829 compared to sequential processing. Furthermore, it is generally possible to transmit measurement data periodically, if the smallest Periodical Time Interval of 120 milliseconds is selected according to the 5G standard. In contrast to the Round Trip Time, the deviation of the Periodical Time Interval from the expected value can be kept largely constant independently of the degree of parallelism, if the thread pool size is decreased.

INDEX TERMS 5G, positioning, NRPPa, enhanced cell ID, network performance.

I. INTRODUCTION

THE POSITIONING of targets in different environments can be a time-consuming and difficult process. Besides the environment, the performance also depends on the *technology* that is used for the recording and transmission of measurement data. If the latter is mandatory and requires an inordinate amount of time, the positioning becomes time-critical, for instance, in emergency cases. To emphasize the importance of time for positioning in indoor environments, potential use cases can be found in hospitals. On the one hand, high accuracy and low latency are required to enable the autonomous navigation of mobile service robots. On the other hand, patients in need must be localized passively without delay. Additionally, the *real-time* tracking of the targets in both scenarios could be a possible use case.

With respect to a suitable technology for positioning, mobile communication can be used. From a historical perspective, the *3rd Generation Partnership Project (3GPP)* added enhanced positioning capabilities to the *fourth generation (4G)* of mobile communication in 2009. As described in [1], these positioning features enable the localization of *User Equipments (UEs)* more accurately within cellular networks than before. With the introduction of 5G, however, positioning has been further developed, where its capabilities are becoming increasingly extensive [2].

This trend can be emphasized by the initiation of positioning of a UE. Especially in 5G networks, the positioning is triggered by different *Location Requests (LRs)* that are summarized in [3]. In terms of the use case of patients in need, the positioning can be initiated by an *emergency*

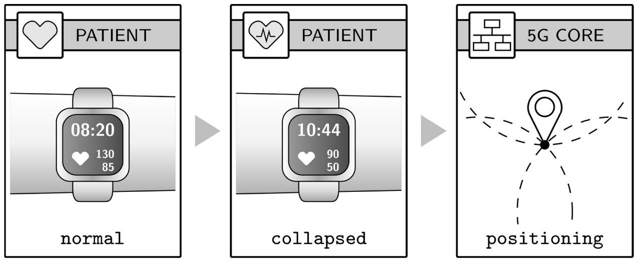

FIGURE 1. **Positioning in hospitals: A patient in need can be localized passively by a 3GPP-compliant 5G core network.**

session. Therefore, it could be initially assumed that a patient is wearing a smartwatch that is connected to a 5G core network and monitors different parameters, for instance, the blood pressure, as shown in Fig. 1. Then, in a scenario where the patient has collapsed, for instance, due to a heart disease, the monitored parameters change rapidly. These changes can cause the smartwatch to send a request to the 5G core network to initiate such an emergency session that can trigger the positioning of the collapsed patient [3].

After the positioning has been initiated, different algorithms can be used for positioning. To investigate them in terms of their time behavior, an experimental 5G positioning setup could be used. Various open source implementations of the 5G core network, such as *OpenAirInterface (OAI)* and *Open5GS* in combination with *Software Defined Radios (SDRs)* and *Radio Access Network (RAN)* solutions like *srsRAN* already offer a reliable platform to enable the implementation and evaluation of different positioning algorithms [4], [5], [6]. However, the location information exchange within a 5G system using positioning procedures, which are defined in [2], are currently only partially provided by these open source solutions and some commercial systems.

In order to realize the location information exchange, the *Location Management Function (LMF)* of the 5G core network and the corresponding positioning protocols, such as the *LTE Positioning Protocol (LPP)* and the *New Radio Positioning Protocol A* have to be implemented. While the former represents the key enabler for UE positioning and is standardized by 3GPP in [3] and [7], the specifications of LPP and NRPPa can be found in [8] and [9], respectively.

Regarding the illustrated use case in Fig. 1, this article deals with the performance of selected 5G positioning procedures in emergency cases to investigate how time-consuming the request of measurement data based on NRPPa can be. However, the NRPPa performance analysis of the article at hand is performed under ideal conditions. Consequently, the investigations are limited to controlled experiments, which means that no real-world scenarios are considered. More precisely, the article at hand presents two contributions:

① detailed introduction to the working principle of NRPPa positioning procedures as part of a 3GPP-compliant 5G positioning system in Section II-C.

TABLE 1. **Frequently used abbreviations.**

3GPP	–	**3**rd **G**eneration **P**artnership **P**roject
AMF	–	**A**ccess and **M**obility Management **F**unction
ASN.1	–	**A**bstract **S**yntax **N**otation One
E-CID	–	**E**nhanced **C**ell **ID**
GMLC	–	**G**ateway **M**obile **L**ocation **C**entre
gNB	–	fifth **g**eneration **N**ode**B**
IMSI	–	**I**nternational **M**obile **S**ubscriber **I**dentity
IP	–	**I**nternet **P**rotocol
LCS	–	**L**o**C**ation **S**ervices
LMF	–	**L**ocation **M**anagement **F**unction
LPP	–	**L**TE **P**ositioning **P**rotocol
NF	–	**N**etwork **F**unction
NG-C	–	**N**ew **G**eneration - **C**ontrol Plane
NGAP	–	**N**ew **G**eneration **A**pplication **P**rotocol
NI-LR	–	**N**etwork **I**nduced - **L**ocation **R**equest
NR-Uu	–	**N**ew **R**adio - **U**TRAN to **u**ser equipment
NRPPa	–	**N**ew **R**adio **P**ositioning **P**rotocol A
POSIX	–	**P**ortable **O**perating **S**ystem **I**nterface
RAN	–	**R**adio **A**ccess **N**etwork
SBI	–	**S**ervice-**B**ased **I**nterface
SCTP	–	**S**tream **C**ontrol **T**ransmission **P**rotocol
TCP	–	**T**ransmission **C**ontrol **P**rotocol
UE	–	**U**ser **E**quipment

② performance analysis of two NRPPa positioning procedures on *resource-constrained devices*. These procedures are used to request measurement data as input for potential E-CID positioning algorithms [2].

In the following, Section II introduces various topics related to 5G positioning that are fundamental for the further understanding of this article. Afterwards, Section III gives an overview of related work. Then, Section IV describes the theoretical approach of how the performance of selected NRPPa procedures is investigated. Based on this, the experimental setup and the evaluation of the achieved results of the NRPPa performance analysis are considered in Sections V and VI, respectively. Section VII discusses different assumptions and simplifications that have to be taken into account when evaluating the achieved results. Finally, Section VIII summarizes the key points of this article. To generally improve the readability of each section, Table 1 and 2 summarize the most frequently used abbreviations as well as the defined mathematical symbols, respectively.

II. BACKGROUND

Based on the corresponding 3GPP standards, this section gives a general overview of 5G positioning. In this context, the upper part of Fig. 2 shows the 5G positioning architecture and its interfaces, which are introduced in Section II-A.

On the other hand, the lower part of Fig. 2 describes a certain sub-procedure to realize positioning within 5G networks. Due to the use case of a patient in need, where positioning shall be realized within a 5G network, this sub-procedure is introduced as part of the 3GPP-compliant network-based positioning approach in Section II-B.

TABLE 2. Overview of defined mathematical symbols.

Symbol	Description	Reference
X	sample of measured values of a metric	Eq. (1)
X_i	single measurement as a random variable	Eq. (1)
t_r	Round Trip Time	p. 8
t_p	Periodical Time Interval	p. 8
$\overline{X}$	sample mean	Eq. (2)
S^2	sample variance	Eq. (3)
S	sample standard deviation	Eq. (6)
$x_i, \overline{x}, s$	concrete values of X_i, $\overline{X}$ and S	p. 8
n	sample size	Eq. (1)
μ	mean	Eq. (4)
$\sigma_{\overline{x}}$	standard error	Eq. (5)
$1-\alpha$	confidence level	p. 8
$z_{\frac{\alpha}{2}}$	quantile of the standard normal distribution	Eq. (7)
μ_t	expected mean of t_p according to 3GPP	Eq. (8)
Ω_{t_r}	investigated parameter tuple of metric t_r	Eq. (12)
Ω_{t_p}	investigated parameter tuple of metric t_p	Eq. (13)
ρ_{prio}	priority of an executed program	Eq. (14)
n_{ue}	number of parallel NRPPa procedures	Eq. (15)
n_{pool}	thread pool size	Eq. (16)
n_{gnb}	threads of the program `gnb`	Eq. (20)
$n_{\text{amfd}}, n_{\text{lmfd}}$	threads of the programs `amfd` and `lmfd`	Eq. (21), (22)
$n_{\text{n2}}, n_{\text{sbi}}$	threads for the N2 or SBI interface	Eq. (23)
n_{rprt}	threads for E-CID Measurement Report	pp. 11, 13
$\hat{n}_{\text{m}}$	max. number of measurement items	Eq. (27)
$t_{\text{st}}, t_{\text{end}}$	timestamps for calculation of x_i	Eq. (28)
$n_{\text{core}}, n_{\text{ran}}$	active threads for performance analysis	Eq. (29), (30)
n_{kern}	number of kernel threads	Eq. (32)
$n_{\text{norm}}, n_{\text{rt}}$	kernel threads with a certain priority ρ_{prio}	Eq. (33), (34)

Due to the fact that the NRPPa can be used within the scope of network-based positioning and plays an important role for the further investigations of this article, the working principle of selected NRPPa positioning procedures regarding E-CID Location Information Transfer is described in Section II-C.

A. 5G POSITIONING ARCHITECTURE AND INTERFACES

According to [10], a 5G system generally consists of different *network functions (NFs)* that are organized in a *service-based architecture*. As shown in the upper part of Fig. 2, these NFs can be classified within a 5G positioning system as follows:

(I) user equipment, which represents the target device that is localized within a specific area, for instance, a smartphone, an *autonomous aerial vehicle (AAV)* or a smartwatch.

(II) radio access network, which represents at least one entity that includes both hardware and software to forward data between a UE and the core network. The provided services depend on the underlying *radio access technology (RAT)*. As illustrated in Fig. 2, several *5G Node B (gNB)* form the RAN in a 5G *standalone* system.

(III) core network, which consists of selected NFs that are necessary for positioning. As described in [3], the LMF, for instance, is the network function that is responsible for determining the location of a target UE by gathering measurement data as input for a certain positioning algorithm.

(IV) external location service client, which can request *location services (LCS)* via the *Gateway Mobile Location Centre (GMLC)* to obtain position-related data of a target UE from outside of the 5G core network. All devices having access to the core network can interact as such a client, for instance, a notebook, tablet, or smartphone.

A more simplified description of the 5G positioning architecture can also be found in other papers [6], [11], [12].

When a target UE, for example, the smartwatch of the patient in need is sending data via its serving gNB over the *control plane* to the 5G core network, the transmitted data are first interpreted by the *Access and Mobility Management Function (AMF)*. Depending on the data, the AMF initiates further procedures while interacting with other NFs of the 5G core network, for instance, to forward position-related data to the LMF.

In contrast to the AMF, the GMLC is working as a gateway for external clients requesting for location services. It also forwards position-related data from the core network to external clients without the need for a prior request, for instance, if positioning was triggered by an initiated emergency session.

To access the GMLC via a LCS client, the *Mobile Location Protocol (MLP)* has to be used [3]. This protocol is defined by the *Open Mobile Alliance (OMA)* in [13].

Besides the AMF, LMF, and GMLC of the 5G core network, the 3GPP defines further network functions in [10], such as the

- *Unified Data Management (UDM)*,
- *Network Exposure Function (NEF)* and
- *Network Data Analytics Function (NWDAF)*

that are also relevant for 5G positioning, as illustrated in Fig. 2. However, due to their complexity, these network functions are not discussed in more detail within this article.

LPP and NRPPa are primarily used for position-related data exchange within a 3GPP-compliant 5G positioning system. In this context, Fig. 2 shows the used protocol stacks of the involved control plane interfaces NR-Uu and NG-C[1] as well as the Le interface that connect the components of the 5G system. Furthermore, the *service-based interface (SBI)* describes the N* interfaces between the NFs of the core network [14].

Furthermore, Fig. 2 shows that LPP is used for the transport of position-related data between a UE and its serving LMF, while NRPPa enables the data exchange between the RAN and the LMF [2]. Consequently, the main task of the RAN and the AMF is to forward position-related

[1]According to [3], it is also referred to as N2 interface.

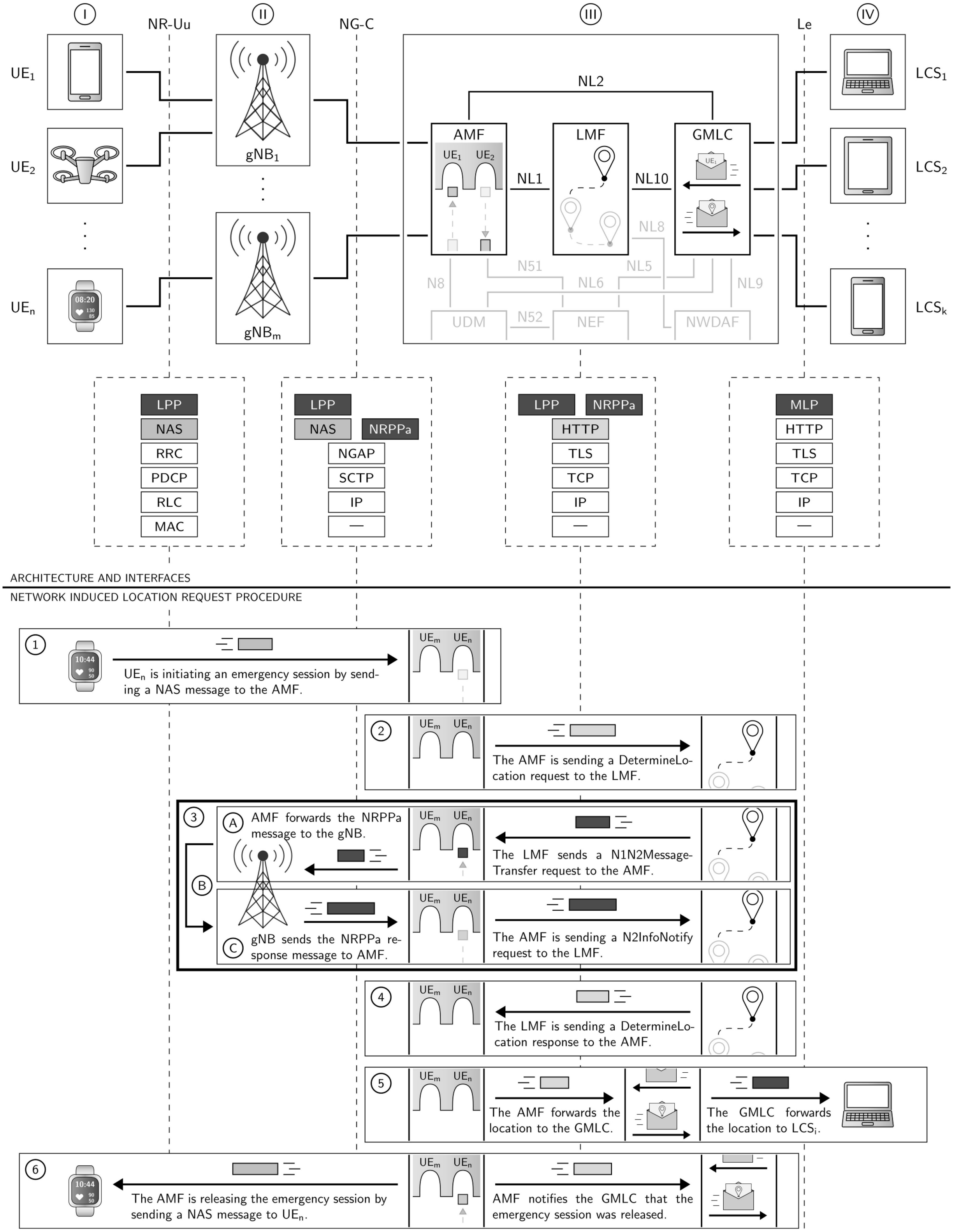

FIGURE 2. Network-based positioning for emergency cases within a 3GPP-compliant 5G positioning system (according to [2], [3]).

data between the LMF and the UE, and the serving gNB, respectively. In addition, they also have to transport data by using different protocol stacks, which depend on the data flow direction. The protocol stacks of the NR-Uu and NG-C interfaces, which encapsulate the LPP and NRPPa, are described by the 3GPP in [15].

B. NETWORK-BASED POSITIONING

Within a 5G system, positioning is generally part of high-level procedures that are triggered by different location requests. With respect to the use case of a patient in need, the target UE can be localized passively. For this purpose, the 3GPP defines the network-based positioning approach in [3] that can be a part of the *Network Induced Location Request (NI-LR) procedure*. Due to the fact that this procedure is only mentioned in other publications, for instance, by Dureppagari et al. in [16], this article describes the NI-LR procedure in more detail. As shown in the lower part of Fig. 2, the NI-LR procedure comprises the following steps in emergency cases:

1. emergency session initiation, which is initiated by the target UE, for instance, the smartwatch of the patient in need, by sending a certain *Non-Access-Stratum (NAS) message* to the AMF. The sent NAS message, which is defined by the 3GPP in [17], includes the NI-LR.
2. UE location request, which is sent by the AMF towards the LMF to obtain the current position of the target UE. This step is realized by invoking the *DetermineLocation* service that is provided by the LMF [7].
3. UE positioning, where the positioning protocols LPP and NRPPa are used depending on the underlying 3GPP-compliant sub-procedure. Due to the considered use case, Fig. 2 shows a simplified version of a network-based positioning sub-procedure, which based on the NRPPa [3].
4. UE location response, which is sent by the LMF towards the AMF and contains, among other things, the current position of the target UE. Depending on the request in the second step, additional information can be included, for instance, the used positioning method as well as a timestamp, as described in [3].
5. UE location forwarding, which enables external LCS clients to request the current position of a UE. Therefore, the AMF forwards the received location data to the GMLC by invoking the *EventNotify* service [18]. Then, the GMLC can sent the data to an external LCS client without the need for a prior request. This step is restricted to emergency cases.
6. emergency session release, which is realized by the AMF by sending a corresponding NAS message to the target UE and by invoking the EventNotify service towards the GMLC. This step is also restricted to emergency cases and enables the release of resources on the UE and on the GMLC side.

The shown network-based positioning sub-procedure enables the LMF to obtain measurement data from the RAN for a target UE by using the NRPPa. These measurements can be used as an input for 3GPP-compliant positioning algorithms that are summarized in [2]. The third step of Fig. 2 shows that this sub-procedure works in accordance with the following sequence of steps:

- (A) NRPPa message transfer initiation, which is initiated by the LMF. It invokes the *N1N2MessageTransfer* service towards the AMF to initiate the transport of the NRPPa request to the serving gNB [18]. After the AMF has interpreted the received data, the encapsulated NRPPa message is forwarded to the target gNB. However, before the AMF is forwarding the data, it checks if a logical connection with the target UE is established. Unless, the *Network Triggered Service Request procedure* is executed, as defined in [19]. Due to its complexity, this procedure is not shown in Fig. 2.
- (B) measurement recording, which is realized by the serving gNB. Depending on the requested parameters and the selected NRPPa procedure, the gNB generates a corresponding NRPPa response message.
- (C) NRPPa message transfer response, which is initiated by the gNB. It sends the generated NRPPa response to the AMF that invokes the *N2InfoNotify* service towards the LMF to forward the measurement results [18].

As mentioned in [3], these steps can be repeated to request further location information of a target UE.

According to 3GPP in [2] and [3], it is also possible to combine LPP and NRPPa procedures to realize network-based positioning. In addition, there are also NRPPa procedures, which allow the LMF to get measurement data independently of a certain UE. However, both scenarios are not considered in more detail in the article at hand.

C. NRPPA E-CID LOCATION INFORMATION TRANSFER

In general, NRPPa procedures are divided into two classes. While a *Class 2* procedure only consists of a single request, *Class 1* procedures always expect a response to the transmitted request [9], [20]. For example, the NRPPa defines a set of procedures with respect to E-CID Location Information Transfer. These procedures are standardized by the 3GPP in [9] and enable the integration of NR E-CID positioning algorithms into a 5G system [2]. In this context, the article at hand investigates the performance of the following NRPPa procedures:

- *E-CID Measurement Initiation* (Class 1) and
- *E-CID Measurement Report* (Class 2).

Both of them can be used within the scope of the network-based positioning sub-procedure.

Fig. 3 shows the working principle of both investigated NRPPa procedures. This is, to the best of the author's knowledge, not currently described by other papers. As illustrated in the first step of Fig. 3, the E-CID Measurement Initiation procedure is always initiated by the LMF, while the gNB initiates the E-CID Measurement Report procedure in the seventh step. According to [2] and [9], the former procedure is used to request measurement data of physical layer measurements as *SS-RSRP* and *SS-RSRQ*, which are

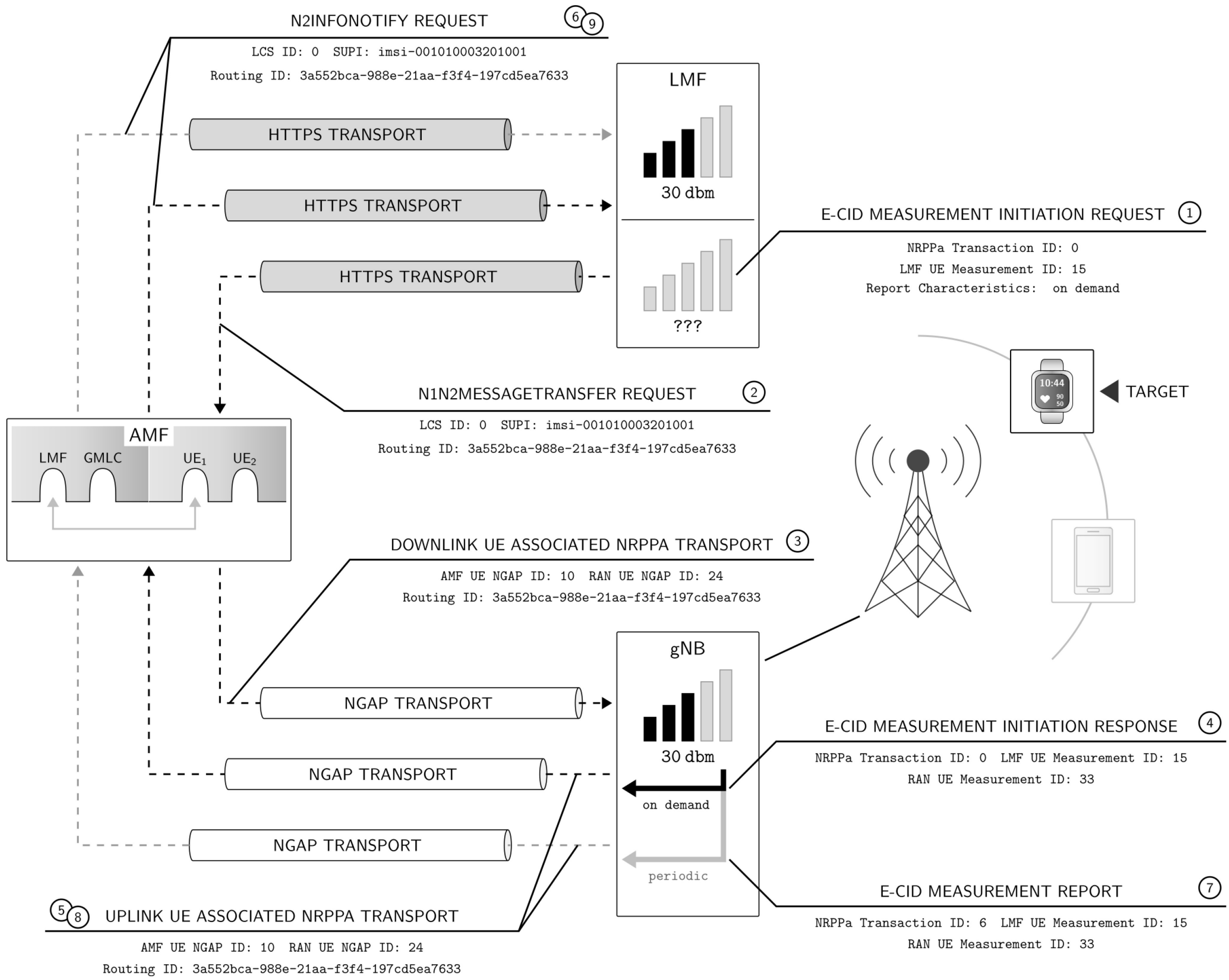

FIGURE 3. Logical UE connection assignment and transport identifiers of the investigated NRPPa procedures.

defined by 3GPP in [21]. To obtain these measurements from the serving gNB, the LMF must decide between the *on demand*, and the *periodic* response message type, as shown in the forth and the seventh step of Fig. 3. If the LMF prepares its E-CID Measurement Initiation request for the latter case, the measurement results are not directly included in the corresponding response message. Instead, the serving gNB is sending the measurement results periodically within additional E-CID Measurement Report messages at pre-defined time intervals, which can be found in [9].

To exchange NRPPa messages between the AMF and a gNB over the NG-C interface, the Class 2 *New Generation Application Protocol (NGAP)* procedures

- *Downlink UE Associated NRPPa Transport* and
- *Uplink UE Associated NRPPa Transport*

are used. Both procedures are defined by the 3GPP in [20] and shown in the third as well as in the fifth and eighth step of Fig. 3, respectively.

In order to enable the LMF to address the target gNB correctly and vise versa, different *identifiers* are defined by the 3GPP. As illustrated in each step in Fig. 3, the following identifiers are used:

- LCS Correlation Identifier, which characterizes the LCS session of a UE between the AMF and the LMF [3]. This identifier is defined as a string, with a maximum length of 255 characters [7]. The AMF generally includes this identifier in a DetermineLocation request, which is sent to the LMF.
- Subscription Permanent Identifier (SUPI), which represents a global unique UE identifier [22]. In this article, this identifier corresponds to the *International Mobile Subscriber Identity (IMSI)* that is used by the AMF and the LMF to address a UE. The pattern of an IMSI is also defined by the 3GPP in [22].
- Routing ID, which is used by the AMF to forward LPP and NRPPa messages to the target LMF [18]. The Routing ID identifies the LMF and is also included in a

NGAP message, which encapsulates NRPPa messages, as shown in Fig. 2. The pattern of this 16 Byte identifier is defined in [23].

- AMF UE NGAP ID and RAN UE NGAP ID, which are allocated by the AMF and a gNB, respectively. They indicate the logical connection of a UE over the NG-C interface. Both are included in a NGAP message and have to be stored in both network functions while the logical connection of the target UE exists [24]). All following NGAP messages that are associated with the target UE have to include the allocated identifiers. The range of both identifiers is equal to $[0, 2^{40}-1]$ and $[0, 2^{32}-1]$, respectively.
- NRPPa Transaction ID, which is included in NRPPa messages belonging to the same NRPPa procedure. It is unique for all parallel NRPPa procedures with the same procedure code and that are initiated by the same network function [9]. The range of this identifier is defined as follows: $[0, 2^{15}-1]$.
- LMF UE Measurement ID and RAN UE Measurement ID, which are allocated by the LMF and a gNB, respectively. They represent the logical connection of a UE while the NRPPa E-CID Location Information Transfer is performed. Both are included in NRPPa E-CID Measurement procedures and follow the same mechanism as the AMF UE NGAP ID and the RAN UE NGAP ID [9]. The range of these identifiers corresponds to $[1, 2^8]$.

To forward NRPPa messages from the AMF to the LMF by using the N2InfoNotify service, as shown in Fig. 2, the AMF requires a callback *Unique Resource Identifier (URI)* to address the LMF correctly. As mentioned in [25], such a URI can be defined implicitly or explicitly by using a *subscription service*. For instance, the LMF can explicitly invoke the *N1N2MessageSubscribe* service towards the AMF to subscribe to receiving NRPPa messages from a gNB. In contrast to this, the *N1N2MessageUnSubscribe* service can be invoked to unsubscribe from the reception of NRPPa messages [18].

III. RELATED WORK

Already in 2022, Pinto et al. described an initial implementation approach of the LMF and its performance in [26] and [27]. For their proposed solution, the authors extended OpenAirInterface by the LMF and focused on the performance in terms of latency and scalability. However, in comparison with the current versions of the corresponding 3GPP standards, Pinto et al. simplify their 5G positioning system, for instance, by emulating the request of measurement data. Instead of taking into account 3GPP-compliant positioning procedures that are based on the LPP or the NRPPa, the measurement data are directly included in a DetermineLocation request message, which is shown in Fig. 2. The authors show that their simplified DetermineLocation request under high workload of the underlying system is served in less than 500 ms. In contrast to their work, this article examines the time behavior of two NRPPa procedures, which have not been taken into account by Pinto et al.

Le Floch et al. also describe an implementation of the LMF as a part of their proposed 5G system, which largely consists of a commercial 5G core network as well as a commercial RAN solution [28]. More precisely, the authors divide the LMF into a controller and a set of localization functions. While the former is responsible for any interactions with the 5G core network, the latter is selected by the controller to execute the corresponding positioning algorithm. Together with a 3GPP-compliant E-CID approach using machine learning methods, Le Floch et al. show that their 5G positioning system can achieve an accuracy of less than 2.5 m for 80% of the previously recorded reference points. In [11] and [29], the authors describe the further development of their 5G positioning system. For instance, they combine their algorithms with *Pedestrian Dead Reckoning (PDR)* to increase the accuracy of positioning.

In contrast to [26], the Le Floch et al. also take the interfaces and positioning procedures into account and implement, for instance, missing NRPPa procedures. However, in comparison with the contributions of this article, the authors focus on the overall accuracy of E-CID positioning algorithms and do not consider the performance of 3GPP-compliant positioning procedures in terms of time.

Similar to Le Floch et al., Malik et al. publish a fully 3GPP-compliant 5G positioning system by extending the OpenAirInterface framework [12]. The implementation of selected NRPPa and NGAP procedures enables the authors to investigate the 3GPP-compliant *Uplink Time Difference of Arrival (UL-TDoA)* positioning method [2]. To validate their achieved results, Malik et al. use two setups. On the one hand, the implemented NRPPa procedures are tested by using a simulator-based test environment. On the other hand, the functionality of the proposed 5G positioning system is evaluated by an outdoor setup in the real world. The authors show that the proposed 3GPP-compliant positioning method can achieve an accuracy of less than 5.5 m. In contrast to the article at hand, Malik et al. also focus on the accuracy of positioning. Additionally, they investigate an UL-TDoA and not an E-CID positioning algorithm.

In summary, while Pinto et al. only focus on the LMF, Le Floch et al. and Malik et al. also take the interfaces and the 5G positioning protocols into account to evaluate the accuracy of the E-CID and UL-TDoA positioning method, respectively. However, it is currently not possible to make a statement about how different NRPPa procedures will perform in terms of time, especially on resource-constrained devices. This investigation is important for emergency cases, such as the use case of a patient in need, where time can be a critical factor. In order to close this gap, the paper at hand examines the Round Trip Time and the Periodical Time Interval of two NRPPa procedures on the Raspberry Pi 5 depending on different parameters, such as the thread pool size, the task priority and the number of parallel executed NRPPa procedures.

IV. THEORETICAL APPROACH

To evaluate the performance in terms of time of the NRPPa procedures, which are mentioned in Section II-C, the article at hand investigates the following metrics:

(I) Round Trip Time t_r, which represents the time that is required between sending and receiving a message. In the article at hand, the time between an on demand E-CID Measurement Initiation Request and its corresponding E-CID Measurement Initiation Response message is considered.

(II) Periodical Time Interval t_p, which represents the time between the reception of two consecutive E-CID Measurement Report messages.

In order to determine these metrics, statistical methods are used. The advantage of this approach is that the *uncertainty* of both metrics is also quantified. Therefore, it is initially assumed that a sample X has been taken for one of the metrics. Let

$$X := \{X_1,\ X_2,\ X_3, \dots,\ X_n \mid n \in \mathbb{N}^+\}, \tag{1}$$

where X corresponds to a set including n *independent random variables* X_i that are subject to the same unknown *probability distribution*. Each X_i is represented by a measured time difference x_i, which depends on the corresponding metric and whose calculation is explained in Section V.

As described in [30], different parameters of X can be determined, for instance, various *point estimators*, such as the sample mean $\overline{X}$ and the sample variance S^2, to obtain an estimation of t_r and t_p, respectively. If these point estimators are applied to Equation (1), $\overline{X}$ and S^2 can be calculated by:

$$\overline{X} = \frac{1}{n} \cdot \sum_{i=1}^{n} X_i, \tag{2}$$

$$S^2 = \frac{1}{n-1} \cdot \sum_{i=1}^{n} \left(X_i - \overline{X}\right)^2. \tag{3}$$

In contrast to the unknown probability distribution of X_i, the sample mean $\overline{X}$ is normally distributed, if the sample size n is larger than 30. This is substantiated by the *Central Limit Theorem (CLT)* that is also described in [30]. In this case, it can be assumed that

$$\overline{X} \sim \mathcal{N}\left(\mu,\ \sigma_{\overline{X}}^2\right), \tag{4}$$

where the *standard error* $\sigma_{\overline{X}}$ is equal to

$$\sigma_{\overline{X}} = \frac{\sigma}{\sqrt{n}} = \frac{S}{\sqrt{n}}. \tag{5}$$

Due to the fact that the introduced point estimators $\overline{X}$ and S^2 are *unbiased*, their *expected values* $E[\overline{X}]$ and $E[S^2]$ correspond to the true parameters μ and σ^2, respectively, when the sample size is increased [30]. Therefore, if the standard deviation σ is unknown, it can be replaced by the estimated sample standard deviation S, as shown in Equation (5). In general, S is equal to the positive square root of S^2:

$$S = \mid \sqrt{S^2} \mid . \tag{6}$$

To describe how close the corresponding point estimator $\overline{X}$ is to the true value of t_r and t_p, respectively, a *confidence interval* can be constructed. The latter is characterized by the *confidence level* $1-\alpha$, which defines the reliability with respect to the true value [30]. If Equation (4) applies, the confidence interval of the mean value μ of a metric can be defined independently of the probability distribution of X_i:

$$\left[\overline{x} - z_{\frac{\alpha}{2}} \cdot \frac{s}{\sqrt{n}},\ \overline{x} + z_{\frac{\alpha}{2}} \cdot \frac{s}{\sqrt{n}}\right] \forall \alpha \in (0, 1), \tag{7}$$

where $z_{\frac{\alpha}{2}}$ is a *quantile* of the standard normal distribution depending on α. $\overline{x}$ and s are concrete values of the sample mean $\overline{X}$ and the sample standard deviation S, respectively [30].

In contrast to t_r, the mean values μ_t of t_p are already defined by 3GPP in [9]. Therefore, the four smallest values of μ_t are investigated within the scope of this article:

$$\mu_t \in \{120,\ 240,\ 480,\ 640\},\ [\mu_t] = [\mathrm{ms}]. \tag{8}$$

According to Equation (7), a very precise 99% confidence interval with a confidence level based on

$$\alpha = 0.01 \tag{9}$$

as well as the corresponding quantile $z_{\frac{\alpha}{2}}$, which is equal to

$$z_{0.005} = 2.58, \tag{10}$$

requires a large sample size n. Therefore, the latter is set to

$$n = 10^4 = 10000. \tag{11}$$

Besides the selected confidence level, the impact of different parameter values of Ω_k on the confidence intervals of both metrics is investigated. Due to the fact that the definition of Ω_k depends on the considered metric, the parameter tuples Ω_{t_r} and Ω_{t_p} are defined by

$$\Omega_{t_r} := \left(n_{\text{ue}},\ n_{\text{pool}},\ \rho_{\text{prio}}\right) \tag{12}$$

and

$$\Omega_{t_p} := \left(n_{\text{ue}},\ n_{\text{pool}},\ \mu_t,\ \rho_{\text{prio}}\right), \tag{13}$$

respectively, where n_{ue} corresponds to the number of parallel NRPPa procedures of different UEs and n_{pool} is equal to the size of the used *thread pools*. ρ_{prio} is the *priority*, which is assigned to the programs of the 5G positioning system. The following values of ρ_{prio} are investigated:

$$\rho_{\text{prio}} \in \{2,\ 120\}. \tag{14}$$

In order to determine how t_r and t_p change if the values of n_{ue} and n_{pool} are doubled, their ranges are limited to the subsets

$$n_{\text{ue}} \in \{2^i \mid i \in \mathbb{N},\ i \leq 8\} \tag{15}$$

and

$$n_{\text{pool}} \in \{2^j \mid j \in \mathbb{N}^+,\ j \leq 3\}, \tag{16}$$

respectively. While the upper bound of n_{ue} is restricted by the range of the LMF UE Measurement ID of the investigated NRPPa procedures, the definition of the lower and the upper bound of n_{pool} as well as the investigated values of ρ_{prio} are explained in Section V-B.

V. EXPERIMENTAL SETUP

To determine the performance of the investigated NRPPa procedures on hardware with limited processing power, this section deals with the developed 5G positioning system and the NRPPa performance analysis on resource-constrained devices. In this context, Section V-A introduces the used hardware components and the initialization systems, which are responsible to execute the programs of the 5G positioning system. Afterwards, Section V-B addresses the determination of the investigated metrics t_r and t_p, and describes the working principle of the contributed 5G positioning system.

A. HARDWARE AND INITIALIZATION SYSTEMS

Fig. 4 shows the layering of the used experimental setup and the general working principle of the 5G positioning system, which is located on top of two initialization systems. The source code of both systems is publicly available in [31]. This section focuses on the hardware layer and the initialization systems, which form the basis for the determination of t_r and t_p.

As shown in Fig. 4, two Raspberry Pi 5 are used on hardware layer. Both single-board computers and their active coolers, whose technical specifications were taken from [32] and [33], respectively, are manufactured by the Raspberry Pi Foundation. The used Raspberry Pi 5 are connected to each other via an Ethernet cable to achieve a data transmission rate of up to 1 GBit/s. In addition, each Raspberry Pi is equipped with 8 GB *Random Access Memory (RAM)* and the BCM2712 *System-on-Chip (SoC)*, which is manufactured by Broadcom and includes a 2.4 GHz quad-core ARM Cortex-A76 CPU. To prevent the CPU from overheating while the NRPPa performance analysis is performed, an active cooler is used that is constantly running at the maximum fan speed. In terms of power supply, both Raspberry Pi are supplied with a maximum output of 27 W via their USB-C port with up to 5 A.

After both Raspberry Pi 5 have been powered on, the Linux operating system[2] is started at the end of the boot process. The Linux kernel provides different services to its *user space programs* by the *system call interface* that is compliant with the *Portable Operating System Interface (POSIX)* standard. As shown in Fig. 4, the Linux operating system is also responsible for the *process and memory management* as well as offering networking capabilities, for instance, different protocol implementations, such as IP, TCP and SCTP. The Linux kernel also enables interactions with hardware devices using different *drivers* [34]. The network interface card that is associated with the Ethernet port or a *radio unit* are examples of such devices.

To prevent other programs from affecting the NRPPa performance analysis, the first user space program that is executed by the Linux operating system has been changed. For this purpose, an initialization system has been implemented in the C programming language. The working principle of this system depends on the *role* of the Raspberry Pi within the 5G positioning system. As shown in Fig. 4, the left Raspberry Pi 5 executes the `init5g-core` initialization system to start the gNB implementation. In contrast to this, the right Raspberry Pi is responsible for the execution of the 5G core network using the `init5g-core` system.

The initialization system `init5g-core` is responsible for the execution of the 5G core network functions AMF and LMF. It also triggers the DetermineLocation request of the AMF to initiate the investigated NRPPa procedures. In accordance with Fig. 4, the following steps are performed:

- (I) parameter update, during which the parameter values of Ω_k are changed depending on the considered metric. Fig. 4 shows the update of Ω_{t_r}:

$$\left(1,\ 2,\ \rho_{\text{prio}}\right) \longrightarrow \left(2,\ 4,\ \rho_{\text{prio}}\right). \tag{17}$$

- (II) program execution, in which the programs `amfd` and `lmfd`, which corresponds to the AMF and the LMF, respectively, are executed. For this purpose, the current parameter values of Ω_k are passed to both programs to control their behavior.[3] Therefore, Ω_k specifies the NRPPa procedure and the number of its parallel executions, the thread pool size of both programs as well as the metric, for which measurement data are obtained.
- (ϟ) UE assignment, which is realized by the interruption of `amfd` by the `SIGUSR1` signal after the `amfd` has notified the initialization system by `SIGUSR1` about the successful registration of `gnb`. In order to establish a logical UE connection on NGAP layer and between the AMF and LMF to transport NRPPa messages, the required identifiers are emulated. Therefore, this step is mandatory to assign a UE to its serving gNB manually.
- (III) signaling procedure, which interrupts the execution of `amfd` by the POSIX-compliant `SIGUSR2` signal. The interrupt invokes the corresponding *signal handler* of `amfd` that triggers the transmission of a DetermineLocation request towards the `lmfd`. When the `amfd` receives the DetermineLocation response, `amfd` is also interrupting `init5g-core` by `SIGUSR2` to indicate the end of the executed NRPPa procedure. This signaling procedure is repeated m times to obtain a large sample X with a sample size n of at least 10^4 elements. Due to the fact that n_{ue}

[2] The used Linux kernel was compiled from source (version 6.6.42). It was also patched with `RT_PREEMPT` to enable its real-time capabilities.

[3] Except the priority ρ_{prio}, which is set by the target program itself.

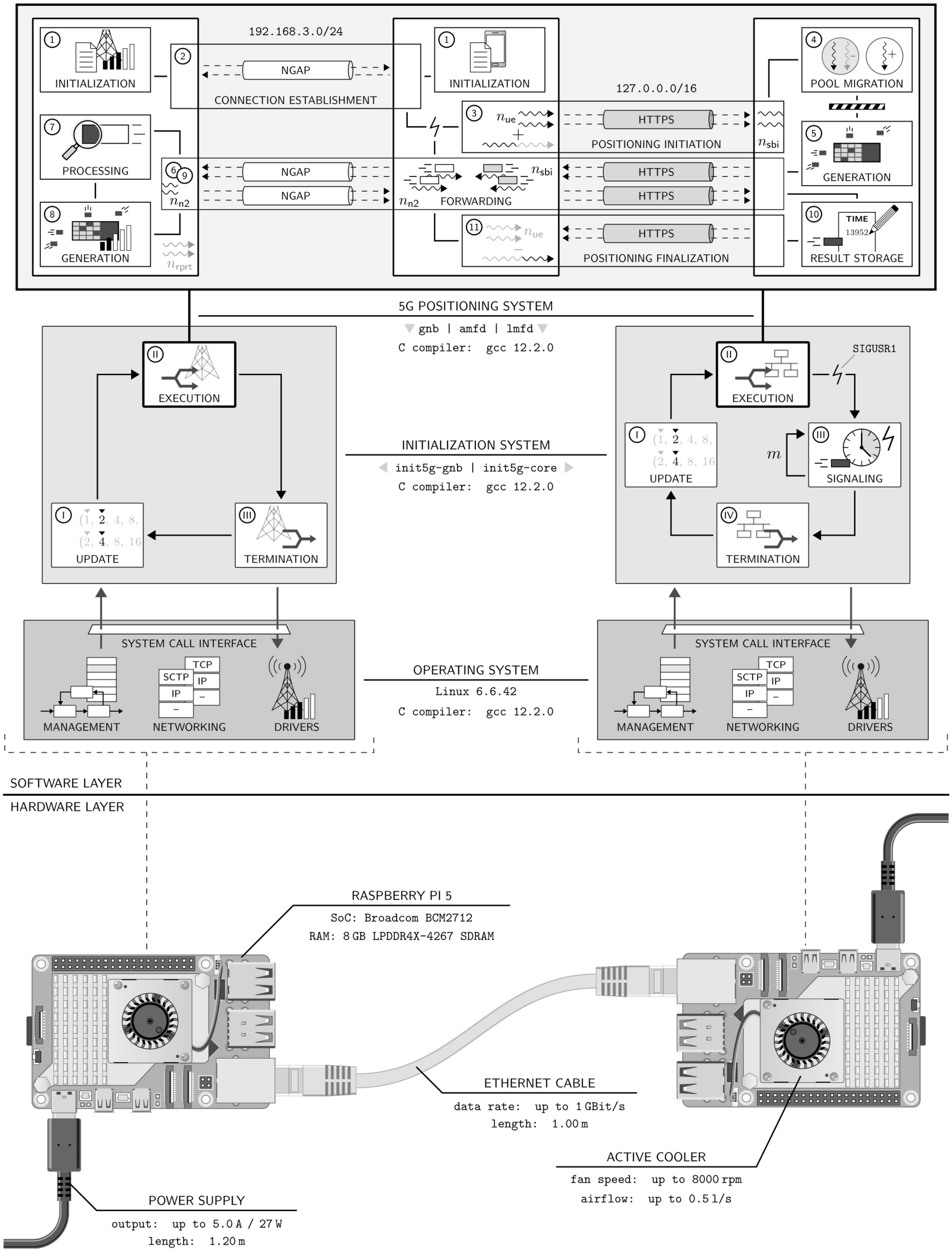

FIGURE 4. Design of the experimental setup: two Raspberry Pi 5 as resource-constrained devices on hardware layer (below) and realization of the NRPPa performance analysis based on $\Omega_{t_r} = (2,\ 4,\ \rho_{\text{prio}})$ within the developed 5G positioning system on software layer (above).

measurement values x_i are obtained after each iteration of n_{ue} parallel NRPPa procedures, m is calculated by

$$m = \begin{cases} n \cdot n_{\text{ue}}^{-1} & \text{if } n \bmod n_{\text{ue}} = 0 \\ \lfloor n \cdot n_{\text{ue}}^{-1} \rfloor + 1 & \text{otherwise} \end{cases}. \tag{18}$$

(IV) process termination, where `amfd` and `lmfd` are notified by the `SIGTERM` signal to initiate their termination by invoking their corresponding signal handlers. After both programs have been terminated, their *exit codes* are requested from the underlying operating system to prevent the inefficient usage of memory resources.

The initialization system `init5g-gnb` is responsible for the execution of the gNB implementation and the update of its parameter values, which is executed synchronously with `init5g-core`. As shown in Fig. 4, `init5g-gnb` performs the following steps:

- I parameter update, which corresponds to the first step of the `init5g-core` initialization system.
- II program execution, during which the program `gnb` is executed. In contrast to `amfd` and `lmfd`, only the updated thread pool size n_{pool} is passed to `gnb`. This is due to the fact that the main behavior of `gnb` depends on received request messages.
- III process termination, which is reached after the `amfd` program is terminated. As a consequence, the connection between `gnb` and `amfd` over the NG-C interface is closed and the `gnb` terminates. Requesting the exit code of `gnb` corresponds to the behavior of `init5g-core`.

The steps of both initialization systems are repeated for different parameter values of Ω_{t_r} and Ω_{t_p}, respectively. In addition, before these steps are executed for the first time, `init5g-core` and `init5g-gnb` initialize the system itself, for instance, they configure different network interfaces, load required drivers and perform CPU frequency scaling. However, these steps are not shown in Fig. 4.

B. THE 5G POSITIONING SYSTEM

After all programs of the 5G positioning system have been executed by the initialization systems, the NRPPa performance analysis is starting. For its realization, the block at the top of Fig. 4 primarily shows the 5G positioning system and the sequence of steps to obtain measurement data of the Round Trip Time t_r with

$$\Omega_{t_r} = \left(2,\ 4,\ \rho_{\text{prio}}\right). \tag{19}$$

In particular, the steps 5 to 10 are crucial for determining the time difference values x_i of the t_r metric. For the calculation of x_i for t_p, the key points A to D, which are mentioned at the end of this section, must also be taken into account.

The working principle of the 5G positioning system consists of several steps that can be described independently of the priority ρ_{prio} as follows:

1. initialization phase, during which `gnb`, `amfd` and `lmfd` reserve and initialize required resources, whereby the initialization phase of `lmfd` and the setting of the priority ρ_{prio} are not shown in Fig. 4. Due to the fact that `gnb` and `amfd` emulate the logical connection of different UEs to be able to transport NRPPa messages correctly, the required identifiers, such as the LCS Correlation Identifier, the IMSI, the AMF UE NGAP ID as well as the RAN UE NGAP ID are loaded from a local file. In addition, the measurement data, which are transmitted within an E-CID Measurement Initiation Response or an E-CID Measurement Report message, are also emulated by the gNB implementation. To realize this, `gnb` is opening a corresponding driver. For the handling of, for instance, incoming messages over the interfaces of each program, different thread pools are also initialized. In general, the advantage of using thread pools is the limitation of newly created threads to save memory resources by reuse. When the 5G positioning system is initialized, the total number of threads n_{i} of `gnb`, `amfd` and `lmfd` is equal to:

$$n_{\text{gnb}} = n_{\text{n2}} + n_{\text{rprt}} + 1, \tag{20}$$
$$n_{\text{amfd}} = n_{\text{n2}} + n_{\text{sbi}} + 2, \tag{21}$$
$$n_{\text{lmfd}} = n_{\text{sbi}} + 1, \tag{22}$$

with

$$n_{\text{n2}} = n_{\text{rprt}} = n_{\text{sbi}} = 0.5 \cdot n_{\text{pool}}, \tag{23}$$

where n_{n2}, n_{rprt} and n_{sbi} represent the size of their corresponding thread pool, as indicated in Fig. 4. The remaining threads corresponds to the main thread of each program, while `amfd` is using an additional thread to initially receive incoming messages over its SBI. As a consequence of Equation (23), n_{pool} should be at least equal to the value of two to ensure that each thread pool contains at least a single thread. This also explains the lower bound of n_{pool}, which is defined by Equation (16).

2. connection establishment between `gnb` and `amfd` by using the 3GPP-compliant *NG Setup procedure*, which is generally used to initialize the NG-C interface between a gNB and the AMF [20]. Because `amfd` handles each successfully registered `gnb` by a separate thread,[4] Equation (21) is updated after this step by

$$n_{\text{amfd}} = n_{\text{n2}} + n_{\text{sbi}} + 3. \tag{24}$$

- ϟ UE assignment and signaling procedure, which have been already described in context of the `init5g-core` initialization system earlier in this section.

3. positioning initiation, during which the `amfd` is sending n_{ue} DetermineLocation requests towards `lmfd` after the `SIGUSR2` signal has been caught. Due to the fact that this step is initiated by a separate thread, n_{ue} threads are created by `amfd`, if n_{ue} NRPPa procedures are executed simultaneously. Consequently, Equation (21) is updated to its final state:

$$n_{\text{amfd}} = n_{\text{n2}} + n_{\text{sbi}} + n_{\text{ue}} + 4. \tag{25}$$

The usage of a separate thread prevents the whole process from being blocked until the `SIGUSR2` signal handler finished its execution.

[4] In the scope of this article, only one `gnb` is registered at the same time.

④ thread pool migration, which is necessary to keep the thread pool size n_{sbi} constant so that `lmfd` is able to handle incoming messages over its SBI. After a DetermineLocation request has been received, the related thread is migrated to a new thread pool that exclusively handles DetermineLocation requests and increases temporarily up to a size of n_{ue}. Then, a new thread is created and added to the old thread pool to restore its size. Therefore, Equation (22) is also updated to its final state:

$$n_{\mathsf{lmfd}} = n_{\mathsf{sbi}} + n_{\mathsf{ue}} + 1. \tag{26}$$

thread synchronization, which is executed after the N1N2MessageSubscribe service was invoked by the n_{ue} threads towards `amfd` to enable the forwarding of NRPPa messages, whereby the subscription procedures are not shown in Fig. 4. The thread synchronization is applied to the n_{ue} threads of `lmfd` to ensure that they ideally execute the investigated NRPPa procedures at the same time. In order to realize the synchronization internally, a POSIX-compliant *barrier function* is used.

⑤ NRPPa request message generation, where the current timestamp t_{st} is requested from the underlying Linux operating system. This is mandatory to determine a value x_i for the metric t_r. In addition, the generated NRPPa E-CID Measurement Initiation Request queries the maximum number of measurement items $\hat{n}_{\mathsf{m}}$ of the SS-RSRP signal metric independently of t_r and t_p to take the worst case regarding the maximum size of transmitted measurement data into account. In this case, $\hat{n}_{\mathsf{m}}$ is determined as follows:

$$\hat{n}_{\mathsf{m}} = 64 \cdot 9 \cdot 64 = 36864, \tag{27}$$

which is described by the 3GPP in [9]. However, the actual size of the NRPPa E-CID Measurement Initiation Request also depends on the message transfer syntax that is specified by the *Abstract Syntax Notation One (ASN.1)* interface description language. According to [9], the Aligned variant of the *Basic Packed Encoding Rules (BASIC-PER)* has to be used.

⑥ message forwarding I, during which the NRPPa E-CID Measurement Initiation Request message is transmitted from `lmfd` to `gnb` via `amfd`. Due to the encapsulation of a NRPPa message by HTTP or NGAP, respectively, the NRPPa is generally transparent to `amfd`. As shown in the third step of Fig. 3, the encoded NRPPa message is encapsulated in a *Downlink UE Associated NRPPa Transport* message before it is sent over the NG-C interface. According to [20], the created NGAP message is also encoded by the Aligned variant of ASN.1 BASIC-PER.

⑦ message processing, which is performed by `gnb` to determine the NGAP message type. After the ASN.1 decoding of the received NGAP message is completed, further processing steps are derived. In the scope the described 5G positioning system, the NGAP message type is always equal to *Downlink UE Associated NRPPa Transport*. This is due to the fact that only NRPPa messages are exchanged between `lmfd` and `gnb`.

⑧ NRPPa response message generation, where the E-CID Measurement Initiation Response message is generated depending on the report characteristics of the decoded E-CID Measurement Initiation Request. More precisely, the emulated measurement data of the SS-RSRP signal metric are generated iteratively by the already opened kernel driver. The latter generates random numbers, which are mapped to the defined range of each value of a measurement item [9]. Afterwards, the item is passed to `gnb` via the system call interface. Because Fig. 4 shows the sequence of an on demand request, the generated measurement items are directly included in the E-CID Measurement Initiation Response message, which is also encoded by ASN.1 BASIC-PER.

⑨ message forwarding II, during which the NRPPa E-CID Measurement Initiation Response message is transmitted from `gnb` to `lmfd` via `amfd`. To transport the data over the NG-C interface, the NRPPa message is included in an *Uplink UE Associated NRPPa Transport* message.

⑩ result storage, which is performed by `lmfd` after the NRPPa E-CID Measurement Initiation Response message was successfully decoded from the transmitted N2InfoNotify message, as shown in the sixth step of Fig. 3. After the current timestamp t_{end} is also requested from the Linux operating system, x_i is calculated by:

$$x_i = t_{\mathsf{end}} - t_{\mathsf{st}}, \; [x_i] = [\mathrm{ns}]. \tag{28}$$

Finally, t_{st}, t_{end} and x_i are written to a local file.

⑪ positioning finalization, which is realized by `lmfd`. After the latter invokes the N1N2MessageUnsubscribe service that is not shown in Fig. 4 towards `amfd` to cancel the forwarding of NRPPa messages, the `lmfd` is sending a corresponding DetermineLocation response to `amfd`. While the content of the received response message is ignored by `amfd`, it is sending the `SIGUSR2` signal to `init5g-core` to invoke its corresponding signal handler. Due to the fact that the newly created n_{ue} threads of `amfd` and `lmfd` also terminate during this step, the Equations (21) and (22) are valid again.

These steps of the 5G positioning system are repeated m times for the current parameter tuple Ω_k, beginning from the third step after a `SIGUSR2` signal was caught by `amfd`.

In order to determine time measurement values x_i for the investigated t_p metric, the described sequence of steps of the 5G positioning system are almost completely the same. In contrast to t_r, the following key points differ:

Ⓐ parameter tuple Ω_k, which is passed to `gnb`, `amfd` and `lmfd` by the initialization systems. To obtain measurement values for the t_p metric, Ω_{t_p} has to be passed to each program of the 5G positioning system.

Ⓑ NRPPa message generation, where the report characteristics property of the E-CID Measurement Initiation Request is set to periodic with a period μ_t. Consequently, the emulated SS-RSRP measurement items are not directly included in the corresponding E-CID Measurement Initiation Response message. Instead, they are generated twice and sent within two additional E-CID Measurement Report messages by `gnb`. As indicated by Fig. 4, n_{rprt} threads are available to execute the periodical transmission.

Ⓒ timestamp requesting, which is responsible for the determination of t_{st} and t_{end}. In contrast to the t_r metric, both timestamps are requested from the underlying Linux operating system after an E-CID Measurement Report message was successfully decoded by `lmfd`. Therefore, at least two of these messages are required to determine a single value x_i.

Ⓓ E-CID Measurement Report Termination Command, which is generated by `lmfd` after two E-CID Measurement Report messages have been received and before x_i is written to a local file. The NRPPa message is sent to `gnb` to terminate the periodical transmission of E-CID Measurement Report messages [9].

While the NRPPa performance analysis is performed, the total number of *active* threads n_j depends on the investigated metric and the role j of each Raspberry Pi within the 5G positioning system. Based on the Equations (20), (24) and (26), n_{core} and n_{ran} are calculated by

$$n_{\text{core}} = n_{\text{amfd}} + n_{\text{lmfd}} = 1.5 \cdot n_{\text{pool}} + n_{\text{ue}} + 4, \tag{29}$$

and

$$n_{\text{ran}} = n_{\text{gnb}} = n_{\text{pool}} + 1, \tag{30}$$

respectively. If t_r is investigated, the thread pool of size n_{rprt} is not used. In this case, Equation (30) is redefined by

$$n_{\text{ran}} = n_{\text{gnb}} - n_{\text{rprt}} = 0.5 \cdot n_{\text{pool}} + 1. \tag{31}$$

It should be taken into account that the processes, which are associated with the initialization systems, also consist of a main thread. However, these threads can be ignored, because they are blocked until the time measurement value x_i has been determined. The same line of reasoning can be applied to the newly created threads of Equation (25).

Assuming that the processing of incoming messages by approximately n_{pool} threads takes proportionally the most time during the data exchange between `lmfd` and `gnb`. In conjunction with the limited number of CPU cores of the Raspberry Pi 5, the upper bound of n_{pool} was set to the defined value that is shown in Equation (16).

In addition to the threads, which are created by `amfd`, `lmfd` and `gnb`, the underlying Linux operating system is also executing a certain number of threads within its *kernel space*. To preempt most of them, a certain priority ρ_{prio} is used. Therefore, let

$$n_{\text{kern}} = n_{\text{norm}} + n_{\text{rt}} \tag{32}$$

the total number of kernel threads with

$$n_{\text{norm}} = 105 \tag{33}$$

and

$$n_{\text{rt}} = 17, \tag{34}$$

where n_{norm} kernel threads have been assigned the lowest or no real-time priority. Consequently, a thread with

$$\rho_{\text{prio}} = 2 \tag{35}$$

always displaces each of the n_{norm} threads.[5] In contrast to this, there are n_{rt} threads to which a real-time priority of 50 and higher have been assigned. These should not be preempted, because they are responsible for the task migration between different CPU cores and the handling of interrupt requests, for instance, incoming data frames over the Ethernet interface.

Furthermore, priorities between 100 and 139 are assigned to *normal* tasks, which are always displaced by a task with a lower priority value. Due to this behavior, the values of ρ_{prio} defined by Equation (14) are used to examine the impact of the Linux operating system on the metrics t_r and t_p.

In order to assign tasks to available CPU cores, the Linux operating system provides different *scheduling policies*. In this article, the `SCHED_RR` *round-robin* scheduling policy is applied to tasks with the real-time priority ρ_{prio}, which is defined by the Equation (35). Otherwise, the default `SCHED_NORMAL` *time-sharing* scheduling policy is used.

VI. EVALUATION OF RESULTS

This section deals with the determined results to evaluate the investigated metrics t_r and t_p that based on the taken time measurement values x_i. In this context, it should be mentioned that the recording of n time measurement values takes a lot of time. For instance, the usage of the parameter tuple Ω_{t_p} with

$$\Omega_{t_p} = \left(n_{\text{ue}},\ n_{\text{pool}},\ 640,\ 120\right) \tag{36}$$

can take up to one week.

In terms of overall performance, the following behavior is expected from the NRPPa procedures:

① Due to the limited number of available CPU cores, both metrics demonstrate a decline in performance, if the number of parallel NRPPa procedures n_{ue} increases.

② The assignment of a higher priority ρ_{prio} improves both metrics due to a lower number of competing threads.

③ A reduction of the thread pool size n_{pool} results in a higher Round Trip Time t_r, if n_{ue} also increases. The accuracy and precision of the Periodical Time

[5] Under Linux, the real-time priority of a *task* is between 1 and 99, where the former represents the lowest and the latter the highest priority.

Interval t_p, on the other hand, will improve due to less workload.
4. If μ_t becomes larger, a higher accuracy is expected.
5. The actual value of t_p is greater than the expected value μ_t. This is due to the fact that each response is sent after a time interval of μ_t.

The results of the statistical evaluation are illustrated in Fig. 5. To ensure transparency and accessibility, all collected raw data and their corresponding statistical parameters have been made publicly available in [31]. Taking into account Equation (18), if n_{ue} becomes larger, the confidence intervals based on more time measurement values x_i than required. However, a higher sample size n influences a confidence interval positively, as follows from Equation (7).

Fig. 5 proves that t_r increases, if n_{ue} is also increasing. According to the achieved results, the corresponding worst case scenario can be observed for

$$\Omega_{t_r} = (256,\ 4,\ 120), \tag{37}$$

where the Round Trip Time t_r is increased by a factor of approximately 829 compared to sequential processing with

$$\Omega_{t_r} = (1,\ 4,\ 120). \tag{38}$$

On the other hand, the impact of higher parallelism on t_p can only be proven for smaller μ_t. However, a reduction of uncertainty can be generally observed independently of the investigated metric, when n_{ue} decreases.

Due to the fact that no clear trend can be identified with regard to the impact of ρ_{prio} on t_r and t_p, respectively, it can be concluded that the assignment of a higher priority ρ_{prio} improve neither of the investigated metrics. In general, this behavior may be attributed to the different scheduling policies of the Linux operating system that are associated with ρ_{prio}.

The reduction of n_{pool} increases t_r as expected, if n_{ue} becomes closer to the number of available CPU cores. In contrast to this, a smaller thread pool size n_{pool} often improves the accuracy and precision of t_p, especially if n_{ue} becomes larger. For instance, if the parameter tuple

$$\Omega_{t_p} = (256,\ 4,\ 120,\ 2) \tag{39}$$

is applied, the deviation of the corresponding Periodical Time Interval t_p from the expected value μ_t is approximately as small as in the scenario compared to no parallelism with

$$\Omega_{t_p} = (1,\ 4,\ 120,\ 2). \tag{40}$$

Therefore, the third expectation can largely be shown.

Also the fact that a larger μ_t results in a higher accuracy can be seen in Fig. 5. This can be observed by the behavior that the deviation from μ_t becomes increasingly smaller.

As expected, the actual value of t_p is generally greater than μ_t. However, there are also estimated mean values of t_p, which are smaller than μ_t. This behavior appears to be largely independent of the priority ρ_{prio} and could depend on the thread pool size n_{pool} and the implementation design.

VII. DISCUSSION

The results of the NRPPa performance analysis are subject to the impact of assumptions and simplifications. Therefore, this section discusses different aspects that have to be taken into account for the evaluation of the achieved results.

On the one hand, the implemented 5G positioning system is *ideal*. This means that only the investigated NRPPa procedures are executed without the occurrence of error scenarios while the performance analysis is performed. Consequently, there are no other activities within the 5G positioning system, which could compete with the investigated NRPPa procedures to influence their time behavior negatively.

Secondly, before the AMF is forwarding a NRPPa message to the serving gNB, the former executes the Network Triggered Service Request procedure to check, if the target UE is in a connected state. With respect to the implementation of the 5G positioning system, it was assumed that the target UE is always connected. Therefore, the aforementioned procedure was replaced by the emulation of UEs and their identifiers. This simplification could also have a negative impact on the investigated metrics, so that the results are more positive than they would be in reality.

In a similar way, the measurement data of a NRPPa response message were also emulated. Assuming that the physical layer measurements will be obtained from a radio unit in future, a self-developed kernel driver was used to emulate the hardware access in the most optimal manner. In contrast to the previous two points, however, it is not possible to ascertain whether the impact on the time behavior is positive or negative despite several *context switches* between user and kernel space.

The investigations of this article based on the simplification that only one gNB is connected to the AMF at the same time. However, it has to be assumed that in reality several gNBs are used. Therefore, the impact of several gNBs on the time behavior of t_r and t_p, respectively, could be investigated in future work.

Furthermore, only two different priorities ρ_{prio} have been considered during the investigation of the metrics t_r and t_p. By setting a higher real-time priority than two and the usage of the `SCHED_FIFO` scheduling strategy instead of the `SCHED_RR` scheduling strategy, the standard error $\sigma_{\overline{X}}$ of the sample mean $\overline{X}$ could be reduced to a greater extent than before. Therefore, this could decrease the deviation of t_r and t_p from their mean value at a high degree of parallelism.

VIII. CONCLUSION

This article proposed a 3GPP-compliant 5G positioning system, for which the performance of different NRRPa positioning procedures have been investigated on resource-constrained Raspberry Pi 5 single board computers. More precisely, the Round Trip Time and the Periodical Time Interval have been examined taking into account different parameters, such as the scheduling priority and the limitation of the total number of threads by using thread pools.

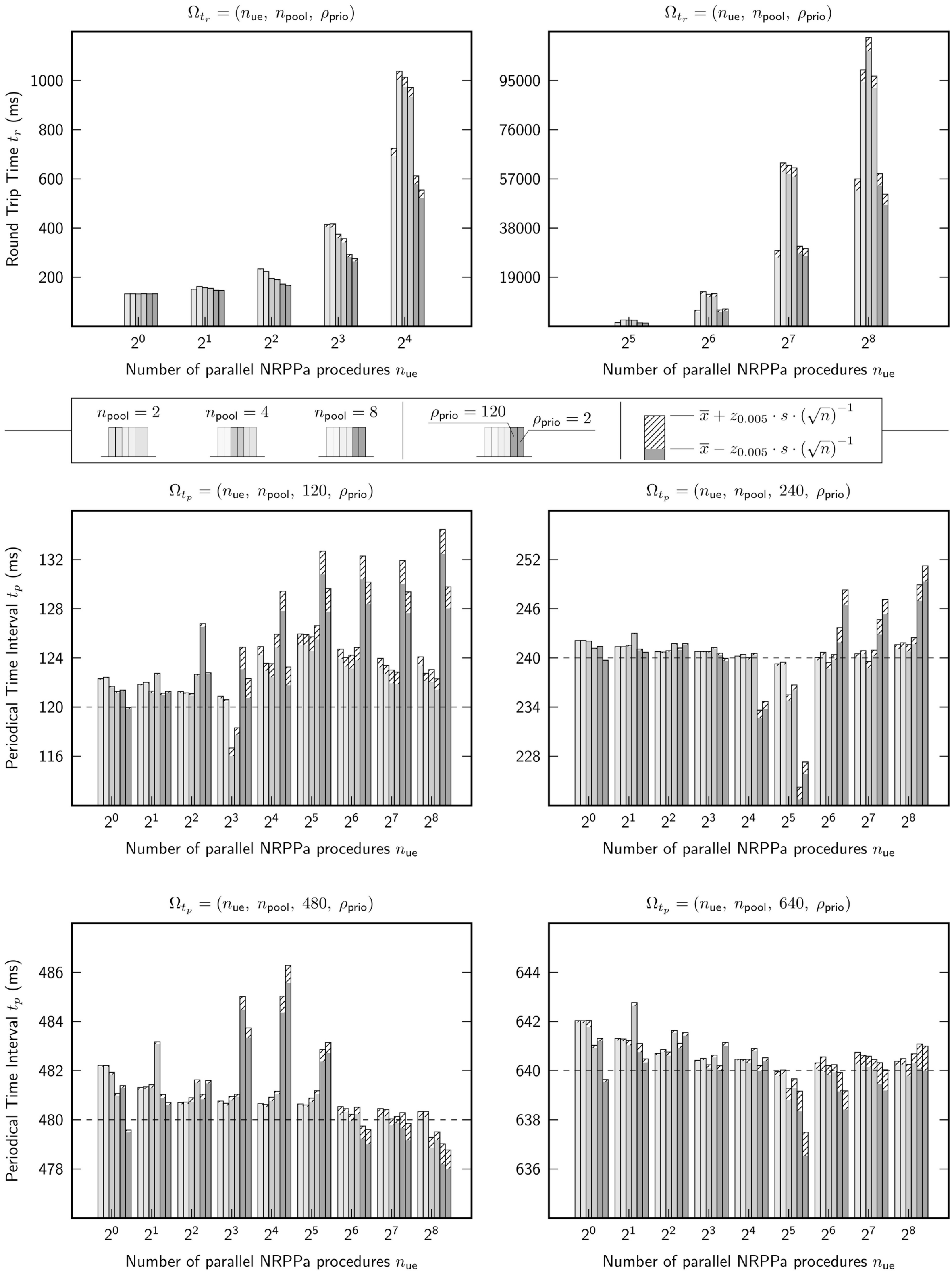

FIGURE 5. Impact of the parameter tuples Ω_k on a 99% confidence interval of the Round Trip Time t_r and the Periodical Time Interval t_p.

Despite several assumptions and simplifications, the article at hand shows that computer hardware with limited processing power is already able to execute the maximum number of 256 parallel NRPPa procedures to enable E-CID Location Information Transfer. Even with a high degree of parallelism, the fastest periodic transmission of measurement data can be ensured, if the affected parameters are set to *ideal* values. However, in terms of the use case of a patient in need, it has to be conceded that a Raspberry Pi 5 is unable to ensure appropriate performance to localize many patients simultaneously.

In particular, the usage of more powerful computer hardware should generally improve the performance of the investigated NRPPa procedures. An alternative approach for handling a high degree of parallelism could be the distribution of positioning using several LMF entities that is also compliant with the corresponding 3GPP standards [3], [7]. Assuming that resource-constrained systems are used again to investigate the time behavior in the latter case, it is necessary to utilize several of these systems, which could potentially be more resource-constrained. In this case, the potential benefits of employing the approach of this article warrant further investigation.

Due to the fact that the NRPPa performance analysis is limited to controlled experiments and simulations, the extension of the investigations by real-world scenarios could be part of future work. Also the usage of further resource-constrained devices as well as the examination of additional NRPPa and LPP procedures could be taken into account to emphasize the benefits of the study of the article at hand.

REFERENCES

[1] "Evolved universal terrestrial radio access network (E-UTRAN); Stage 2 functional specification of user equipment (UE) positioning in E-UTRAN," 3GPP, Sophia Antipolis, France, Rep. TS 36.305, Dec. 2023.

[2] "NG radio access network (NG-RAN); Stage 2 functional specification of user equipment (UE) positioning in NG-RAN," 3GPP, Sophia Antipolis, France, Rep. TS 38.305, Mar. 2024.

[3] "5G system (5GS) location services (LCS); stage 2," 3GPP, Sophia Antipolis, France, Rep. TS 23.273, Mar. 2024.

[4] L. Mamushiane, A. Lysko, H. Kobo, and J. Mwangama, "Deploying a stable 5G SA testbed using srsRAN and Open5GS: UE integration and troubleshooting towards network slicing," in *Proc. Int. Conf. Artif. Intell., Big Data, Comput. Data Commun. Syst. (icABCD)*, Aug. 2023, pp. 1–10.

[5] J. E. Håkegård, H. Lundkvist, A. Rauniyar, and P. Morris, "Performance evaluation of an open source implementation of a 5G standalone platform," *IEEE Access*, vol. 12, pp. 25809–25819, 2024.

[6] I. Palamà et al., "5G positioning with software-defined radios," *Comput. Netw.*, vol. 250, Aug. 2024, Art. no. 110595.

[7] "5G system; location management services; stage 3," 3GPP, Sophia Antipolis, France, Rep. TS 29.572, Mar. 2024.

[8] "LTE Positioning Protocol (LPP)," 3GPP, Sophia Antipolis, France, Rep. TS 37.355, Mar. 2024.

[9] "NG-RAN; NR positioning protocol a (NRPPa)," 3GPP, Sophia Antipolis, France, Rep. TS 38.455, Mar. 2024.

[10] "System architecture for the 5G system (5GS); Stage 2," 3GPP, Sophia Antipolis, France, Rep. TS 23.501, Mar. 2024.

[11] A. Le Floch, R. Kacimi, P. Druart, Y. Lefebvre, and A.-L. Beylot, "A comprehensive framework for 5G indoor localization," *Comput. Commun.*, vol. 228, Dec. 2024, Art. no. 107968.

[12] A. Malik et al., "From concept to reality: 5G positioning with open-source implementation of UL-TDoA in OpenAirInterface," 2024, *arXiv:2409.05217*.

[13] "Mobile location protocol specification," Open Mobile Alliance, San Diego, CA, USA, 2002.

[14] "5G System; technical realization of service based architecture; stage 3," 3GPP, Sophia Antipolis, France, Rep. TS 29.500, Mar. 2024.

[15] "NR; NR and NG-RAN overall description; stage 2," 3GPP, Sophia Antipolis, France, Rep. TS 38.300, Mar. 2024.

[16] H. K. Dureppagari, C. Saha, H. S. Dhillon, and R. M. Buehrer, "NTN-based 6G localization: Vision, role of LEOs, and open problems," *IEEE Wireless Commun.*, vol. 30, no. 6, pp. 44–51, Dec. 2023.

[17] "Non-access-stratum (NAS) protocol for 5G system (5GS); stage 3," 3GPP, Sophia Antipolis, France, Rep. TS 24.501, Mar. 2024.

[18] "5G system; access and mobility management services; stage 3," 3GPP, Sophia Antipolis, France, Rep. TS 29.518, Mar. 2024.

[19] "Procedures for the 5G system (5GS); stage 2," 3GPP, Sophia Antipolis, France, Rep. TS 23.502, Dec. 2023.

[20] "NG-RAN; NG application protocol (NGAP)," 3GPP, Sophia Antipolis, France, Rep. TS 38.413, Mar. 2024.

[21] "NR; Physical layer measurements," 3GPP, Sophia Antipolis, France, Rep. TS 38.215, Mar. 2024.

[22] "Numbering, addressing and identification," 3GPP, Sophia Antipolis, France, Rep. TS 23.003, Jun. 2024.

[23] "5G system; common data types for service based interfaces; stage 3," 3GPP, Sophia Antipolis, France, Rep. TS 29.571, Mar. 2024.

[24] "NG-RAN; architecture description," 3GPP, Sophia Antipolis, France, Rep. TS 38.401, Mar. 2024.

[25] "5G system; principles and guidelines for services definition; stage 3," 3GPP, Sophia Antipolis, France, Rep. TS 29.501, Jun. 2024.

[26] A. Pinto, G. Santaromita, C. Fiandrino, D. Giustiniano, and F. Esposito, "Characterizing location management function performance in 5G core networks," in *Proc. IEEE Conf. Netw. Funct. Virtual. Softw. Defined Netw. (NFV-SDN)*, Nov. 2022, pp. 66–71.

[27] "Experimenting with localization management functions in 5G core networks," in *Proc. 28th Annu. Int. Conf. Mobile Comput. Netw.*, Oct. 2022, pp. 806–807.

[28] A. Le Floch, R. Kacimi, P. Druart, Y. Lefebvre, and A.-L. Beylot, "Accurate E-CID framework for indoor positioning in 5G using path tracing and machine learning," in *Proc. Int. ACM Conf. Model. Anal. Simul. Wireless Mobile Syst.*, Oct. 2023, pp. 9–17.

[29] A. L. Floch, R. Kacimi, P. Druart, Y. Lefebvre, and A.-L. Beylot, "Indoor localization in current 5G networks: The way to go," in *Proc. IFIP Netw. Conf. (IFIP Netw.)*, Jun. 2024, pp. 285–293.

[30] D. C. Montgomery and G. C. Runger, *Applied Statistics and Probability for Engineers*, 6th ed. Hoboken, NJ, USA: Wiley, Inc., 2014.

[31] N. Kalis, *Performance Analysis of 5G NRPPa Positioning Procedures*, Zenodo, Geneva, Switzerland, Apr. 2025.

[32] (Raspberry Pi Ltd., Cambridge, U.K.) *Raspberry Pi 5*. Aug. 2024. [Online]. Available: https://datasheets.raspberrypi.com/rpi5/raspberry-pi-5-product-brief.pdf

[33] (Raspberry Pi Ltd., Cambridge, U.K.) *Raspberry Pi Active Cooler*. Apr. 2024. [Online]. Available: https://datasheets.raspberrypi.com/cooling/raspberry-pi-active-cooler-product-brief.pdf

[34] J. Corbet, A. Rubini, and G. Kroah-Hartman, *Linux Device Drivers*, 3rd ed. Sebastopol, CA, USA: O'Reilly Media, 2005.

NICO KALIS was born in Wismar, Germany, in 1998. He received the bachelor's and master's degrees in applied computer science from the University of Wismar, Germany, in 2021 and 2023, respectively. He is currently pursuing the Ph.D. degree with the Institute of Applied Microelectronics and Computer Engineering, University of Rostock, Germany.

His field of research includes positioning, computer networking, and communication especially in cellular networks.

CHRISTIAN HAUBELT received the Diploma degree in electrical engineering from the University of Paderborn, Germany, in 2001, and the Ph.D. degree in computer science and the Habilitation (postdoctoral lecture qualification) degree in computer engineering from the University of Erlangen-Nuremberg, Germany, in 2005 and 2010, respectively. Since 2011, he has been a Professor of Embedded Systems with the University of Rostock, Germany. His research interests include electronic system level design, smart sensor systems, virtual prototyping, and multiobjective optimization.

FRANK GOLATOWSKI (Member, IEEE) received the Dr.-Ing. degree in computer engineering from the University of Rostock, Germany, in 1999. He is a Senior Research Associate and a Lecturer with the Institute of Applied Microelectronics and Computer Engineering, University of Rostock, and the Head of the Research Group on Industrial Informatics. He has published over 200 publications in international conferences and journals. His research interests include embedded service-oriented architectures and Industrial Internet-of-Things for various domains, such as automation and medical technology, with a focus on interoperability, reliability, and real-time capability. He has received several awards, including the ITEA Award of Excellence 2022 and 2023.