Accepted for presentation in: 2022 IEEE Real-Time Systems Symposium (RTSS)

Context-based Latency Guarantees Considering Channel Degradation in 5G Network Slicing

Andrea Nota
Chair of Embedded Systems
TU Dortmund University
Dortmund, Germany
andrea.nota@tu-dortmund.de

Selma Saidi
Chair of Embedded Systems
TU Dortmund University
Dortmund, Germany
selma.saidi@tu-dortmund.de

Dennis Overbeck
Communication Networks Institute
TU Dortmund University
Dortmund, Germany
dennis.overbeck@tu-dortmund.de

Fabian Kurtz
Communication Networks Institute
TU Dortmund University
Dortmund, Germany
fabian.kurtz@tu-dortmund.de

Christian Wietfeld
Communication Networks Institute
TU Dortmund University
Dortmund, Germany
christian.wietfeld@tu-dortmund.de

***Abstract*—Mission critical applications in domains such as Industry 4.0, autonomous vehicles or smart grids are increasingly dependent on flexible, yet highly reliable communication systems. The Fifth Generation of mobile Communication Networks (5G) promises to support critical communications on a single unified physical communication network through a novel approach known as network slicing. We focus in this work on context-based hard performance guarantees by formalizing an analytical method for bounding response times in critical systems. This approach allows to consider different contexts based on models of degradation of channel quality, and avoids a global highly pessimistic worst-case bound computed for worst possible channel conditions. We demonstrate that the proposed method for computing context-based response times guarantees successfully bounds results obtained in realistic mobility scenarios using a machine-learning based 5G simulation framework.**

***Index Terms*—5G, Network Slicing, Latency Guarantees, Channel Degradation.**

I. Introduction

Driven by increased connectivity and available computation capabilities, safety-critical real-time functionalities are today not confined anymore to embedded devices and need to execute over multiple network-connected devices. Wireless communication is currently emerging at the heart of connectivity solutions for supporting communication in safety-critical systems. The Fifth Generation of mobile Communication Networks (5G) is currently an established technology foreseen for use in fields, like industry automation and Vehicle-to-Everything (V2X) communication, where stringent timing and reliability requirements need to be met.

Network slicing [1] constitutes a key enabler technology that allows in 5G to manage different classes of services and their requirements by integrating them into a single physical communication network, where arbitration between different slices can be performed. Network slicing allows to provide a suitable abstraction for physical wireless networks where resource allocation is performed at the slice level. Providing guarantees on the timing behavior of such systems remains however a major challenge that needs to be tackled using formal analysis.

Wireless communication resources offer in fact a unique challenge for providing timing guarantees as they deal with highly dynamic environments where several parameters are unknown at operation time [2]. In addition to data transmission arrival requests and packet sizes, channel conditions play an important role in determining communication latency. Channel conditions define the quality of a radio signal in a wireless link. It depends on the Block Error Rate (BLER) defined as the probability of incorrectly decoding the transport block, and varies based on different macro and micro physical and environmental effects like physical interference, weather conditions, and mobility of users equipment. Reasoning about timing performance of radio access networks resources in order to provide guarantees becomes therefore very difficult. In practice, only empirical methods are used to estimate variations in transmission times without providing any response times guarantees.

In order to provide formal timing guarantees, timing analysis methods [3], [4] are used to bound interference effects and compute timing guarantees on the latency of individual transmissions. Bounding the timing effects of shared resources requires a careful analysis of requests arrivals (inter-arrival times) and a suitable understanding (modeling) of the characteristics of considered resources. In this paper, we aim at using methods classically used in the embedded systems domain for response time analysis to the wireless communication domain which is inherently highly dynamic. Timing analysis approaches assume that information required for the analysis are known at design time, or at least considering worst-case conditions. We believe that we have to break away from the idea of one holistic worst-case bound in systems which are highly dynamic since this can only be highly pessimistic based on worst-case possible operational conditions of the system.

We, instead, introduce latency guarantees based on specific context of operation to better capture dynamics effects, in particular variations of channel conditions.

According to [5], a context is any information that can be used to characterise the situation of an entity. An entity is a person, place, or object that is considered relevant to the interaction between a user and an application, including the user and applications themselves. We consider similarly the notion of context from a user perspective and the parameters that can influence its behavior (in this case, response time on 5G communication link). In classical response time analysis, the context is restricted to applications sharing the same resource, an information which is static or bounded in the worst-case. We extend here the notion of context based on the environment of operation, considering variations of channel conditions. Our *contribution* is summarized as follows,

- We provide in this paper an analytical formulation for response time analysis of 5G wireless communication networks, considering a notion of local guarantees based on the context of operation.
- For that, we extend the classical busy-window formulation considering variations in the quality of the channel, by defining models of degradation of channel conditions reflected in packets transmission times.
- We validate our proposed analytical approach using a real 5G simulation system. The framework builds on machine learning for predicting channel conditions, considering realistic scenarios for Photovoltaic (Smart Grid) system and Electric Vehicle Charging applications, and performs scheduling and response time computations in 5G networks.

II. Related Work

Providing timing guarantees in wireless communication domains is very challenging due to their dynamic properties such as channel conditions, type of application and amount of data that needs to be transmitted. For that reason, the utilization of Machine Learning (ML)-aided scheduling algorithms in the next generation of mobile communication networks becomes more important [6]. The work of [7] provides an overview of different ML concepts focusing on handovers between base stations, while [8] studies the impact of newly introduced shortened Transmission Time Interval (TTI) with the 5G standard and its impact on latencies. Authors in [9] propose a novel approach based on NFV and SDN driven queuing strategies which demonstrates the ability to provide hard service guarantees through a dedicated SDN controller for Management and Orchestration (MANO) that manages individual slices. Authors in [10] follow with giving an optimization of the radio resource management in terms of end-to-end latency through the definition of the so-called Configured Grants (CGs). The proposed Slice-Aware Machine Learning-based Ultra-Reliable Scheduling (SAMUS) system is an Radio Resource Management (RRM) scheduler prototype using Configured Grants (CGs) which aim to minimize latency intensive scheduling requests by pre-allocating radio resources based on prediction of traffic demands and channel conditions by utilizing the ARIMA method [2] as proof of concept.

However, these techniques are empirical since the results are provided through the observation and measures of phenomena rather than from theory and analytical methods. In our work, the channel conditions are given by ML-based predictions but the timing bounds are established through analytical methods based on contexts. In fact, the network under 5G is expected to be completely context aware [11]. For any given device, the network is continuously aware of its individual location, features, its surroundings and environment. This includes information regarding all the devices present in its neighbourhood and their capabilities. Regarding that, 5G system will need to be context-aware that uses context information in a real-time mode depends on network, devices, applications, and the environment of users' with the objective of providing high quality of service (QoS). Authors in [12] provide a potential architectural solution for mission-critical context-aware collaboration which has to adhere to strict timing deadlines and ultra-low latency data transmissions. The work in [13] is another example of context exploitation, but used to reduce global energy consumption. The authors propose a detailed context architecture and framework for context based scheduling algorithms which can exploit any context information related to user's equipment and eNodeB to achieve the desired goals based on the proposals for 5G.

In order to provide guarantees, network slicing is seen as key enabler in 5G communication systems, as it is capable of providing guaranteed resources to the tenants of a virtualized network slice. The technologies which mark the basis for network slicing can be grouped in Software-Defined Networking (SDN) and Network Function Virtualization (NFV) [14]. The work in [15] demonstrates promising results utilizing Software-Defined Networking (SDN) in a wired testbed to provide hard service guarantees based on the network calculus approach, laying the foundation for further research within wireless networks using the concept of network slicing [16]. Approaches such as [17]–[19] propose an analytical formulation for computing bounds on latencies in wireless communication systems. In [18], authors provide an analysis for bursty traffic in proportional fair scheduling algorithms in Orthogonal Frequency-Division Multiple Access (OFDMA)-based wireless systems, using M/M/1 queuing modelling for each user. In [17] an analytical Markov model is proposed to characterize resource sharing in wireless communication networks. In [19] an analytical model based on the busy-window formulation [20] for bounding response times in 5G network slicing was also proposed. The model however focuses solely on variations of packet sizes assuming that channel conditions are always stable. We propose in this paper an alternative context-aware method for providing timing guarantees in 5G network slicing based on the busy-window approach. It relies on event arrival curves, similarly to network calculus, and computes considering initial channel conditions and models of degradation, for different contexts, a bound on the accumulated delay due to interference and variations of channel quality.

III. 5G Network Slicing: Background

In the following, we provide background information on 5G resources characteristics and network slicing, relevant to the contribution of this work.

A. Radio Access Resources in 5G

Wireless communication resources are radio frequency waves transmitted using subcarriers which can be multiplexed considering multiple frequency and time domains, following the classical OFDMA. A Resource Element (RE) is therefore the smallest time-frequency resource (i.e., one OFDM symbol) which consists of one subcarrier modulated over time. A Resource Block (RB) is a group of subcarriers contiguous in frequency over symbol in time. The granularity of a RB in terms of number of RE varies based on the channel bandwidth and subcarriers spacing. Specifically, in 5G, one RB contains 12 subcarriers in frequency domain similar to LTE. In LTE resource block bandwidth is fixed to 180 KHz but in NR it is not fixed and depend on subcarrier spacing. For the sake of simplicity, we consider in the rest of the paper, similarly to [10], that resources blocks are the basic resource that can be allocated to a given application. The number of bits that can be transmitted within a resource block is not fixed but rather depends on the used modulation scheme.

B. Modulation Coding Scheme (MCS)

For any wireless communication scheme, MCS determines the amount of bits being transmitted per symbol or resource element. The MCS is typically adjusted by the base station and depends heavily on the quality of the received radio signals on the channel. A better quality leads to higher MCS and therefore more useful bits that can be transmitted within a symbol. The base station monitors periodically channel conditions and modifies the applied MCS when required, which leads to changes in the available data rate. 5G New Radio (NR) supports several modulation schemes, namely QPSK, 16 QAM and 256 QAM. While 2 bits can be transmitted per RE considering QPSK, 4 bits can be transmitted considering 16 QAM and 8 bits considering 256 QAM respectively. Data rate for every transmission depends therefore on the ratio between useful bit and total transmitted bits.

C. Network Slicing and Resource Allocation

Network slicing views resources in 5G as a grid of multiple resource blocks, each block is two dimensional and corresponds to an allocation in the radio frequency and time domains, see Fig 1. Based on the size of transmissions, the latency criticality of the slice and the modulation scheme, the network slicing scheduler allocates dynamically the required number of resource blocks to be used by a given application. Whenever wireless channels condition changes (i.e., improves or deteriorates), the modulation scheme is modified thereby leading to new allocation of Resource blocks.

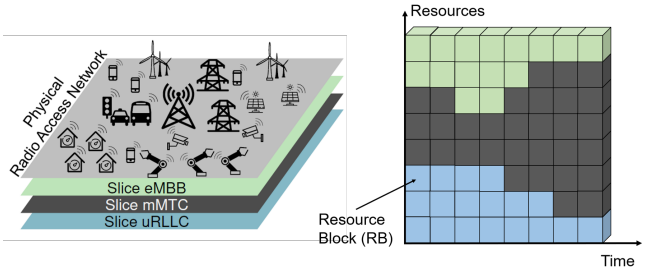

Fig. 1: Overview of the 5G slicing layers and possible resource allocation of the 5G grid for slices.

IV. Assumptions and System Model

We consider a system with a set of slices $S = \{s_1, ...s_l\}$ that requests simultaneously the 5G resource grid RG_{5G}. We consider, similarly to [19], that each slice is assigned a priority value to reflect its criticality. Priority values are assigned to slices in a descendent order according to their criticality level (i.e., higher criticality levels are denoted with lower priority values). The dynamics of data transmissions are modeled using event-arrival functions where an event refers to a transmission (composed of one or multiple packets) through the 5G wireless network. Mechanisms as packet retransmission are not taken into account for the purpose of our work since we focus on how the channel quality influences timing guarantees and not on the correct reception of each packet. In the following, we define applications characteristics and detail specificity of 5G resources and our defined channel conditions relevant context.

A. Applications characterization

Definition 1 (Slice): A slice $s_i = (\alpha_i, N_{UEi}, T_i)$ is an application with a given criticality level. Based on criticality, a slice is assigned a *statically fixed* priority level α_i. Every slice s_i has a number N_{UEi} of User Equipments (UEs) executing the same application and performing data transmissions with possibly a different periodicity. T_i is the aggregated sequence of all transmissions performed by UEs within a slice i. Note that all UEs belonging to the same slice have the same priority α_i. We do not distinguish between packets from different UEs within the same slice.

Definition 2 (Data Transmissions): Data transmissions $T = \{e_1, ..., e_n\}$ are defined as a sequence of events. Every event $e_k = (t_k, w_k)$ is a transmission request defined as the time t_k where the request is activated and a workload w_k which corresponds to the size of the packet to be transmitted by request e_k. Note that for each slice i, all data transmission requests inherit the priority level from their corresponding slice.

We use event models commonly used to model tasks activation in real-time analysis methods like real-time calculus or compositional performance analysis [20] to bound the arrival time of data requests.

Definition 3 (Packets Arrival Models): Event models are used to characterize for every slice i the arrival of data transmissions. They are defined using the function $\eta_i^+(\Delta t)$

which denotes for every slice i the maximum number of transmissions issued within a time window Δt. The inverse function $\delta_i^-(n)$ denotes the minimum time interval between the first and last transmission in any sequence of n transmissions from slice i. For a strictly periodic (or sporadic with minimum inter-arrival time) arrival of streams from UEs of slice i with a period P_i, $\eta_i^+(\Delta t)$ is defined as follows,

$$\eta_i^+(\Delta t) = \lceil \frac{\Delta t}{P_i} \rceil \tag{1}$$

B. Resources Characterization and Context Definition

As mentioned previously, 5G is composed of wireless communication resources as radio frequency waves multiplexed in the frequency and time domain.

Definition 4 (5G Resource Grid): A 5G resource grid RG_{5G} is defined as a matrix of $n \times m$ Resource Blocks (RBs). A RB represents the smallest unit to be allocated to the system. The total number of RBs in the resource grid is limited by maximal channel bandwidth and subcarriers spacing (or numerology [21]).

Definition 5 (Channel Condition and Modulation Scheme): Let MCS be the value of the current modulation and coding scheme. Note that the value of MCS does not change continuously but based on defined thresholds on the values of Block Error Rate (BLER) and corresponding acceptable signal to noise ratio (SNR)[1] for each modulation scheme. We bound this change considering d_{mcs}^- as a minimum time between two BLER thresholds and therefore a change in two values of MCS.

Definition 6 (MCS Data Rate): Let b_{mcs} be the available bandwidth based on the selected modulation and coding scheme. Calculations are standardized following the 3GPP 38.214 (chapter 5.1.3.2)[2], involving modulation scheme, subcarrier spacing configuration, and the number of resources symbols. To give a numerical example, considering 106 RBs and an MCS = 14, which is a middle value of channel quality, we are able to transmit 59432 bits in 1 ms.

Definition 7 (Packet Latency): The execution time (i.e., transmission latency) C of a packet e_k is defined as the ratio between the workload w_k of the packet and the data rate b_{mcs} for a given MCS.

$$C_k = \frac{w_k}{b_{mcs}} \tag{2}$$

1) Characterizing Variations of Channel Conditions: In a dynamic environment such as wireless communication, we focus on parameters relative to changes in channel conditions that influence response time. We consider changes in channel conditions as changes reflected in the value of MCS[3]. It determines the bandwidth available expressed as the amount of bits which can be transmitted in a slot of 1 ms, which corresponds to the subframe length of a packet in 5G. A lower MCS value will lead to less bandwidth available for transmission. Note that the transmission of one packet can be longer than d_{mcs}^- to witness different variations of MCS values and therefore available bandwidth. Degradation of channel conditions have particularly a large effect on the response time. In the following we define models of degradation of channel conditions that will be later incorporated in response time analysis.

Definition 8 (Variations of Channel Conditions): The base station monitors periodically (every d_{mcs}^-) the channel conditions to adapt the MCS value whenever required. Since we do not consider continuous changes in channel qualities, but rather changes in the value of MCS at every sensing period d_{mcs}^-, we represent channel variations in a discrete form. Let $X_{mcs} = [mcs_1, ..., mcs_n]$ be a vector of MCS values reflecting variations in channel conditions. Note that for any $\{(i,j), j \geq i\}$, the time Δt separating variations in mcs $(X_{mcs}[i], X_{mcs}[j])$ is $\Delta t = (j-i) * d_{mcs}^-$.

Definition 9 (Degradations of Channel Conditions): Channel quality can improve or deteriorate over time based on many micro and macro environmental factors (e.g., weather conditions, physical channel interference and noise), thereby leading to an improvement or deterioration of available bandwidth. Given a value mcs_i in the vector X_{mcs}, a degradation is defined as an immediate decrease in the value of MCS, that is, $\exists i$, s.t. $X_{mcs}[i+1] < X_{mcs}[i]$, or as a stabilization of the value of mcs after a previous decrease of MCS value. In this case, degradation is defined as the absence of improvement after a previously occurring degradation, thereby leading to a stationary state of degradation. That is,

$$\begin{cases} \exists_{i',\ (i'<i)} \text{ s.t. } X_{mcs}[i'+1] < X_{mcs}[i'] \\ \forall_{k,\ (i'+1 \leq k < i)} \text{ s.t. } X_{mcs}[k+1] = X_{mcs}[k] \end{cases}$$

Improvements can be defined in a similar fashion considering increase of the value of MCS.

Definition 10 (Models of Degradation of Channel Conditions): The number and distribution of occurrences of degradation has a great impact on latency and response time. Due to the limitation of 5G data rates and variations of operational environment (e.g., network coverage when considering mobility), there is usually never a continuous degradation of channel conditions. That means that degradations are always followed (after some time) by improvements. We therefore define models of variations that consider cycles of degradations and improvements of different lengths. Let $M = \{(m,\mu),(k,\xi)\}$ be a model of variation of channel conditions where (m,μ) defines the number of m *subsequent* and μ *stationary* degradations, and (k,ξ) defines the number of k *subsequent* and ξ *stationary* improvements. Note that we need to distinguish between subsequent and stationary degradations and improvement because they will have an impact on the response time. As detailed in the evaluation section, if we consider the same initial channel condition, and an overall number of degradations, for example 5, having only 5 subsequent degradation will have a worse response

[1] 3GPP TS 38.104 V16.6.0 Section 8.2.6

[2] https://5g-tools.com/5g-nr-tbs-transport-block-size-calculator/

[3] We do not consider detailed physical models affecting the quality of channel signals but rather consider abstractions of changes in channel conditions considering the value of MCS.

time than one subsequent degradation followed by 4 stationary degradations.

Definition 11 (Rate of Change): Let ρ be the rate of change defined as the difference between consecutive MCS values, $\rho_{i,j}$ is defined as follows,

$$\rho_{i,j} = \frac{|b_{mcs}(j) - b_{mcs}(i)|}{b_{mcs}(i)} \tag{3}$$

For the sake of simplicity, we consider in the following a maximum ρ^+ and minimum ρ^- rate of change of MCS value, that is $\rho^+ = \{\forall_{i,j},\ max(\rho_{i,j})\}$ and $\rho^- = \{\forall_{i,j},\ min(\rho_{i,j})\}$. A relevant property is that channel conditions usually deteriorate or improve *gradually*, thereby leading to a gradual change in the value of MCS, that is ρ^+ and $\rho^- \leq 1$. Note that for computing the worst-case, ρ^+ is considered for degradations and ρ^- for improvements.

2) Context-based Variations: A context in wireless communication networks generally refers to the context of operation of the network including individual locations, features, surrounding and environment. Contexts and operation conditions can change over time. Since we focus on channel degradations and their influence on the response time, we consider as context features from the environment and operation conditions that reflect variations in channel conditions.

Definition 12 (Context): A context $\Phi_{[t,t+\Delta t)} = \{mcs_{init}, (\rho^+, \rho^-), M_\phi\}$ is defined over an interval of time $[t, t+\Delta t)$ where variations of channel conditions are spanning over multiple periods d^-_{mcs} of MCS changes. It is defined with an initial MCS value mcs_{init} at time t reflecting initial channel conditions in that context, a maximum and a minimum rate of change ρ^+ and ρ^-, respectively, and a model of degradation $M_\phi = \{(m_1, \mu_1), (k_1, \xi_1), ..., (m_n, \mu_n), (k_n, \xi_n)\}$ reflecting several possible cycles of degradations and improvements in that context over $[t, t + \Delta t)$.

Definition 13 (Global Worst-Case Guarantees): Global worst-case response time guarantees can be computed considering possible static worst-case channel conditions and variations over all contexts, that is $\forall [t, t + \Delta t)$, worst-possible initial conditions $mcs_{init} = min(\{mcs\})$[4], and maximum number of degradations $max(\{(m, \mu)\})$ followed by a minimum number of improvements $min(\{(k, \xi)\})$.

Definition 14 (Context-Based Worst-Case Latency Guarantees): Global worst-case guarantees can be highly pessimistic. We define therefore context-based worst-case guarantees. Given a context $\Phi_{[t,t+\Delta t)}$, it provides worst-case response times bounds in that context considering mcs_{init} and M_ϕ.

V. Context-based Response Time Analysis

Changes in channel conditions influence the modulation and coding scheme (mcs) applied by the base station to the current transmission and consequently, it leads to an increase or decrease in the data rate. For the purpose of our analysis to bound the worst-case response time, we reflect changes in mcs on changes in the transmission time of individual packets based on defined models of degradation of channel conditions. Note that two packets with the same size can have different transmission times depending on the modulation scheme.

As stated in the previous section, the number and also distribution of degradation of channel conditions reflected in the mcs value and defined models of degradation have a great impact on latency and response time. In the following, we first explain how the different models of degradation influence latency for individual packets. We later provide analytical bounds on the worst-case latency of each slice performing q transmissions depending on context-based varying conditions. We define the worst-case response time, first in isolation and then considering other higher-priority interfering slices.

A. Basic Slice Latency Bound

Given a context $\Phi_{[t,t+\Delta t)}$, we are interested in bounding the response time of slices performing transmissions in that context, considering variation in channel conditions and degradation effects. Let us first consider one slice in isolation performing q packets transmission. Depending on the load of packets (i.e., packets size), their transmission time can be longer than d^-_{min}. Packets can therefore witness during transmission multiple variation of channel conditions, compared to the initial mcs_{init} value where the transmission of packets has started. In our context, this results in a change in the execution time of the packet which should be notified to the analytical tool for the next busy window computation.

Let $C'_\Phi(q)$ be the transmission time of q packets from slice i in context Φ. It is defined as follows,

$$C'_\Phi(q) = C^t_\Phi(q) + \Delta C^+_\Phi(q) - \Delta C^-_\Phi(q) \tag{4}$$

where, $C^t_\Phi(q)$ is the transmission time of q packets considering initial mcs value mcs_{init} at time t for the entire transmission of q packets (i.e., assuming that initial channel condition does not change), $\Delta C^+_\Phi(q)$ is the delay of transmission considering degradation of channel condition in that context, $\Delta C^-_\Phi(q)$ is the reduced transmission time due to an improvement of channel condition. In the following, we detail each term of the equation.

a) Stable Channel Conditions: We refer to $C^t_\Phi(q)$ as the *nominal transmission time* of q packets considering that channel conditions, starting from initial mcs value does not vary during the entire transmission of packets. It is defined as follows,

$$C^t_\Phi(q) = \frac{\sum_{k=0}^{q}\ w_k}{b_{init}} \tag{5}$$

where, $\sum_{k=0}^{q}\ w_k$ is the overall transmission load of q packets. Transmission time C^t_Φ is directly derived using the ratio between load and bandwidth (ref. Eq 2).

Starting from initial channel conditions at time t, variation in channel conditions over $[t, t + \Delta t)$ can lead to an increase of transmission time in case of degradation or decrease due to improvement. Note that variation of channel conditions are directly reflected in the available bandwidth. For a fixed load of q packets, transmission time therefore decreases and increases proportionally to bandwidth.

[4]Note that smaller MCS values lead to lower available bandwidth.

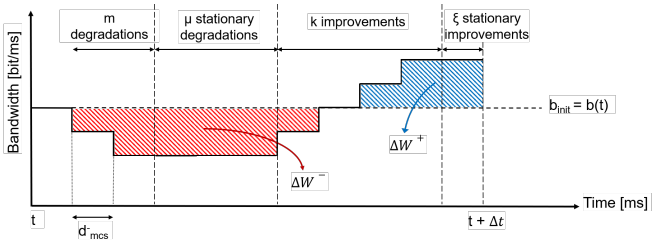

Fig. 2: Example of a context $\Phi_{[t,t+\Delta t)}$ with (m, μ) degradations and (k, ξ) improvements. Illustration of the effect on decreased and increased bandwidth compared to initial mcs value and bandwidth b_{init}. Note that in this simple example, ρ^+ and ρ^- have a the same value.

b) Effect of Degradations: Considering (m, μ) degradations of initial channel conditions in context Φ and (k, ξ) improvements during the transmission of packets, the nominal execution time $C^t_\Phi(q)$ is delayed by a factor of $\Delta C^+_\Phi(q)$, defined as follows,

$$\Delta C^+_\Phi(q) = \frac{\Delta W^-}{\beta^+_{deg}} \tag{6}$$

where, ΔW^- is the accumulated load that cannot be send due to degradation (compared to the nominal case) and β^+_{deg} is the highest possible achievable bandwidth at the end of (m, μ) degradation. $\Delta C^+_\Phi(q)$ is defined considering a ratio between overall accumulated workload and the achievable bandwidth at the end of the degradation phase, where improvements occur.

The accumulated load ΔW^- is directly derived as follows, as a product of time units and reduced bandwidth compared to b_{init}.

$$\Delta W^- = d^-_{min} * \ (\Delta\beta_{deg} + \Delta\beta_{deg,imp}) \tag{7}$$

It is defined considering 2 phases: a degradation phase resulting in $\Delta\beta_{deg}$ where bandwidth is decreasing compared to b_{init}, followed by an improvement phase resulting in $\Delta\beta_{deg,imp}$ where bandwidth is increasing but is still below or equal to b_{init}, see (red area) in Fig 2. Since increase or decrease of bandwidth occurs at every d^-_{min}, it is sufficient to multiply difference in bandwidth by time d^-_{min} to derive workload.

During the degradation phase, the difference $\Delta\beta_{deg}$ in bandwidth (compared to b_{init}) is derived as follows,

$$\begin{aligned}\Delta\beta_{deg} = &\sum_{\lambda=0}^{m}(b_{init} - \lambda * \rho^+ * b_{init}) \\ &+\mu * (b_{init} - m * \rho^+ * b_{init})\end{aligned} \tag{8}$$

considering m subsequent degradations. After every d^-_{min} a new degradation occurs, every new degradation leads to a $\rho^+ * b_{init}$ decrease of bandwidth value[5]. After m subsequent degradations, additional μ stationary degradations can occur where bandwidth is stable at $(b_{init} - m * \rho^+ * b_{init})$ for μ periods of d^-_{min}.

Degradations are followed by (k, ξ) improvements. Improvements occur as well gradually after every d^-_{min}, leading to an additional phase with difference $\Delta\beta_{deg,imp}$ defined as follows,

$$\Delta\beta_{deg,imp} = \begin{cases} \sum_{\lambda=0}^{m}(b_{init} - \lambda * \rho^- * b_{init}) & \text{if } k \geq m \\ \sum_{\lambda=0}^{(m-k)} \ (b_{init} - \lambda * \rho^- * b_{init} + \\ \xi * (b_{init} - (m-k) * \rho^- * b_{init})) & \text{otherwise} \end{cases} \tag{9}$$

where increased bandwidth is still below or equal to b_{init}. We distinguish between the case $k \geq m$ where b_{init} is reached after m out of k gradual improvements (in terms of load, this phase is symmetric to the first phase of m degradation), and the case $k < m$ followed by ξ stationary state of improvement, where the bandwidth remains below b_{init}.

As mentioned previously, the accumulated load ΔW^- is sent at the end of the degradation phase with highest achievable bandwidth β^+_{deg}. It is defined as follows,

$$\beta^+_{deg} = \begin{cases} b_{init} & \text{if } k \geq m \\ b_{init} - [(m-k) * \rho^+ * b_{init}] & \text{otherwise} \end{cases} \tag{10}$$

where, depending on the number of following improvements k, the bandwidth can reach again b_{init} when $k \geq m$, or stay at a lower bandwidth level $(m-k) * \rho^+ * b_{init}$ when $k < m$.

c) Effect of Improvements: Considering (k, ξ) improvements of initial channel conditions during the transmission of q packets, the nominal execution time is decreased by a factor of $\Delta C^-_\Phi(q)$, defined as follows,

$$\Delta C^-_\Phi(q) = \frac{\Delta W^+}{\beta^-_{imp}} \tag{11}$$

where, ΔW^+ is the additional accumulated load that can be send due to improvements (compared to the nominal case) and β^-_{imp} is the lowest possible achievable bandwidth at the end of (k, ξ) improvements.

Similarly to the effects of degradation on bandwidth, the accumulated load ΔW^+ can be defined as follows,

$$\Delta W^+ = d^-_{min} * \Delta\beta_{imp} \tag{12}$$

where, $\Delta\beta_{imp}$ is the difference in increased bandwidth compared to b_{init}. Note that this case assumes that there are more improvements than degradations (i.e., $k \geq m$), see (blue area) in Fig 2.

During the improvement phase beyond b_{init}, the difference $\Delta\beta_{imp}$ in bandwidth (compared to b_{init}) is derived as follows,

$$\begin{aligned}\Delta\beta_{imp} = &\sum_{\lambda=1}^{(k-m)} (\lambda * \rho^- * b_{init} - b_{init}) + \\ &\xi * ((k-m) * \rho^- * b_{init} - b_{init})\end{aligned} \tag{13}$$

considering $(k-m)$ subsequent improvements after b_{init}. Similarly to degradations, a new improvement occurs after

[5]We bound uniformly the rate of change (i.e., increase or decrease) of bandwidth considering maximum rate of change ρ^+.

TABLE I: Overview of all notations and respective description used in Section IV and V.

Parameter	Description
$\eta_i^+(\Delta t)$	Maximum number of transmissions issued within a time window Δt for slice i
d_{mcs}^-	Minimum time between a change in two values of mcs
ρ	Rate of change defined as the difference between consecutive mcs values
M	Model of variation of channel conditions where (m, μ) defines the number of m *subsequent* and μ *stationary* degradations, followed by (k, ξ) defines the number of k *subsequent* and ξ *stationary* improvements
$\Phi_{[t,t+\Delta t)}$	Context defined with an initial mcs value mcs_{init} at time t , a maximum rate of change ρ^+ and a model of degradation M_ϕ
$C'_\Phi(q)$	Transmission time of q packets from slice i in context Φ
$C^t_\Phi(q)$	*Nominal transmission time* of q packets considering that channel conditions do not vary during the entire transmission of packets
$\Delta C^+_\Phi(q)$	Delay derived by the accumulated load that cannot be send due to degradations
β^+_{deg}	Highest possible achievable bandwidth at the end of (m, μ) degradations
ΔW^-	Accumulated load that cannot be send due to channel degradation (compared to the nominal case)
$\Delta\beta_{deg}$	Difference in bandwidth (compared to b_{init}) during the degradation phase
$\Delta\beta_{deg,imp}$	Difference in bandwidth (compared to b_{init}) during improvement phase but is still below or equal to b_{init}
$\Delta C^-_\Phi(q)$	Time recovered due to channel improvements
$\omega^+_{i,\Phi}(q)$	Worst-case response time in a given context $\Phi_{[t,t+\Delta t)}$ required to perform q transmissions from slice i in the presence of interfering higher priority slices j
γ_i	Blocking time due to lower priority slices, defined as 1 ms

every d^-_{min}, leading to a $\rho^- * b_{init}$ increase of bandwidth value. After $k-m$ subsequent improvements after b_{init}, additional ξ stationary improvements can occur where bandwidth is stable at $((k-m) * \rho^- * b_{init})$ for ξ periods of d^-_{min}.

The additional accumulated load ΔW^+ can be sent at β^-_{imp} as the lowest possible achievable bandwidth at the end of (k, ξ) improvements. It is defined as follows,

$$\beta^-_{imp} = b_{init} + ((k-m) * \rho^- * b_{init}) \tag{14}$$

where, the bandwidth increase reaches $(k-m) * \rho^- * b_{init}$ improvements after b_{init}.

d) Multiple Variations of Channel Conditions in Context: For the sake of simplicity, we considered so far that a context Φ contains one cycle of degradations and improvements, that is $M_\phi = \{(m, \mu), (k, \xi)\}$. More generally, if we have multiple cycles of degradations and improvements $M_\phi = \{(m_1, \mu_1), (k_1, \xi_1), ..., (m_n, \mu_n), (k_n, \xi_n)\}$ in context Φ (i.e., subsequent red and blue areas in Fig 2), the definition of $C'_\Phi(q)$ in Eq 4 is extended as follows,

$$C'_\Phi(q) = C^t_\Phi(q) + \sum_{\lambda=1}^{n} [\Delta C^+_\Phi(q) - \Delta C^-_\Phi(q) \mid (m_\lambda, \mu_\lambda), (k_\lambda, \xi_\lambda)] \tag{15}$$

where, the sum of delays and reduced transmission times can be computed based on every degradation and improvement cycle. Note that every cycle can have different values of $m, \mu, k,$ and ξ.

B. Context-based Worst-case Response Time

Slicing allows to map resource blocks to applications which are then called slices. Based on that, resource sharing and therefore timing interference occurs. Interference effects lead to a delay for lower-priority slices. Therefore, when multiple slices (i.e., applications) are active at the same time with high packet loads, the 5G grid is not sufficient to serve simultaneously all slices. We consider, similarly to [19], that slices have different priorities reflecting different levels of criticality or importance, and that the scheduler allocates resource blocks in the 5G grid first to higher priority slices and then to slices with lower priority, thereby leading to an increase in response time. We consider in the following the 5G grid as a single resource and rely on the busy-window approach to bound the response time of each slice in the presence of other interfering slices.

The busy-window was first introduced in [22], to bound the maximal time interval during which a task is "busy" processing an event. It was later extended in [23] to multiple event busy-window, which constitutes the maximal time required to process q activations from a given task. Let $\omega^+_i(q)$ be the q-event busy windows of a slice i that describes the maximum time interval required to complete the transmission of q consecutive packets considering network slicing under static priority preemptive scheduler.

The busy-window is based on a careful estimation of tasks execution times (usually considering worst-case execution time) as inputs for the response time analysis. As mentioned previously, providing global worst-case time guarantees considering worse possible channel conditions will lead to highly pessimistic results. Therefore, we chose instead to bound, given an operational context, the response time considering corresponding initial channel condition and degradation model. Given the busy-window, we assume at the start of every context critical instant where all slices are activated simultaneously and accounting for maximum interference from higher priority slices. Note that we chose the standard busy-window formulation where we reflect the notion of contexts in worst-case execution times of individual slices as defined in the previous section.

Let $\omega^+_{i,\Phi}(q)$ be the worst-case response time in a given context $\Phi_{[t,t+\Delta t)}$ required to perform q transmissions from slice i in the presence of interfering higher priority slices j.

It is defined as follows,

$$\omega^{+}_{i,\Phi}(q) \leq \gamma_i + C_{i,\Phi}(q) + \sum_{\forall j \in hp(i)} C_{j,\Phi}(\eta_j(\omega^{+}_{i,\Phi}(q))) \quad (16)$$

where, $\gamma_i = TTI$ is the blocking time due to lower priority slices, defined as 1 ms, which corresponds to a Subcarrier Spacing (SCS) of 15 kHz within the NR specification. $C_{i,\Phi}(q)$ is the execution time of slice i in isolation given the context Φ as defined in the previous section, and $C_{j,\Phi}(\eta_j(\omega^{+}_{i,\Phi}(q)))$ is the delay due to the transmission of $\eta_j(\omega^{+}_{i,\Phi}(q))$ packets from other interfering slices in context Φ.

a) Discussion: Reasoning about specific contexts for response time analysis requires deriving conditions on the convergence of the busy-window and the size of required interval $[t, t + \Delta t)$ used to define a context.

1) The busy-window is a fixed point computation, where results from one computation iteration are used for computing the transmission latencies in the next iteration. Since execution times per packet vary in an increasing or decreasing order based on the channel conditions, the load may be too high in the considered interval and iterations may never reach a fixed-point. Intuitively, starting from a schedulable context at initial mcs and b_{init} at time t, a sufficient condition on schedulability is that the additional gain of time due to improvements should be more or equal to the delay due to degradations.
2) For the sake of simplicity, we assumed that a context is defined on an interval $[t, t + \Delta t)$ sufficiently large (i.e., known models of degradation over a sufficiently large interval) so that execution starts and completes in that interval. Defining an interval length can be based first on execution time $[t, t + C_{i,\Phi}]$ as an initial value and is then extended with the iteration of the busy-window computation.

VI. Experimental Evaluation

The experiments section is divided in two parts. We firstly study the behavior of our proposed analysis through multiple experiments focused on deeply understanding how the Worst-Case Response Time (WCRT) computation is influenced by variations in channel conditions. Successively, we validate the proposed analysis considering realistic use cases and a comparison with the SAMUS simulation framework [10] of 5G network slicing for providing the latency of every packet. On both experiments we focus only on the latency of packets from the lower priority slice since the response time of packets from the highest priority slice is equal to their basic execution time, considering no timing interference from lower priority slices causing additional delays. In fact, when multiple slices are active at the same time and request to send packets, resource blocks are first allocated to higher priority slices and then lower priority ones. This leads to timing interference from higher priority slices reflected in transmission delays for lower priority ones. The proposed busy-window based formal analysis for bounding the response time of lower priority slices is implemented using pyCPA [20] tool for the worst-case response time computation. In the following, we focus mainly on the effects of channel quality on the latency, and the delay introduced by the interfering slice considered in the analysis.

TABLE II: Overview of the experiment settings.

Packet Size Slice$_1$	4100 Bytes
Packet Size Slice$_2$	2000 Bytes
Number of UEs for Each Slice	1
Channel Quality	Changing every 1 ms

A. Effects of Channel Condition Variations

In the first set of experiments, we analyze the impact of channel quality on the latency bounds of the lowest priority slice computed by the proposed context-based worst-case response time analysis. We observe how the WCRT varies depending on the number and distribution of occurrences of degradation by considering multiple models M of variations of channel conditions. In Table II, considered parameters are reported. After fixing the packet size for Slice$_1$ (lowest priority) and Slice$_2$ (highest priority) to 4100 and 2000 Bytes, respectively, we assume that the channel quality varies every 1 ms. We compute the worst-case response time of a sequence of multiple transmissions (20 packets in this case). We investigate the effect of channel conditions on the WCRT by considering the following different variants of contexts configurations that are comparable,

1) Same initial channel conditions and fixed number of degradations and improvements. We vary the distributions (i.e., position or interleaving) of the number of degradations and improvements in the interval. The two extreme cases occur when all degradations appear as bursts, at the beginning or end, of the transmission interval.
2) Multiple initial channel conditions with a fixed number of improvements and variable number of degradations of initial channel conditions.
3) Multiple initial channel conditions with a fixed number of degradations and variable number of improvements separating two bursts of degradations of initial channel conditions.

We detail in the following obtained results.

1) Position of the Channel Degradations: We consider an initial MCS value of 13, in addition to a fixed number of 3 degradations and 5 improvements. We test all different permutations M of a total of 3 occurrences of degradation and 5 improvements.The first extreme case occur when all degradations of the initial mcs value happen at the beginning of the interval followed later by improvements. In the second case, all improvements of the initial mcs occur at the beginning of the interval followed by all degradations occurring at the end. As mentioned previously, for all these permutations, the number of degradations and improvements remains the

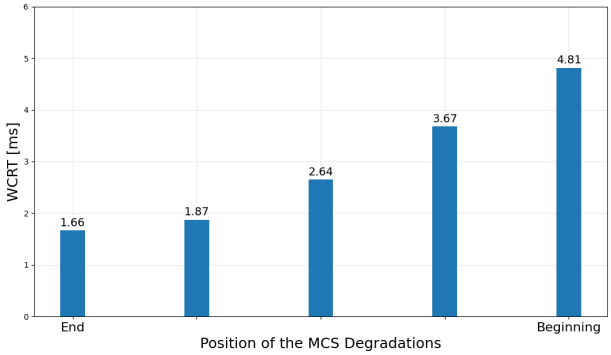

Fig. 3: Worst-Case Response Time (WCRT) considering different distribution of the mcs degradations. From left to right, all the degradations are positioned and then moved gradually from the end to the beginning of the considered $[t, t + \Delta t)$.

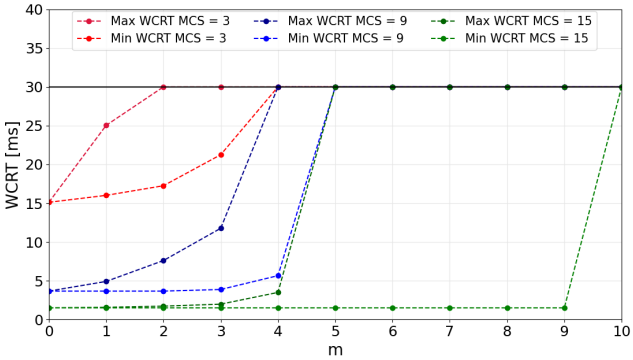

Fig. 4: Worst-Case Response Time (WCRT) for different mcs_{init} and number of degradations m. Min (resp. Max) WCRT corresponds to all m degradations occuring at the end (resp. beginning) of the transmission interval.

same. Only the distribution of improvements compared to degradations plays therefore a role. We also consider the same rate of change for each occurring degradation (resp. improvement).

Results are reported in Fig. 3, where we observe that the computed bound on the response time (WCRT) is higher for the configuration where all degradations happen at the beginning followed then by all improvements. This behavior is explained by considering that the sooner the mcs starts decreasing, the less bandwidth is available for transmissions, which leads to a higher accummulated backlog that can only be transmitted when improvements occur and would therefore require later more time for transmission. As we interleave degradations and subsequent improvements, the worst-case response time decreases as the accumulated backlog is reduced whenever there is an improvement of channel conditions leading to a higher bandwidth available for transmission of current packets and backlog.

2) Amount of Channel Degradations: In the second experiment, we compute WCRT for multiple contexts defined with different mcs_{init}. We test as previously for each context the effect of number of degradations. For that, we set $M = \{(m, 0), (10-m, 0)\}$. We consider only the two extreme cases of distribution of degradations (i.e., all degradations at the beginning or end of the transmission interval). We refer to both cases respectively as max possible WCRT and min possible WCRT in this context, based on the number and distribution of possible degradations. The results are depicted in Fig. 4.

We observe that the latency bound depends on mcs_{init} and the number of degradations m. The lower is the value of mcs_{init}, and the higher is the value of WCRT since smaller bandwidth is associated with smaller values of mcs. Similarly, as m increases, the number of degradations of channel conditions across the interval increases, leading therefore to longer time periods before an improvement of channel condition is observed. This leads consequently to higher WCRT values. Note that computations are limited by the value 30ms expressing a non-computable bound for the response time. In

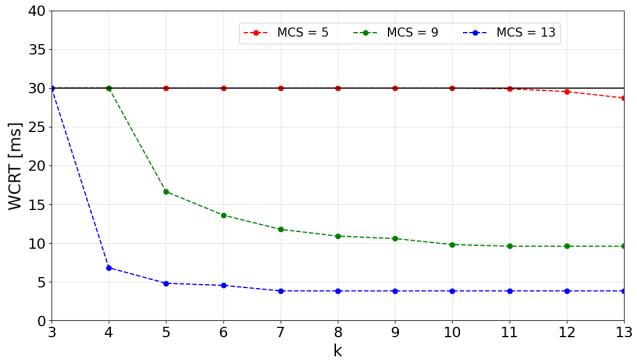

Fig. 5: Worst-Case Response Time (WCRT) for different mcs_{init} and number of improvements k.

this case, no guarantees can be provided. This occurs due to limitations of the analysis and used pyCPA tool. Indeed, the analysis and the tool are limited by schedulability conditions (i.e., non schedulable contexts) in scenarios where mcs value is very low, meaning that individual packets have a very high transmission time due to very poor channel conditions, leading to a full utilization of the channel.

3) Distance Between a Set of Channel Degradations: For the last set of experiments, we investigate the effect of variable number of improvements separating 2 bursts of degradations as this has additionally an effect on latency. For that, we set $M = \{(3, 0), (k, 0)\}$ which corresponds to change the distance between 2 different burst of degradations. As before, we apply the context-based analysis considering three different mcs_{init} as illustrated in Fig. 5. In this case, a WCRT value of 30 ms is observed as the limit of computable WCRT given pyCPA and the schedulability condition. Note that computable WCRT for lower mcs values depends on the number of subsequent improvements which then improves the utilization of the channel with increased bandwidth due to improvements. For example, for $mcs_{init} = 5$, 11 consecutive improvements are at least required to satisfy the schedulability condition and compute the WCRT. By increasing mcs_{init}, the effects of number of improvements for utilization is reduced and the

WCRT can be computed for lower values of k.

Note that, in the last 2 experiments, we consider similar initial channel conditions mcs_{init}. Results from Fig. 4 and Fig. 5 are however not comparable since experiments consider different contexts where the focus changes w.r.t. the considered model of degradations.

B. Validation of Analysis Results

In the following, we consider validation of the proposed context-based worst-case analysis against a 5G simulation framework and considering data of variations of channel conditions and application transmission requests from realistic use case.

1) The SAMUS 5G Network Slicing Simulation Framework: The framework depicted in Fig. 6 and developed in [10] consists of a 5G Resource Grid Simulator (5G-RGS) as well as a Radio Resource Manager (RRM) scheduler named Slice-Aware Machine Learning-based Ultra-Reliable Scheduling (SAMUS). The 5G-RGS is based on the 5G specifications and uses a matrix-based resource grid, to provide and simulate resource scheduling. The modeled User Equipments (UEs) function as input for the grid, generating packets and combining them with parameters such as TTI, modulation order and RBs needed to transmit the packets. For the calculation of RBs needed to transmit the specific packet payload, the Transport Block Size (TBS) is used, stating the size of the transmission on the Medium Access Control (MAC) layer towards the physical layer. The latency is calculated for each packet as the time between instantiating the packet and processing it on the base station's side. One TTI is defined as 1 ms, which corresponds to a Subcarrier Spacing (SCS) of 15 kHz within the New Radio (NR) specification. The described 5G-RGS acts as the simulation environment for the SAMUS scheduler prototype.

The key aspect of this scheduling approach is the harnessing of parameters such as channel conditions (provided by the UEs via Channel Quality Indicators (CQIs)) and emerging data sent by each UE within the network. For the prioritization of network slices, the Greedy Network Slicing approach is used from [16]. The prediction of channel quality and the amount of data is a key feature in the framework. The prediction is done using the Auto-Regressive Integrated Moving Average (ARIMA) method, which showed promising results in providing accurate predictions [2] and was trained on data sets depicting data traffic from Smart Grid (Photovoltaic (PV)) systems as well as Electric Vehicle (EV) charging stations that we use in our evaluation.

Fig. 7 schematizes the procedure to validate the proposed method which will be detailed in Section VI-B2. The channel conditions in terms of mcs values, the amount and arrival times of the data packets of each 5G slice are inputs of both systems. The analytical model translates the mcs values into a degradation model which is then exploited by the proposed busy-window analysis for the computation of a set of WCRTs associated to multiple contexts Φ. The SAMUS framework

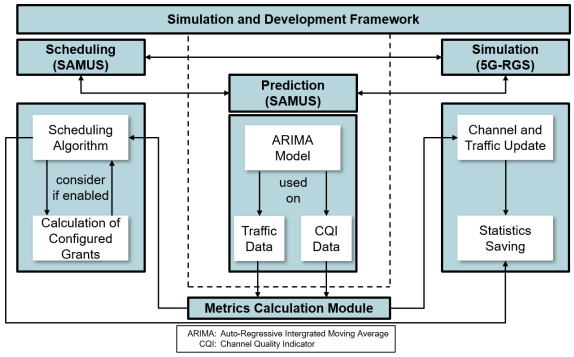

Fig. 6: SAMUS Framework: Interactions and overview of modules within the simulation and development framework.

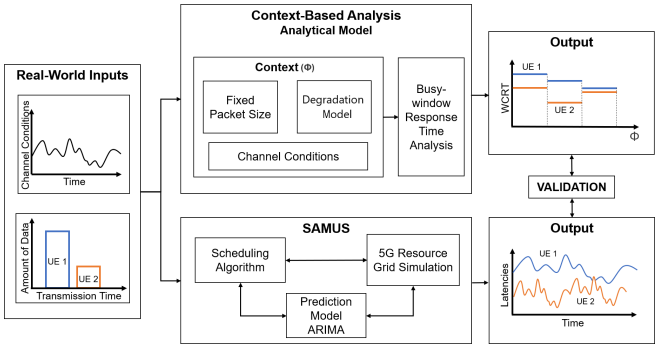

Fig. 7: Illustration of the adopted procedure during the validation process of our computer response times guarantees.

instead uses a matrix-based resource grid, to provide and simulate resource scheduling. The latency (or response time) is then calculated for each packet as the time between instantiating the packet and processing it on the base station's side. The bounds on the packets latencies obtained by the proposed method are then compared and validated with the ones of the 5G networks simulator.

2) Real-world Experiments: For the evaluation of our analytical model for context-based worst-case response time analysis with SAMUS, we consider applications and 5G network slicing configurations, in order to be able to compare results of the computed context-based response time bounds and simulation results of a real system. We consider therefore the following slices with different real-time requirements and criticality, using the 5G network through several users equipments where each UE is performing multiple data transmissions with different sizes and arrival times.

- Smart Grid (SG) slice: Data traffic in this slice is modeled after photovoltaic systems transmitting data proportionally to solar activity, data is obtained from National Renewable Energy Laboratory (NREL) [6]. This slice has the highest priority.
- Electric Vehicle (EV) Charging slice: EV charging point communication was modeled based on occupancy data

[6]https://www.nrel.gov/grid/solar-power-data.html

TABLE III: Overview of the wireless communication settings used in the experiments.

Channel Bandwidth	20 MHz
5G Subcarrier Spacing	15 kHz
5G MCS Index Table	256 QAM (Table 2)
SR Occasion	Every ms

TABLE IV: Overview of the experiment settings.

Packet Size EV	1950 Bytes
Packet Size SG	1500 Bytes
Number of UEs for Each Slice	1
Channel Quality	Changing every 20 ms
Adopted $M = \{(m, \mu), (k, \xi)\}$	From real-world measurements

gathered from chargecloud for the German city of Bonn [7]. This slice has the second higher priority.

As shown in Table III, we consider a 5G wireless network parameter configuration with a channel bandwidth of 20 Mhz that corresponds to 106 available RBs, subcarrier spacing of 15 kHz, packet TTI of 1 ms and mcs or quality of wireless channel that is monitored and can vary every 20 ms. As mentioned previously, since the RBs are limited, the execution of lower priority slices will be delayed whenever higher priority ones are activated, thereby leading to an increase of the response time of lower priority slices. We then compare the results obtained from the context-based analysis with data obtained from the SAMUS framework considering fixed packets sizes. In standard 5G configurations, the base station senses periodically ($d^{-}_{min} = 20ms$) the SNR of the received signal and adapts the value of mcs in case the channel condition improves, deteriorates or remain stable. Also in this case, the channel quality are predicted using the Auto-Regressive Integrated Moving Average (ARIMA) method [2].

As summarized in Table IV, each slice is composed by 1 UE which is sending every ms a packet of 1950 Bytes and 1500 Bytes for EV and SG slices, respectively.

a) Worst-Case Response Time Evaluation: Fig. 8 summarizes the results in terms of response times of the EV charging slice for both simulation and analysis. Note that, the latencies are given as integer values since each packet is scheduled within the subframe length of 1 ms in 5G. The blue bars identify the latency for each packet obtained through the SAMUS framework for a simulation of 1 minute. Most of the packets show a response time $\leq 1ms$ since they are completely transmitted in the same time slot in which they arrived. In some cases, some packets need an additional slot to be transmitted (values at 2 ms), while the worst case is shown at the end of the simulation with a peak of 5 ms. Every peak $> 1ms$ is caused by multiple factors such as larger packet sizes, lower mcs and higher interference from the highest priority slice (SG). Instead of providing a global highly pessimistic worst-case bound computed for worst possible channel conditions,

[7]https://new-poi.chargecloud.de/bonn (January 2020)

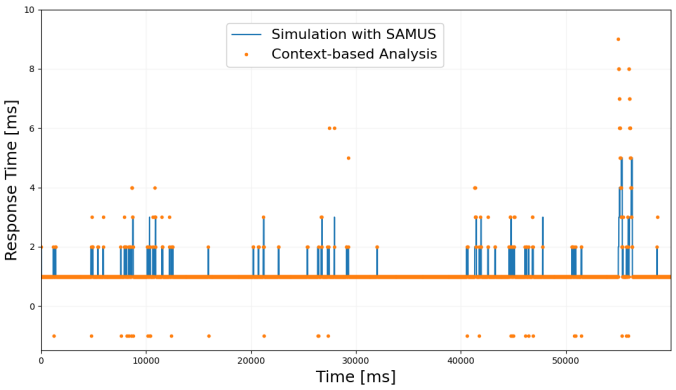

Fig. 8: Comparison between the response times obtained through SAMUS simulation (blue bars) and the context-based analysis (orange points).

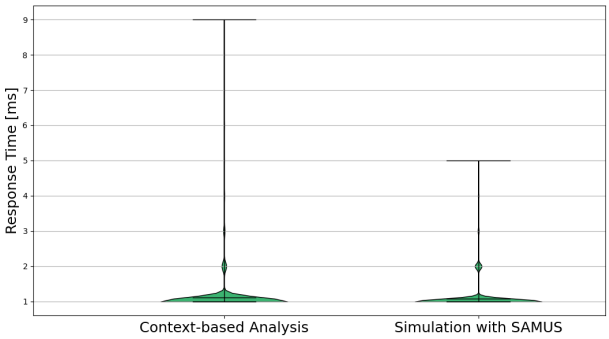

Fig. 9: Distribution of the response times obtained through SAMUS and the context-based analysis.

we evaluate the response time for each context Φ in order to provide a local bound of it (orange points in Fig. 8). The response time computed analytically successfully bounds the one obtained by the SAMUS simulator. However, the analysis introduces a pessimism which does not exceed 4 ms in this case.

The second limitation of the analytical model is represented by the values < 0. Each point at -1 represents a context for which the worst-case response time could not be computed (i.e., non schedulable contexts). As already discussed in Section VI-A2, this happens due to the observed limitations of schedulability condition and the pyCPA tool which is not able to provide guarantees in case the current bandwidth b_{mcs} is extremely low, thereby resulting in a highly loaded system. As previously, the results of the highest priority slice SG are omitted because the response time of every packet corresponds simply to its execution time due to the prioritization.

Fig. 9 shows the distribution of the response times of the EV charging slice for both analysis and simulation. Obtained results are comparable from both plots. The most commonly occurring response time is 1 ms (93% of the cases) while the remaining 7% of the packets has a higher value, distributed around 2 and a maximum of 5 ms in the simulation. In the analysis, the values of the analysis are spread among more values (between 2 until a maximum of 9 ms) due to the

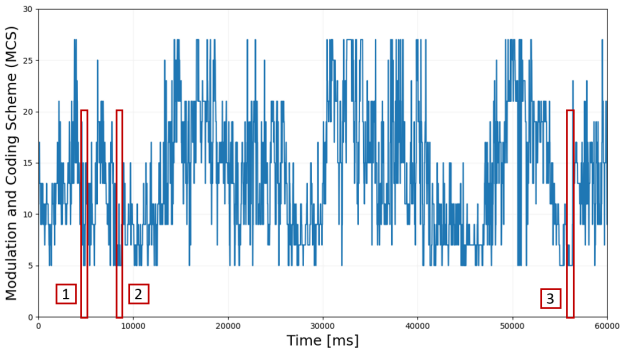

Fig. 10: Real-world variations of the channel quality for 1 minute of simulation.

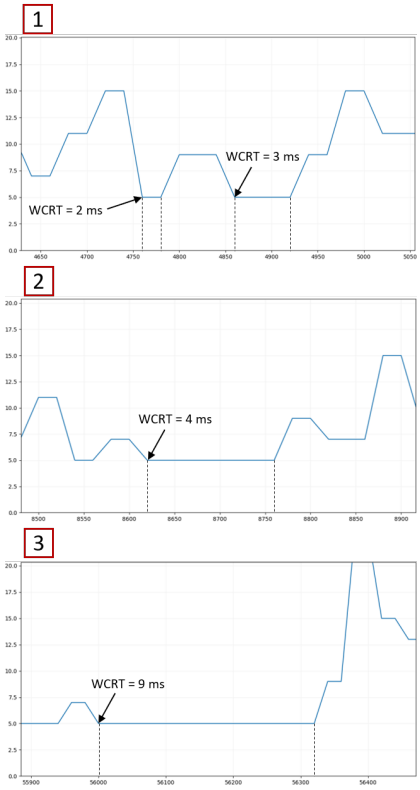

Fig. 11: Zoom of the first and second (1), third (2) and fourth (3) contexts.

pessimism of the analytical method. The distribution from the analysis is able nevertheless to drive good guarantees on the expected distribution obtained experimentally.

Note that the overhead of the analysis approach (i.e., obtain one point of WCRT for each context in Fig. 8) is directly proportional to the number of applications or slices. In practice, the number of applications is much smaller than the number of UEs (in our use case, 2 applications for 18 UEs).

b) Effect of Stationary Degradations: As a last step, we identify conditions that have a direct effect on the increase of the WCRT. From now on, we focus mostly on how the presence on stationary degradations μ have a direct impact on the response time. TableV summarizes for the selected

TABLE V: Overview of the properties of each context.

Time [ms]	mcs_{init}	μ	WCRT [ms]
4760 (1)	5	1	2
4860 (1)	5	3	3
8620 (2)	5	7	4
56000 (3)	5	16	9

contexts their time, mcs_{init}, number of stationary degradations μ and WCRT. The channel conditions, expressed as mcs values, of the lower priority slice and observed during the car-driven experiment are plotted in Fig. 10.

Differently from the mcs_{init} value which, in this particular experiment, is always at 5, the number of stationary degradations μ is the parameter which influences the WCRT. Note that μ represents the number of times the mcs remains constant before an improvement. A practical representation of the effect of μ on the packet latency can be clearly visualized in Fig. 11.

From the experiment in Section VI-A2, we observed that the longer is the time during which we are not observing any channel quality improvement, the higher is the WCRT. Similarly, in this case, we can notice that the WCRT is proportional to the amount of time in which the mcs remains stable after a degradation to the lowest value. The longer the time interval for which the channel quality remains low, the more the WCRT increases. Starting for the first case where the WCRT = 2 ms and the mcs remains 5 for only 20 ms, we can observe a WCRT = 9 ms when the mcs is stable to 5 for 320 ms (16 times d^{-}_{mcs}).

VII. Conclusion

We provide in this paper a formal response time analysis approach for 5G network slicing under dynamic channel conditions. The provided method considers initial channel conditions and models of degradation for bounding the response time for a given context. Experiments show that with appropriate characterization of models of degradation, we are able to successfully and tightly bound the timing of slices in a given context. Evaluation has also demonstrated that the number of degradation and their distribution, also considering duration of stationary degradation, have a large impact on response times. Future work include incorporating predictive solutions for estimating variations in channel conditions and incorporating prediction errors in the analysis, as well as considering more detailed physical models (e.g., multi-beamforming and steering, effects of signal reflection) of the environment as part of a context.

Acknowledgment

This work has been partly funded by the Federal Ministry of Education and Research (BMBF) via the project 6GEM (funding reference 16KISK038) and is supported by the Federal Ministry for Economic Affairs and Climate Action (BMWK) via the project 5Gain under funding reference 03EI6018C.

References

[1] Q. Chen, X. Wang, and Y. Lv, "An overview of 5g network slicing architecture," *AIP Conference Proceedings*, vol. 1967, no. 1, p. 020004, 2018. [Online]. Available: https://aip.scitation.org/doi/abs/10.1063/1.5038976

[2] C. Arendt, S. Böcker, and C. Wietfeld, "Data-driven model-predictive communication for resource-efficient iot networks," in *2020 IEEE 6th World Forum on Internet of Things (WF-IoT)*, 2020, pp. 1–6.

[3] A. Hamann, M. Jersak, K. Richter, and R. Ernst, "Design space exploration and system optimization with symta/s-symbolic timing analysis for systems," in *Proceedings of the 25th IEEE Real-Time Systems Symposium (RTSS 2004), 5-8 December 2004, Lisbon, Portugal*. IEEE Computer Society, 2004, pp. 469–478. [Online]. Available: https://doi.org/10.1109/REAL.2004.17

[4] L. Thiele, S. Chakraborty, and M. Naedele, "Real-time calculus for scheduling hard real-time systems," in *IEEE International Symposium on Circuits and Systems, ISCAS 2000, Emerging Technologies for the 21st Century, Geneva, Switzerland, 28-31 May 2000, Proceedings*. IEEE, 2000, pp. 101–104. [Online]. Available: https://doi.org/10.1109/ISCAS.2000.858698

[5] G. D. Abowd, A. K. Dey, P. J. Brown, N. Davies, M. Smith, and P. Steggles, "Towards a better understanding of context and context-awareness," in *Handheld and Ubiquitous Computing, First International Symposium, HUC'99, Karlsruhe, Germany, September 27-29, 1999, Proceedings*, ser. Lecture Notes in Computer Science, H. Gellersen, Ed., vol. 1707. Springer, 1999, pp. 304–307. [Online]. Available: https://doi.org/10.1007/3-540-48157-5_29

[6] M. E. Morocho-Cayamcela, H. Lee, and W. Lim, "Machine learning for 5g/b5g mobile and wireless communications: Potential, limitations, and future directions," *IEEE Access*, vol. 7, pp. 137 184–137 206, 2019.

[7] F. D. Calabrese, L. Wang, E. Ghadimi, G. Peters, L. Hanzo, and P. Soldati, "Learning radio resource management in rans: Framework, opportunities, and challenges," *IEEE Communications Magazine*, vol. 56, no. 9, pp. 138–145, 2018.

[8] T. Bag, S. Garg, Z. Shaik, and A. Mitschele-Thiel, "Multi-numerology based resource allocation for reducing average scheduling latencies for 5g nr wireless networks," in *2019 European Conference on Networks and Communications (EuCNC)*, 2019, pp. 597–602.

[9] F. Kurtz, C. Bektas, N. Dorsch, and C. Wietfeld, "Network slicing for critical communications in shared 5g infrastructures - an empirical evaluation," in *2018 4th IEEE Conference on Network Softwarization and Workshops (NetSoft)*, 2018, pp. 393–399.

[10] C. Bektas, D. Overbeck, and C. Wietfeld, "SAMUS: Slice-aware machine learning-based ultra-reliable scheduling," in *2021 IEEE International Conference on Communications (ICC)*, Montreal, Canada, jun 2021.

[11] B. Bangerter, S. Talwar, R. Arefi, and K. Stewart, "Networks and devices for the 5g era," *IEEE Communications Magazine*, vol. 52, no. 2, pp. 90–96, 2014.

[12] S. Nunna, A. Kousaridas, M. Ibrahim, M. Dillinger, C. Thuemmler, H. Feussner, and A. Schneider, "Enabling real-time context-aware collaboration through 5g and mobile edge computing," in *2015 12th International Conference on Information Technology - New Generations*, 2015, pp. 601–605.

[13] M. Alam, D. Yang, K. Huq, F. Saghezchi, S. Mumtaz, and J. Rodriguez, "Towards 5g: Context aware resource allocation for energy saving," *Journal of Signal Processing Systems*, vol. 83, p. 279–291, 2016.

[14] M. Richart, J. Baliosian, J. Serrat, and J.-L. Gorricho, "Resource slicing in virtual wireless networks: A survey," *IEEE Transactions on Network and Service Management*, vol. 13, no. 3, pp. 462–476, 2016.

[15] N. Dorsch, F. Kurtz, and C. Wietfeld, "Enabling hard service guarantees in software-defined smart grid infrastructures," *Computer Networks*, vol. 147, pp. 112–131, dec 2018.

[16] C. Bektas, S. Böcker, F. Kurtz, and C. Wietfeld, "Reliable software-defined RAN network slicing for mission-critical 5G communication networks," in *2019 IEEE Globecom Workshops (GC Wkshps)*, Waikoloa, Hawaii, USA, dec 2019. [Online]. Available: https://ieeexplore.ieee.org/document/9024677

[17] I. Vilà, O. Sallent, A. Umbert, and J. Pérez-Romero, "An analytical model for multi-tenant radio access networks supporting guaranteed bit rate services," *IEEE Access*, vol. 7, pp. 57 651–57 662, 2019.

[18] G. Zhang, J. Xu, L. Liu, Y. Yang, Q. Li, and M. Hamalainen, "Theoretical analysis of pf scheduling with bursty traffic model in ofdma systems," in *2017 IEEE International Conference on Communications (ICC)*, 2017, pp. 1–6.

[19] A. Nota, S. Saidi, D. Overbeck, F. Kurtz, and C. Wietfeld, "Providing response times guarantees for mixed-criticality network slicing in 5g," in *2022 Design, Automation Test in Europe Conference Exhibition (DATE)*, 2022, pp. 552–555.

[20] J. Diemer, J. Rox, and R. Ernst, "Compositional Performance Analysis in Python with pyCPA," in *WATERS 2012*, 2012. [Online]. Available: http://retis.sssup.it/waters2012/accepted/102_Final_paper.pdf

[21] G. T. 23.501, "System Architecture for the 5G System (Release 16)," Dec. 2019.

[22] J. Lehoczky, "Fixed priority scheduling of periodic task sets with arbitrary deadlines," in *[1990] Proceedings 11th Real-Time Systems Symposium*, 1990, pp. 201–209.

[23] S. Schliecker, J. Rox, M. Ivers, and R. Ernst, "Providing Accurate Event Models for the Analysis of Heterogeneous Multiprocessor Systems," in *Proc. 6th International Conference on Hardware Software Codesign and System Synthesis (CODES-ISSS)*, Atlanta, GA, Oct. 2008. [Online]. Available: http://doi.acm.org/10.1145/1450135.1450177