Dynamic resource scheduling optimization for ultra-reliable low latency communications: from simulation to experimentation

Ngoc-Lam Dinh, Rodolphe Bertolini, Mickael Maman

Dynamic Resource Scheduling Optimization for Ultra-Reliable Low Latency Communications: From Simulation to Experimentation

Lam Ngoc Dinh, Rodolphe Bertolini and Mickael Maman
CEA-Leti, Université Grenoble Alpes, F-38000 Grenoble, France
{ngoc-lam.dinh, rodolphe.bertolini, mickael.maman}@cea.fr

***Abstract*—In this paper, we propose a dynamic and efficient resource scheduling based on Lyapunov's optimization for Ultra-Reliable Low Latency Communications, taking into account the traffic arrival at the network layer, the queue behaviors at the data link layer and the risk that the applied decision might result in packet losses. The trade-off between the resource efficiency, latency and reliability is achieved by the timing and intensity of decisions and is adapted to dynamic scenarios (e.g., random bursty traffic, time-varying channel). Our queue-aware and channel-aware solution is evaluated in terms of latency, reliability outage and resource efficiency in a system-level simulator and validated by an experimental testbed using OpenAirInterface.**

I. Introduction

The 5G and beyond network enables the exploitation of new emerging use cases, such as Ultra-Reliable Low Latency Communication (URLLC). Advanced resource scheduling optimizations are required to jointly reduce latency and improve reliability while maintaining appropriate efficiency. While many mechanisms exist in the literature for the first two, it is still unclear how to efficiently utilize resources while maintaining reliable, low latency communications [1]. Given the requirements for delay and reliability, two approaches to activate available resources are proposed. On the one hand, reactive strategies activate additional resources on demand, which enable efficient resource utilization, but significantly increase latency, as the demand for additional communication resources is not instantaneous. On the other hand, proactive approaches systematically apply additional resources to stretch the latency below the deadline and usually consider the worst case with a margin. Therefore, this approach implies a high cost in terms of resource utilization, especially when worst-case impairments are very rare. Our goal is to design an early decision maker, as patented in [2], defining one or more decision moments to dynamically optimize the resource scheduling by adapting reactive-proactive modes to cope with various dynamic scenarios. The efficiency-latency-reliability trade-off is achieved by the timing and intensity of the decisions. The earlier (resp. stronger) the decision is made, the greater the latency gain (resp. reliability gain) at the cost of resource efficiency, and vice versa.

To highlight the benefits of early decision making in resource scheduling, Figure 1 illustrates the probability density function when the system reacts by setting a series of actions to achieve a latency gain. The clusters represent the latency when a transmission or several retransmissions (RTXs), are required for the receiver to decode the packet. At the end of each cluster, the system knows whether the packet was successfully delivered or not. Figure 1 shows how decisions made at different times (e.g., parallel RTXs at actions a_1 and a_2) can reduce latency at the cost of resource efficiency (i.e. after action a_1, packets that only needed one RTX were allocated two RTXs).

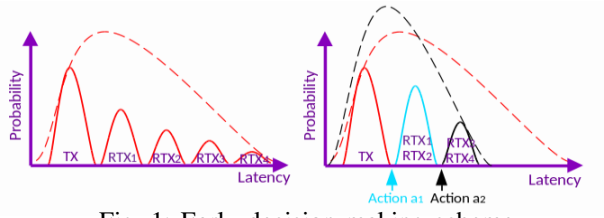


Fig. 1: Early decision making scheme

In order to highlight the importance of the above-mentioned trade-off for improving URLLC communications, we investigate the early decision maker in the well-known Hybrid Automatic Repeat reQuest (HARQ) retransmission protocol. In the literature, adaptation of the HARQ strategy is usually achieved by adapting the modulation and coding scheme [3], the transmission power [4] and the maximum number of retransmissions [5] but rarely at the scheduler level. A K-repetition scheme [6] and a proactive scheme with early termination [7] have been proposed, allowing for a number of redundant retransmissions upon receipt of the acknowledgment by the sender. By doing so, one can opportunistically decode the packet at the receiver in a shorter time at the expense of inefficient resource usage [8]. However, the adaptation of HARQ strategies at the scheduling level in a rapidly changing environment is limited in current research.

The randomness of bursty traffic and time-varying channel pose critical problems for URLLC, thus, dynamic scheduling (i.e. queue aware and channel aware) is required. In [9], they proposed a Closed-Loop ARQ protocol that dynamically reallocates the remaining resources between uplink and downlink slots upon the result of last uplink transmission. In [10], the transmission decision of the scheduling under delay and power constraints is based on data packet arrival, occupancy of the transmission queue and time-varying channel. In [11], a joint transmission - computation optimization achieves an optimal tradeoff between power and latency by taking into account the system dynamics. Hereafter, we propose a resource efficient, delay optimal, reliable scheduling adapted to dynamic scenarios (e.g., time-varying channel and traffic).

The contributions of this paper are as follows: *(1)* We formulate the dynamic resource scheduling problem by considering the traffic arrival in the network layer, the queue behaviors in the data link layer and the risk of applying vulnerable decision which causes packet loss. *(2)* The proposed solution includes resource efficiency considerations for URLLC applications whereas most solutions in the literature only consider latency reliability tradeoff. *(3)* We consider end-to-end performance by developing a system-level simulator based on NS-3 [12] applying to the 5G New Radio (NR). This simulator handles several HARQ processes and measures the latency between the transmitter Radio Link Control (RLC) layer and the receiver RLC layer assuming that transmission buffer size is infinity. We therefore consider both the queuing latency at the scheduler (due to reactive/proactive approaches) and (re)transmission latency (i.e. PHY/MAC). *(4)* We validate that our solution is implementable in OpenAirInterface framework compliant with 5G NR solution and we show the gain brought by our solution with real time hardware constraints.

The remainder of the paper is organized as follows: Section II presents the dynamic resource scheduling optimization including the system model, the problem formulation and its adaptation to HARQ procedure. Sections III and IV show the performance of the solution in simulation and experimentation respectively. Section V concludes the paper.

II. Dynamic Resource Scheduling Optimization

A. System Model

In this section, we describe the system model making the trade-off between the latency, the reliability and the resource efficiency, as illustrated in Figure 2. A series of actions $a_j \in \{a_0, ..., a_{max}\}$ is made at the corresponding action slot t. We can define two queues: The arrival rate queue $Q_1(t)$ is the RLC transmission buffer and contains the application packets. After completing the scheduler operations at MAC layer, the gNB prepares a Transport Block (TB) whose data is extracted from $Q_1(t)$ and sends it over the air. The scheduling rate queue $Q_2(t)$ keeps a copy of this TB and takes into account the ongoing scheduling processes that are not yet decoded at the UE side. Due to the dynamic nature of not only the traffic but also the channel behaviour, the lengths of $Q_1(t)$ and $Q_2(t)$ can be considered as random variables. The state of $Q_1(t)$ and $Q_2(t)$ demonstrated a two-stage queuing system whose length should be minimized.

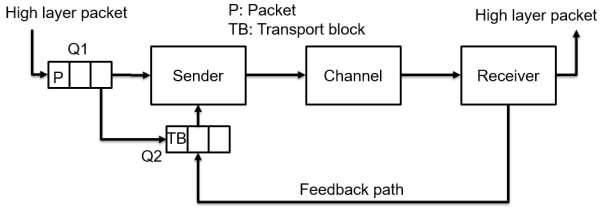


Fig. 2: System model

The queuing dynamic is defined as follows:

$$Q_1(t+1) = \max\{Q_1(t) - \alpha_a.TB_{a_0}, 0\} + A_1(t) \quad (1)$$

$$Q_2(t+1) = \max\{Q_2(t) - (1-\alpha_a).1_{TB}.TB_{a_j}, 0\} + A_2(t) \quad (2)$$

where $Q_i(t+1)$ are the backlogs of the queue i at the action slot $t+1$. $A_1(t)$ represents the total amount of high layer packets that arrive Q_1 at time t. During this action slot, an amount of TB_{a_j} will be served. The indicator function 1_{TB}, in Equation (2), is equal to 1 if the scheduling process of TB is successful and is 0, otherwise. If the first transmission of TB_{a_0} is a failure, $A_2(t) = TB_a$ will be added to Q_2, otherwise $A_2(t) = 0$ as the scheduling process of TB_{a_0} is ending. In order to control which queue will be served, we introduced the control variable α_a (1 and 0 mean serving $Q_1(t)$ and $Q_2(t)$ respectively). Knowing that ongoing processes have a higher priority, $\alpha_a = 0$ when $Q_2(t) > 0$.

B. Dynamic Resource Scheduling in HARQ Procedure

In this section, we apply the system model to HARQ procedure. Figure 3A first illustrates the classic HARQ procedure (i.e. send-wait-react mode). A delay L_{12} is introduced to demonstrate the TB preparation time from the gNB scheduler to the antenna. Then, a feedback will be encoded within an Uplink Control Information (UCI) message and sent back to the gNB after $T_{fb} = K_1$ slot(s), thus illustrating the processing time at the UE. In 5G NR standard, this processing time reflects a delay between the reception of the UL grant in the DL and the transmission of the corresponding UL data. Afterwards, the gNB has the information about the corrupted HARQ process on the UE side and decides to retransmit the erroneous TB after L_{12} slots. This process continues until the corrupted TB is successfully decoded by the UE or the maximum number of retransmissions R_{max} is reached. By doing this, the resources are perfectly utilized, but the latency could be unacceptable for URLLC communications.

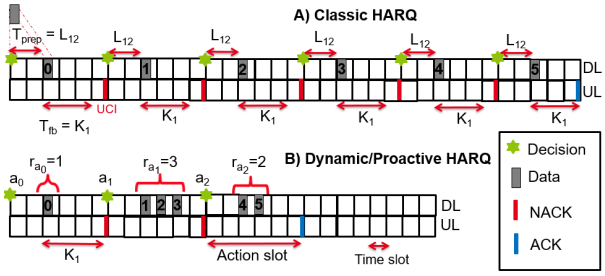


Fig. 3: Classical or Dynamic DL HARQ procedures

Instead of limiting the maximum number of allowed RTXs R_{max} for the scheduling process, our dynamic resource scheduling is restricted in terms of maximal possible actions a_{max}. Each action $a_j \in \{a_0, ..., a_{max}\}$ can allocate r_{a_j} proactive RTXs, between r_{min} and r_{max}, and a_0 corresponds to the first transmission. The decision maker we designed, dynamically chooses the number of actions a_j and their intensities (i.e, r_{a_j}) to reduce latency and improve resource efficiency and reliability. With respect to the resources allocated for proactive RTXs of a TB_n, the decision maker selects an element-wise positive resource allocation vector $(r_{n,a_0}, r_{n,a_1}, ..., r_{n,a_{max}})$. If r_{n,a_j} proactive RTXs are allocated by action a_j, the risk (i.e $\zeta(a_j)$) is expressed as follows:

$$\zeta(a_j) = \mathbb{P}[(\mathrm{SINR}_{tb_n}^{\sum_{k=0}^{j} r_{n,a_k}} \leq \mathrm{SINR}^t) \,|\, \mathrm{SINR}_{tb_n}^{\sum_{k=0}^{j-1} r_{n,a_k}}] \quad (3)$$

where $\mathrm{SINR}_{tb_n}^{\sum_{k=0}^{j-1}(r_{n,a_k})}$, $\mathrm{SINR}_{tb_n}^{\sum_{k=0}^{j}(r_{n,a_k})}$ are respectively the SINR of TB_n at previous (a_{j-1}) and current (a_j) action. SINR^t is the target SINR to decode TB_n.

Our proposed procedure dynamically adapts the resource scheduling to the traffic arrival in the network layer, the queue behaviors in the data link layer and the risk that the applied decision causes loss. It also automatically adapts the maximum number of RTXs to the channel conditions. Finally, to reduce the control overhead due to multiple feedbacks to the transmitter, we grouped their feedbacks into a single feedback that represents the current proactive retransmission status.

C. Problem Formulation

The objective is to optimally select r_{a_j} based on various factors, such as the current status of the $Q_1(t), Q_2(t)$, the current action index a_j and the risk that the applied decision causes loss. The main reliability constraint is to reduce the risk of the last action $\zeta(a_{max})$ below a predefined value ζ_o. However, the constraint associated with poor decision making must be defined for each upcoming action, not just for the last action. We define the risk for the current action $\zeta(a_j)$. In this case, the procedure has to retrigger other actions later, which consumes not only time and resources but also the reliability of the communication, when we are close to the maximum number of actions allowed. The index of the current action (i.e a_j) is thus very important. Clearly, the higher j is, the greater the sensitivity of TB loss will be if a wrong decision is applied, and the earlier the action (i.e low j) is, the higher the total number of RTXs can be. We define an objective function f_{obj} as the weighed sum of average number of resources allocated to each TB and the current risk, as follows:

$$f_{obj} = \lim_{N\to\infty} \frac{1}{N} \sum_{n=0}^{N-1} \sum_{a_0}^{a_{max}} r_{n,a_j} \times 1_{a_j} + \alpha \times f(a_j) \times \zeta(a_j) \quad (4)$$

where the indicator function 1_{a_j} is equal to 0 if the action a_j is successful (i.e. $\zeta(a_j) < \zeta_o$) and is 1, otherwise. $\alpha \geq 0$ is a constant value trading off risk and resource allocation. A higher value of α implies greater importance of risk minimization over the number of resources allocated (i.e. reliability over resource efficiency). The function $f(a_j)$ increases with the action index a_j. In our study, we consider $f(a_j) = j$.

Thus, our optimization problem $\mathcal{P}_1$ is to minimize the objective function f_{obj} subject to several constraints:

$$\min_{\{r_{n,a_j}\}_{n,a_j}} \quad f_{obj} \qquad (\mathcal{P}_1)$$

$$\text{s.t.} \quad \lim_{t\to\infty} \frac{\mathbb{E}\{Q_i(t)\}}{t} = 0, \quad \forall i \in \{1,2\}; \qquad (\mathcal{C}_{1,2})$$

$$r_{min} \times 1_{a_j} \leq r_{n,a_j} \leq r_{max} \times 1_{a_j}, \ \forall a_j \leq a_{max} \quad (\mathcal{C}_3)$$

$\mathcal{C}_{1,2}$ concerns the stability constraint of the queue $Q_{1,2}(t)$. $\mathcal{C}_3$ limits the number of decisions into a_{max} actions and constrains the maximal number of proactive RTXs at action a_j to r_{max}.

D. Proposed Algorithm

Our dynamic decision maker algorithm is based on Lyapunov's optimization tools, which do not require a-priori knowledge of stochastic processes in the ongoing system such as channel dynamics or traffic behaviours, to solve the optimization problem $\mathcal{P}_1$. Given a time-slotted system, we define the current state in the slot t as $\Theta(t) = (Q_1(t), Q_2(t)))$. Next, we define the one-slot conditional Lyapunov drift $\Delta(\Theta(t))$ representing the expected change of the Lyapunov function over a slot as follows:

$$\Delta(\Theta(t)) = \mathbb{E}\{L(\Theta(t+1)) - L(\Theta(t)) \mid \Theta(t)\} \quad (5)$$

Where $L(\Theta(t)) = \frac{1}{2}\left[Q_1^2(t) + Q_2^2(t)\right]$ is the Lyapunov function. By minimizing both $\Delta(\Theta(t))$ and f_{obj}, we can solve the problem $\mathcal{P}_1$ because the queues are stable in terms of average rate and the objective function is minimized . However, according to [13], a *performance-delay* trade-off between these dual objective optimizations can be parameterized by a constant V. By setting a large positive value to V, the control algorithm will favor minimizing the objective function f_{obj} over the stability of the average rate queues. Our objective is then to minimize the following *Lyapunov-drift-plus-penalty* function:

$$g(t) = \Delta(\Theta(t)) + V.\mathbb{E}\{f_{obj} \mid \Theta(t)\} \quad (6)$$

As defined in [13], the upper bound, $\gamma(t)$, can be derived for any action, any possible value of $\Theta(t)$ and any parameter $V > 0$ as follows:

$$\begin{aligned} \gamma(t) = B + V.\mathbb{E}\{f_{obj} \mid \Theta(t)\} \\ + \sum_{i=1}^{2} Q_i(t).\mathbb{E}\{A_i(t) - b_i(t) \mid \Theta(t)\} \end{aligned} \quad (7)$$

where B is a constant that satisfies:

$$\begin{aligned} B \geq \frac{1}{2} \sum_{i=1}^{2} \mathbb{E}\{A_i^2(t) - b_i^2(t) \mid \Theta(t)\} \\ - \sum_{i=1}^{2} \mathbb{E}\{A_i(t).\min\{Q_i(t), b_i(t)\} \mid \Theta(t)\} \end{aligned} \quad (8)$$

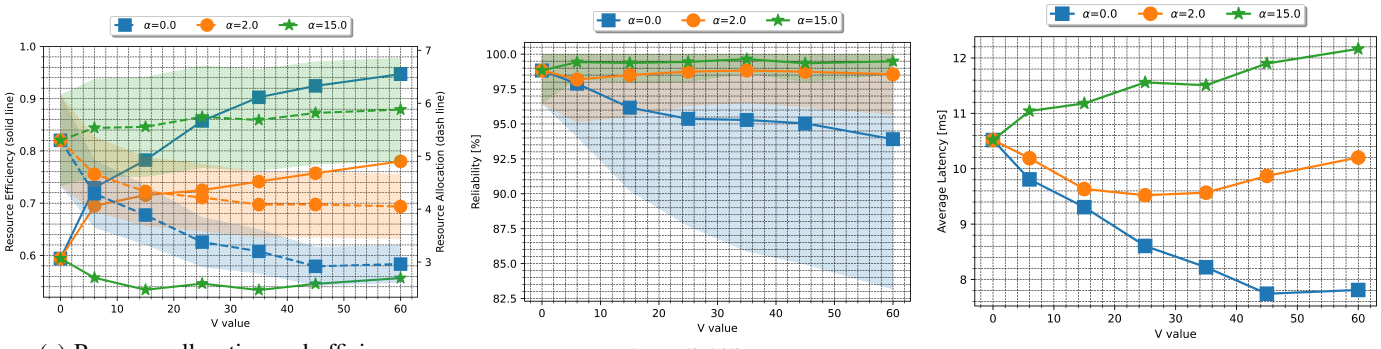
Through the opportunistic minimization framework of a conditional expectation [13], by minimizing $\gamma(t)$, the upper bound of the dual objective optimization, we can guarantee that the optimization problem $\mathcal{P}_1$ will be satisfied.

III. Simulation Results

A. Simulation Model and Assumptions

The network contains 1 gNB and 1 UE at fixed distance. In this work, packets are generated in exponentially distributed ON and OFF periods that follow the Internet Protocol (IP) traffic model [14]. The average duration of the ON and OFF periods are t_{ON} and t_{OFF}, respectively. In the ON state, packets of variable size are generated with an arrival rate λ_{ON} and fill $Q_1(t)$. In the scheduling process, K_1 and L_{12} are modelled to illustrate the feedback processing time and data preparation time at the UE and gNB, respectively. For simplicity, we assumed that the core network latency and propagation delay are negligible. Table I summarizes application, optimization and communication parameters.

With respect to the quality of data transmission over different Resource Blocks (RBs) is, an effective signal-to-noise ratio SINR_{eff}, which combines individual SINR received from individual RBs, is modeled using Effective Exponential SINR Mapping (EESM) method. In HARQ-IR, the SINR_{eff}^{r} after r



(a) Resource allocation and efficiency

(b) Reliability

(c) Average latency

Fig. 4: Trade off between resource efficiency, reliability and latency as a function of optimization parameters

TABLE I: Simulation and Experimentation Parameters

Parameters	Simulation	Experimentation
DL/UL Duplex	FDD	TDD
DL/Flex/UL	N/A	6/1/3 slots per Frame
(f_c, BW)	(3.61 GHz, 50 MHz)	
Numerology	1	
$(m, \eta_{S.eff}, CR)$	(5 , 0.7402 , 0.3701)	(3 , 0.4902 , 0.2451)
P_{tx}	8 dBm	-8 dBm
(U_{tx}, S_{rx})	$(4 \times 4, 2 \times 2)$	(1, 1)
a_{max}	5	
(r_{min}, r_{max})	(1, 5)	
(K_1, L_{12})	(2,2) time slots	(6,3) time slots
Packet Decoding	Risk $\zeta_0 = 10^{-4}$	CRC check
Traffic Type	ON-OFF	PING
Traffic Parameters	t_{on}/t_{off}= 1/3 Data Rate= 1.5 Mbps	Every 50 ms 64-Byte packet

RTXs is derived as [15], where SINR_{eff}^{r-1} is the effective SINR after the previous RTX, i.e., $r-1$ retransmission, $\text{SINR}_{x,r}$ is the SINR experienced by the x-th RB in the r-th RTX, and ω is the set of RBs. The value of β depends on the MCS selection.:

$$\text{SINR}_{eff}^{r} = -\beta \times \ln\left(\frac{1}{|\omega|} \times \sum_{x \in \omega} e^{-\frac{\text{SINR}_{x,r} + \text{SINR}_{eff}^{r-1}}{\beta}}\right) \quad (9)$$

B. Performance Evaluation

Performance is evaluated in terms of RAN latency, packet loss and resource efficiency. We define resource efficiency as the ratio of the number of radio resources required for a TB to be successfully received to the number of radio resources allocated by the scheduler. We also define RAN latency as the time between the arrival of IP packets in the RLC layer of the gNB and their arrival in the IP layer at the UE side.

Figure 4a shows the evolution of resource efficiency (solid line) as well as the average total number of radio resources allocated (dash line) as the function of V and α. We selected three values of α (i.e. 0, 2 and 15) that depend on the awareness of reliable transmission's objective. When reliability is not considered ($\alpha = 0$), our algorithm tends to spend less radio resources for each action and thus, resources are used efficiently. When V increases, we put more emphasis on minimizing resource allocation, so resource efficiency is further improved. When α appears and grows, the goal of reducing packet loss is also taken into account. The decision maker adapts to the channel conditions and allows more generous allocations for each action and this leads to high resource allocation with high standard deviation and low resource efficiency.

Reliability of communication is guaranteed at the cost of low resource utilization as shown in Figure 4b. When α is high, the transmission error is significantly low and the communication reliability no longer depends on the V-value (i.e., 99.5% and 98.5% of the total packets successfully reach the IP layer at the UE side for $\alpha = 15$ and $\alpha = 2$, respectively). However, the dependent relationship between the transmission reliability and the V-value is observed for $\alpha = 0$. In this case, we barely follow the minimization of the number of resources allocated for each action rather than the reliability of its transmission, thus, we noticed more error-prone transmissions when V increases. Figure 4c shows that the average latency at the RAN mainly depends on α. Redundant radio resources are scheduled when α is high to improve reliability, but this can result in increased queuing delay as incoming packets must wait longer in the queue before being served.

Figure 5 compares the Complementary Distribution Function (CDF) of latency for different HARQ schemes: *(i)* **Classic HARQ** procedure, *(ii)* Fixed number of **parallel RTXs**, *(iii)* **Proactive HARQ** adaptation with a fixed maximum number of RTXs ($R_{max} = 10$) as defined in [16] *(iv)* our proposed optimization (**Dynamic HARQ**) in which $R_{max} = \sum_{a_0}^{a_{max}} r_{n;a_j}$. According to Figure 4, we select two pairs of (V, α) parameters: $(25, 2)$ for good reliability and considerably low latency and $(60, 0)$ for very good resource efficiency and latency. As expected, the latency of Classic HARQ is the highest and spreads out over time. 2-parallel and 5-parallel HARQ improve latency at the cost of decreasing resource efficiency to 0.8 and 0.6, respectively due to the lack of adaptation when needed. Dynamic HARQ offers two tradeoffs. When $V = 60$ and $\alpha = 0$, we improve resource efficiency and latency but not reliability. When $V = 25$ and $\alpha = 2$, we improve reliability at the cost of a slight degradation in latency in the best case.

IV. Validation Through Experimentation Using OpenAirInterface

A. Experimental Testbed

In this section, we propose an experimental testbed of the resource scheduling optimization using an open and reconfigurable Software-Defined Radio (SDR) environment. Our

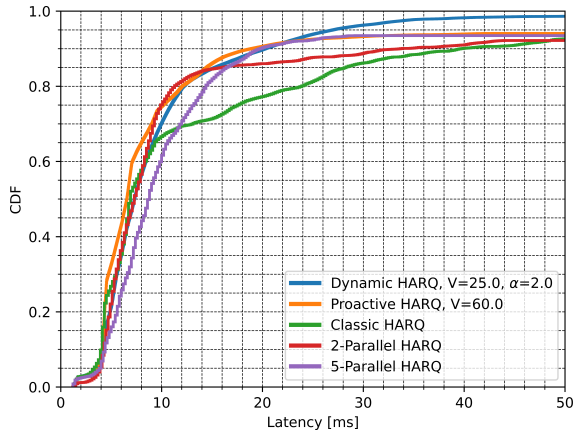


Fig. 5: Simulated latency CDF for different HARQ schemes

implementation is based on OpenAirInterface (OAI) [17], an open-source framework that aims to provide a pluggable cellular network solution, to avoid the limitations of vendor implementations and to allow for protocol customization. OAI provides an end-to-end 5G NR cellular network implementation including the Radio Area Network (RAN) and the Core Network (CN). As shown in Figure 6, our testbed consists of two high-end computers, one that is used for running an instance of an OAI UE (green rectangle) and an instance of an OAI 5G CN (yellow rectangle), one that is used for running an instance of an OAI gNB (red rectangle); and two USRP b210, radio head of the UE and the gNB, connected to the two computers. Since 3.61 GHz belongs to the 5G licensed C-Band, we use SMA cables instead of antennas to interconnect USRPs. In our experimental scenario, we use variable attenuators to experiment several channel conditions between the UE and the gNB. Packets generated are PING of 64 $Bytes$ every 50 ms.

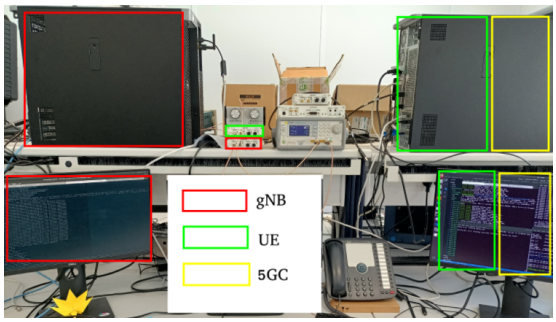


Fig. 6: OAI experimentation Testbed

We have implemented the different resource scheduling optimization schemes in the develop version of OAI 5G NR RAN. This includes (but is not limited to) modifying the redundant version of a TB and the HARQ process to handle parallel retransmissions of the same TB by gNB and UE MAC entity and collecting metrics needed to our algorithm, such as DL channel status, buffer length (i.e. Q_1 and Q_2) in the RLC layer and risk probability based on SINR measurements.

B. Deviations From Simulations

The current status of 5G implementation in OAI coupled with the limited capabilities of the USRP do not offer the same freedom as the NS-3 simulator. In this paper, our aim is not to directly compare the performances between simulation and experimentation, but to verify if our solution is feasible in a real environment and to show the gain brought by our solution with real time hardware constraints.

Our implementation starts from the developed version of OAI 5G NR RAN. The main difference with simulation concerns the spectrum usage technique. While simulations use an Frequency Division Duplex (FDD), OAI uses a Time Division Duplex (TDD). In a 10-slot frame, the first 6 slots are dedicated to DL and the last 3 slots to UL. The seventh slot is a flexible slot (Flex) composed of 6 DL symbols and 4 UL symbols. According to this implementation, the Radio Resource Control (RRC) layer of the UE sets K_1 to a minimum of 6 slots to allow OAI sufficient processing time, and the gNB scheduler sets a delay L_{12} of at least 3 slots. Moreover, in our experiment, each TB contains a PING packet instead of aggregated application packets. Table I summarizes OAI experimentation parameters.

For the implementation of our different resource scheduling optimization schemes, Equation (3) needs the DL channel status. The Channel Quality Indicator (CQI) is usually calculated with the SINR of the transmission occurring in the Downlink Shared Channel (DLSCH) to be acknowledged in the current UCI. As the CQI in the UCI is not implemented in the current version of OAI, we extrapolate it using the UL channel estimation performed by the gNB in our TDD configuration and we estimate the risk (i.e $\zeta(a_j)$) based on online statistics.

C. Performance Evaluation

Figure 7 depicts the resource allocation and resource efficiency as a function of optimization parameters (i.e. V and α). Due to the implementation deviations detailed in the section IV.B, the size of the queues, and thus the weights V and α, are different from those of the simulations.

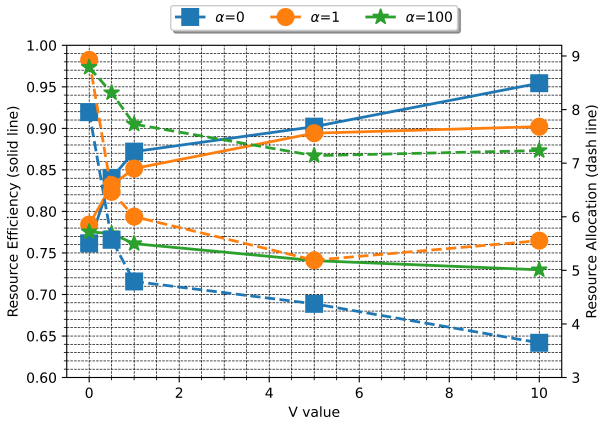


Fig. 7: Resource efficiency and resource allocation per TB as a function of optimization parameters

For $\alpha = 0$, the decision maker does not consider reliability and mainly optimizes resource efficiency. When V increases, the decision maker allocates fewer resources, which leads to greater resource efficiency. For this experiment, the right average level of the number of resources allocated is between 3 and 4. When α increases, the decision maker trades off

reliability (which needs more resources) and efficiency (which limits the number of resources allocated). So the larger alpha is, the less efficient the scheduling is.

Figure 8 shows the latency CDF obtained by experimentation with OpenAirInterface for different HARQ procedures (i.e. Classic HARQ, 2,5-Parallel HARQs, Proactive HARQ with $V = 0.06$, Dynamic HARQ with $V = 5$ and $\alpha = 100$). In this experiment, the latency is measured at the MAC layer, instead of the upper application layer. The approximate 3 ms staircase shape of the curve is explained by the TDD DL/UL duplexing of our experiment. Indeed, since (K_1, L_{12}) is (6,3) time slots, there are, for example, 3 ms (i.e. 6 consecutive DL slots) before the UL transmission.

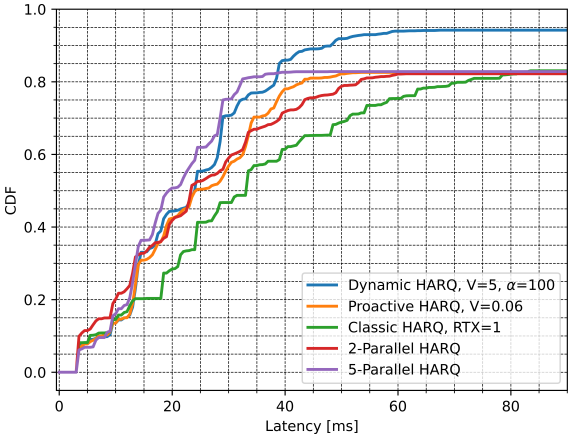


Fig. 8: Experimental latency CDF for different HARQ schemes

We can see that the stronger the action, the lower the latency. The gain in latency is 30% and 55% between Classic HARQ and 2,5-Parallel schemes, respectively. The average latency of successfully completed HARQs is 29.5 ms, 21.7 ms, and 17.8 ms for Classic HARQ and 2,5-Parallel HARQs, respectively. Proactive HARQ automatically adjusts the intensity of each action and trades off resource efficiency and latency, but it is limited by the maximum number of RTXs ($R_{max} = 10$). Due to this limitation, it achieves the same upper bound of 88% HARQ completion as the other schemes. The average latency of Proactive HARQ with $V = 0.06$ is 21.7 ms, similar to that of 2-Parallel HARQ. A close upper bound is achieved by our Dynamic HARQ with $V = 5$ and $\alpha = 1$. By setting α greater than 1 (i.e. 100), our dynamic decision maker outperforms other reliabilities, since it reaches 95% of completion, while ensuring an average latency of 18.2 ms.

V. Conclusions

In this paper, we propose a reliable, resource and delay-optimized scheduling suitable for dynamic scenarios (e.g., random bursty traffic, time-varying channel) based on Lyapunov optimization for Ultra-Reliable Low-Latency Communications. It takes into account the traffic arrival at the network layer, the queue behaviors at the data link layer and the risk that the applied decision might trigger packet loss. The trade-off between the resource efficiency, latency and reliability is achieved by the timing and intensity of decisions and can be parameterized with V and α. Our queue-aware and channel-aware solution is evaluated in a system-level simulator and validated by an experimental testbed using OpenAirInterface. In future work, we will extend our solution to multiple users and accesses. Specifically, we will focus on the dual mode of grant-based access (i.e. scheduled) and grant-free access (i.e. opportunistic with collisions) in 5G-NR and study how competing users can opportunistically share resource pool.

Acknowledgment

This work was partially supported by the ECSEL Joint Undertaking (JU) programme, under grant number N°826276 (CPS4EU project) and the European Union H2020 / Taiwan Project 5G CONNI [18] under grant N°861459.

References

[1] C. She, C. Yang, and T. Q. S. Quek, "Radio Resource Management for Ultra-Reliable and Low-Latency Communications," *IEEE Communications Magazine*, vol. 55, no. 6, pp. 72–78, 2017.

[2] M. Maman, L. N. Dinh, and E. Calvanese Strinati, "Procede et dispositif d'orchestration de l'execution de mecanismes dans un reseau sans fil," FR Patent 2103542, 2021.

[3] S. Pfletschinger, D. Declercq, and M. Navarro, "Adaptive HARQ With Non-Binary Repetition Coding," *IEEE Transactions on Wireless Communications*, vol. 13, no. 8, pp. 4193–4204, 2014.

[4] M. Jabi, M. Benjillali, L. Szczecinski, and F. Labeau, "Energy Efficiency of Adaptive HARQ," *IEEE Transactions on Communications*, vol. 64, no. 2, pp. 818–831, 2016.

[5] H. Mukhtar, A. Al-Dweik, and M. Al-Mualla, "Content-aware and occupancy-based hybrid ARQ for video transmission," in *2016 IEEE 59th International Midwest Symposium on Circuits and Systems (MWSCAS)*, pp. 1–4, 2016.

[6] T.-K. Le, U. Salim, and F. Kaltenberger, "An Overview of Physical Layer Design for Ultra-Reliable Low-Latency Communications in 3GPP Releases 15, 16, and 17," *IEEE Access*, vol. 9, pp. 433–444, 2021.

[7] Y. Liu, Y. Deng, M. Elkashlan, A. Nallanathan, and G. K. Karagiannidis, "Analyzing Grant-Free Access for URLLC Service," *IEEE Journal on Selected Areas in Communications*, vol. 39, no. 3, pp. 741–755, 2021.

[8] T. Jacobsen, R. Abreu, G. Berardinelli, K. Pedersen, P. Mogensen, I. Z. Kovacs, and T. K. Madsen, "System Level Analysis of Uplink Grant-Free Transmission for URLLC," in *2017 IEEE Globecom Workshops (GC Wkshps)*, pp. 1–6, 2017.

[9] B. Han, Y. Zhu, M. Sun, V. Sciancalepore, Y. Hu, and H. D. Schotten, "CLARQ: A dynamic ARQ solution for ultra-high closed-loop reliability," *IEEE Transactions on Wireless Communications*, vol. 21, 2022.

[10] M. Wang, J. Liu, W. Chen, and A. Ephremides, "Joint queue-aware and channel-aware delay optimal scheduling of arbitrarily bursty traffic over multi-state time-varying channels," *IEEE Transactions on Communications*, vol. 67, no. 1, pp. 503–517, 2019.

[11] D. Han, W. Chen, and Y. Fang, "Joint channel and queue aware scheduling for latency sensitive mobile edge computing with power constraints," *IEEE Transactions on Wireless Communications*, vol. 19, no. 6, pp. 3938–3951, 2020.

[12] N. Patriciello, S. Lagen, B. Bojovic, and L. Giupponi, "An E2E simulator for 5G NR networks," *Simulation Modelling Practice and Theory*, vol. 96, p. 101933, Nov. 2019.

[13] M. J. Neely, *Stochastic Network Optimization with Application to Communication and Queueing Systems*, vol. 3. 2010.

[14] M. Marvi, A. Aijaz, and M. Khurram, "On the Use of ON/OFF Traffic Models for Spatio-Temporal Analysis of Wireless Networks," *IEEE Communications Letters*, vol. 23, no. 7, pp. 1219–1222, 2019.

[15] S. Lagen, K. Wanuga, H. Elkotby, S. Goyal, N. Patriciello, and L. Giupponi, "New Radio Physical Layer Abstraction for System-Level Simulations of 5G Networks," in *ICC 2020 - 2020 IEEE International Conference on Communications (ICC)*, pp. 1–7, June 2020.

[16] L. N. Dinh, I. Labriji, M. Maman, and E. Calvanese Strinati, "Toward URLLC with Proactive HARQ Adaptation," *To appear in EUCNC2022*.

[17] N. Nikaein, M. K. Marina, S. Manickam, A. Dawson, R. Knopp, and C. Bonnet, "Openairinterface: A flexible platform for 5g research," *ACM SIGCOMM Computer Communication Review*, pp. 33–38, 2014.

[18] E. Calvanese Strinati, T. Haustein, M. Maman, W. Keusgen, S. Wittig, M. Schmieder, S. Barbarossa, M. Merluzzi, H. Klessig, F. Giust, D. Ronzani, S. P. Liang, J. S. J. Luo, C. Y. Chien, J. C. Huang, J. S. Huang, and T. Y. Wang, "Beyond 5G Private Networks: the 5G CONNI Perspective," in *2020 IEEE Globecom Workshops (GC Wkshps)*, 2020.