Hindawi
Wireless Communications and Mobile Computing
Volume 2019, Article ID 2729370, 19 pages
https://doi.org/10.1155/2019/2729370

Research Article

Access Time Analysis of MCPTT Off-Network Mode over LTE

Yishen Sun, Wesley Garey, Richard Rouil, and Priam Varin

Wireless Networks Division, National Institute of Standards and Technology, Gaithersburg, MD 20899, USA

Correspondence should be addressed to Yishen Sun; yishen.sun@nist.gov

Received 30 November 2018; Accepted 27 January 2019; Published 2 April 2019

Guest Editor: Maurizio Casoni

Public safety organizations around the world started migrating toward Long-Term Evolution (LTE) networks to support the increasing needs for video and data. To address the unique voice communication requirements of first responders, the 3rd Generation Partnership Project (3GPP) introduced new capabilities that aim at providing similar functionalities as the traditional Land Mobile Radio (LMR) systems, namely, Direct Mode communication and mission critical push-to-talk (MCPTT). Direct Mode communication, also called Proximity Services (ProSe), allows public safety users to communicate directly with each other regardless of the network status. MCPTT was the first mission critical service, and first application, standardized by 3GPP to provide both on- and off-network voice capability. Assessing the performance of those capabilities is critical to accelerate their deployment and adoption by first responders. In this study, we evaluate the performance of an off-network mode MCPTT device over ProSe by focusing on the access time, a measure of the delay incurred before a user can talk. We develop analytical models for various types of calls and verify the accuracy of the predicted access time using ns-3 simulations. We perform sensitivity analysis to show the validity of the models for various scenarios. Finally, we show how the models can be used to guide parameter configuration for both MCPTT and ProSe to optimize the performance.

1. Introduction

Public safety agencies require real-time applications, and voice communication remains the most important means of communication for first responders in emergency situations. While most existing push-to-talk (PTT) communication is carried over Land Mobile Radios (LMRs) to achieve public safety grade voice quality and reliability, first responders have already adopted broadband technology, including 3rd Generation Partnership Project (3GPP) Long-Term Evolution (LTE), to access video and data. Thanks to the effort of public safety organizations around the world, 3GPP has extended LTE's capabilities to support many of the situations faced by first responders and increase the network's reliability. In particular, extensions were made to fulfill the mission critical voice communication requirements defined by the public safety community [1, 2]. Those requirements include PTT, which is the standard means of communicating amongst first responders (similar to walkie-talkie), and Direct Mode, which allows first responders to communicate unit-to-unit without using the network. Also included is the ability to talk to multiple units at once (i.e., group call), the identification of the person speaking, and notification of emergency situations.

PTT is not new, and a lot of research has been done on the Open Mobile Alliance (OMA) PTT over Cellular (PoC) services [3] in recent years. Early evaluations of PoC were done over 3G technologies and the results showed significant setup delays [4] and talking wait time on the order of seconds [5, 6]. Even over LTE, PoC is far from being able to provide the performance needed by first responders. For example, Kuwadekar and Al-Begain evaluated PoC using a real network under different conditions [7]. The call setup delay was measured in the testing, which is the time from when the PTT button was pressed to the time when all the UEs were connected so that the user could speak. The results showed that if the network was not loaded, the call setup delay ranged from 0.7 s to 4.6 s, and that the average call setup delay was 2.556 s, which is higher than what is seen in LMR systems. Kuwadekar and Al-Begain also showed that PTT call setup delay increased roughly linearly with respect to the amount of emulated background calls that they injected into the network, and the call setup delay exceeded 10 s once the number of simultaneous background calls exceeded 700. Tests of PTT during a festival, when the local network was being used by up to around 10,000 people, resulted in a 100% call failure rate. Those high delays are attributed to the Session Initiation Protocol (SIP) used by PoC and provided by the Internet Protocol (IP) Multimedia Subsystem (IMS). An enhanced version of PoC for public safety, called Push-to-Communicate for Public Safety (PCPS) [8], was developed and aligned with 3GPP efforts to standardize a PTT application capable of meeting public safety needs in terms of features and performance. This Mission Critical Push-to-Talk (MCPTT) application was introduced in LTE Release 13 and further enhanced in later releases [9]. The current efforts to evaluate MCPTT mainly focus on on-network scenarios, where the devices are connected to the network via a base station. In [10], Kim et al. proposed an algorithm to shorten the time of the initial uplink transmission in order to reduce the control plane latency. In [11], Solozobal et al. looked at using mobile edge computing to provide a flexible architecture in 5G networks where the user plane can be deployed closer to the users, thus improving performance. The proposed solution would also allow the network operator to deploy the control plane at the edge of the network to provide a standalone service if the evolved NodeB (eNodeB) is disconnected from the core network.

To support Direct Mode capability in LTE, 3GPP introduced Proximity Services (ProSe) capability in Release 12 and has continued to enhance ProSe in subsequent releases [12]. ProSe allows a piece of User Equipment (UE), such as a laptop or a phone, to discover and communicate with other UEs. ProSe supports both one-to-one and one-to-many transmissions, thus allowing for efficient group communication. The research on ProSe communication has primarily focused on the coverage and capacity aspect of this new feature. Wang and Rouil evaluated the expected device-to-device (D2D) coverage using ProSe [13] for various channel conditions and conducted a sensitivity analysis to determine the main factors impacting coverage. This work considered only a single link and did not capture the effects of multiple users accessing the same resources. Griffith et al. developed analytical and simulation models to evaluate the probability that a control packet reaches a certain number of devices assuming a particular control channel resource pool configuration [14]. In this work, multiple devices were considered but conservative assumptions were made regarding the channel effects. A follow-up work by Griffith et al. was done to evaluate data transmissions over the shared channel [15] and a similar work was completed by Cipriano and Panaitopol [16]. The results of those contributions can help operators allocate resources based on a desired performance target but do not consider end-to-end performance metrics such as latency.

To the best of our knowledge, this is the first work that evaluates the performance of off-network MCPTT over ProSe. Using the MCPTT access time as the key performance indicator (KPI), we propose analytical models to estimate the access time under different scenarios. The models can be used to guide parameter configuration, and to better understand the interaction between MCPTT and ProSe. We have also developed a ns-3 model for simulating MCPTT over ProSe [17], which we used to validate the predicted delays. Major functions of the simulation model include floor control, call control, call type control, and emergency alert. All protocol operations are based on 3GPP MCPTT specifications [18, 19]. We designed test cases and used them to check the correctness of the protocol models that we implemented in the simulator [20].

The rest of the paper is organized as follows. In Section 2, we describe D2D communication over ProSe and our baseline model to estimate the transmission delay. In Section 3, we leverage the ProSe model to develop a set of theoretical models to estimate the MCPTT access time for the various types of calls. We included simulation results to compare with our analytical estimates. In Section 4, we conduct a sensitivity analysis of the ProSe and MCPTT parameters. Section 5 concludes the paper and outlines future work.

2. Modeling of Message Transmission Time over ProSe

2.1. D2D Communication over ProSe Sidelink. The need for wireless communication among public safety users while out-of-coverage can be met by utilizing Mode 2 of D2D communications supported by 3GPP LTE [12]. In Mode 2, UEs operate without any network supervision, which is the case when the UEs are out of coverage of any base stations, and voice and data communications are carried over the air interface referred to as the ProSe sidelink, or "sidelink" for short. D2D communication over sidelink is performed over periodically repeating sets of sidelink resources in the time domain, where a single set of sidelink resources compose a sidelink period. Each sidelink period contains instances of two channels that are separated in the time domain: the Physical Sidelink Control Channel (PSCCH) and the Physical Sidelink Shared Channel (PSSCH), as depicted in Figure 1. Each channel is defined by a resource pool, whose elements are defined by a combination of certain Resource Block (RB) indexes in the frequency domain, and certain subframe indexes in the time domain. The transmitting UE uses the PSCCH to send a Sidelink Control Information (SCI) message to indicate to which UEs its subsequent data messages are addressed, where and when the data transmissions on the PSSCH will occur (i.e., resource assignment in the time and frequency domains), and how (i.e., the modulation and coding scheme). Upon successful reception of the SCI message, any receiving UEs can tune to the corresponding resources in the PSSCH. Transmissions in the PSSCH use a Time Resource Pattern (TRP), which is a subframe indication bitmap of a fixed length N_{TRP} (e.g., 8 subframes for LTE in Frequency Division Duplex (FDD) mode) repeated in the time domain through the length of PSSCH, to identify which subframes are used by a transmitting UE. Each TRP is identified by an index I_{TRP} corresponding to the predefined subframe indication bitmap established in [21]. Every data transmission on the PSSCH is performed with four HARQ transmissions. In addition, if outgoing data is to be scheduled for transmission in a given occurrence of the PSSCH, the data has to be available at least 4 ms before the beginning of the

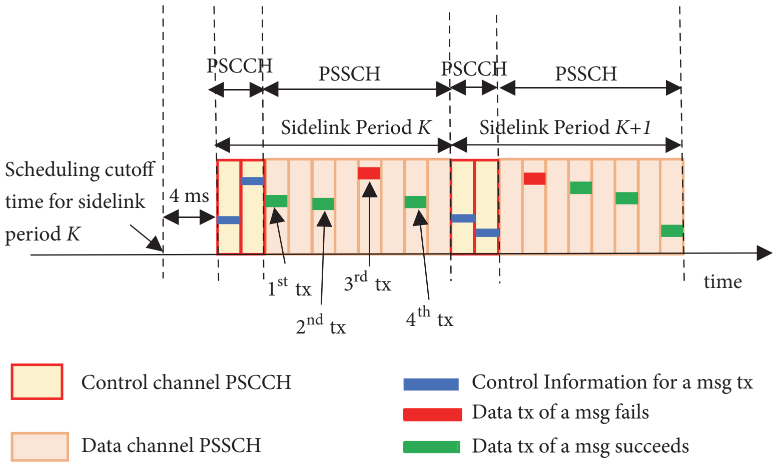


Figure 1: D2D communication over ProSe Sidelink, with periodically repeating control and shared channels.

corresponding sidelink period. In this paper, we identify this gap as the scheduling cutoff time. Sidelink communication is considered half duplex, which means that a device is not able to receive messages if it is transmitting in the same subframe.

2.2. One Way Transmission Time. Let X be a message that a transmitting UE must send and let the one-way transmission time of message X over the sidelink, denoted as $T_{tx}(X)$, be the duration from when the message X becomes available at the transmitting UE to the time when message X is received successfully by the receiving UE. The calculation of $T_{tx}(X)$ considers the packet error rate (PER) of each transmission attempt, the random occurrence of PTT button push timing, and ProSe configuration parameters. To be more specific, $T_{tx}(X)$ can be written as a function $f()$ that has the form:

$$\begin{aligned} T_{tx}(X) = f\Big(& T_{SL}, T_{PSCCH}, T_{delay}, [p_1, \dots, p_N], \\ & Prob.(PTT\ \ pushing), N_{TRP}, I_{TRP}, N, l) \end{aligned} \tag{1}$$

where

(i) T_{SL} is the duration of a sidelink period;

(ii) T_{PSCCH} is the duration of the control channel (PSCCH) of a sidelink period;

(iii) T_{delay} is the transmission plus propagation delays from the sending UE to the targeted receiving UE;

(iv) $p_n, n = 1, 2, \dots, N$, is the PER of the n-th transmission attempt within a sidelink period, with N being the total number of transmission attempts of a packet in one sidelink period;

(v) $Prob.\ (PTT\ \ pushing)$ represents the distribution of the PTT button push timing;

(vi) N_{TRP} and I_{TRP} are ProSe time resource pattern related parameters;

(vii) l is an integer between 1 and L, with $L = \lfloor (T_{SL} - T_{PSCCH})/(N \times N_{TRP}) \rfloor$, and the MCPTT message is included in the l-th Transmission Block (TB) of the sidelink period data channel (PSSCH).

Since $T_{tx}(X)$ remains the same for all MCPTT messages discussed in this paper and thus does not depend on X, we refer to it as T_{tx} for short.

The minimum value of T_{tx} that one message may experience occurs when the following three conditions are met, as shown in Figure 2(a):

(1) the message becomes available for transmission right before the scheduling cutoff time of the upcoming sidelink period K (i.e., 4 ms ahead of the start of sidelink period K);

(2) the message is scheduled to be transmitted in the first subframe of the data channel of the sidelink period (e.g., $I_{TRP} = 0$); and

(3) the first transmission attempt of the message is successfully received by the targeted UE.

Thus, the minimum value for T_{tx} in units of milliseconds is

$$\min(T_{tx}) = 4 + T_{delay} + T_{PSCCH}; \tag{2}$$

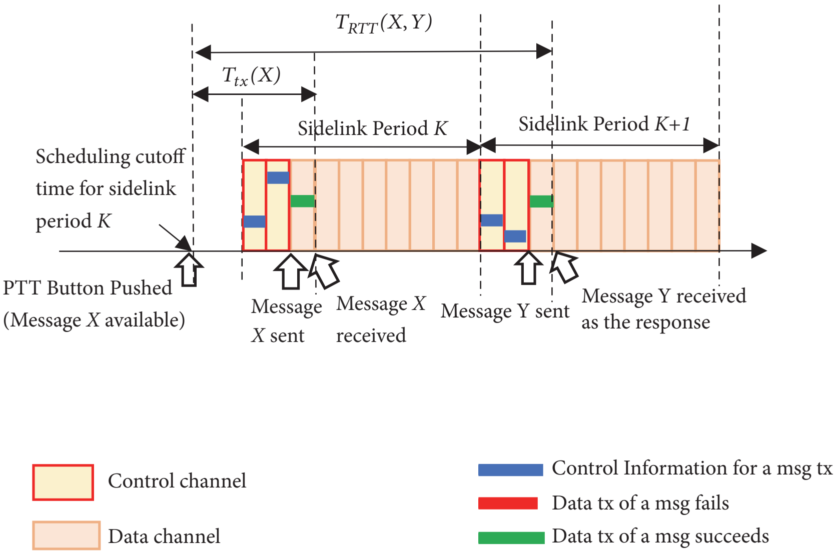
The maximum value of T_{tx} that one message may experience, assuming that only the last transmission attempt is successfully received by the targeted UE, is shown in Figure 2(b). The message transmission time can be even larger if none of the transmission attempts in one sidelink period succeed, and we will discuss this situation in Section 4. When both of the following two conditions are met, the maximum value of T_{tx} shown in Figure 2(b) occurs:

(1) the message becomes available for transmission right after the scheduling cutoff time of the upcoming sidelink period K, so it must wait for the full length of the sidelink period K and is scheduled to be transmitted in the subsequent sidelink period $K + 1$;

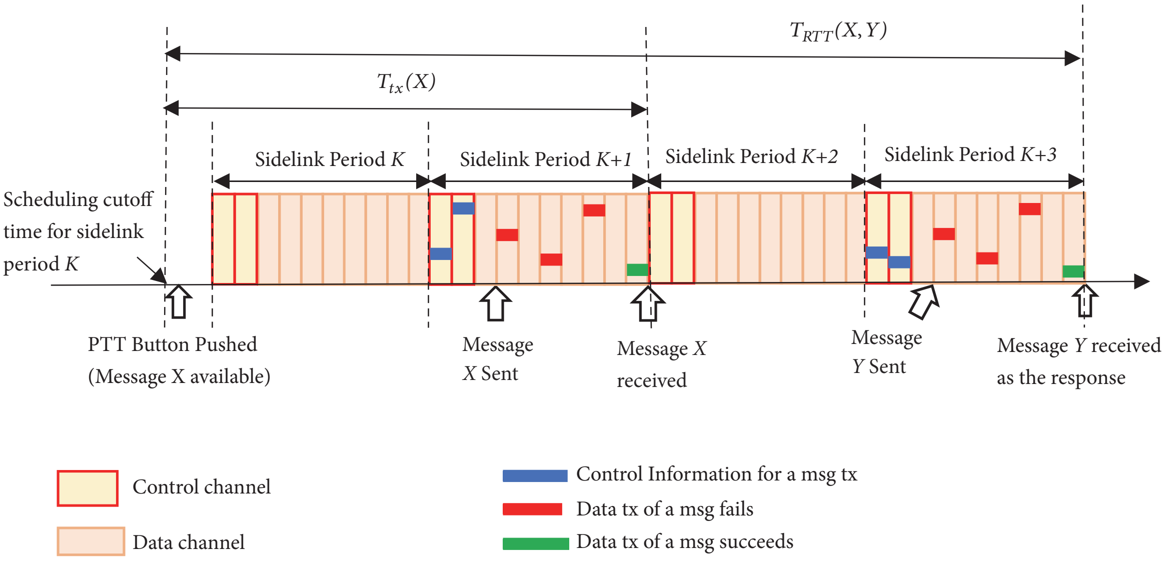
(2) the last transmission attempt of the message is scheduled to be transmitted in the last subframe of the data channel of the sidelink (e.g., $I_{TRP} = 7$) period.

Thus, the maximum value of T_{tx} in units of milliseconds is

$$\max(T_{tx}) = 3 + T_{delay} + 2 * T_{SL}; \tag{3}$$


$T_{RTT}(X,Y)$
$T_{tx}(X)$
Sidelink Period K
Sidelink Period $K+1$
Scheduling cutoff time for sidelink period K
PTT Button Pushed (Message X available)
Message X sent
Message X received
Message Y sent
Message Y received as the response
Control channel
Data channel
Control Information for a msg tx
Data tx of a msg fails
Data tx of a msg succeeds

(a) Minimum case



(b) Maximum case

FIGURE 2: Message TRANSMISSION TIME over ProSe T_{tx} and T_{RTT}.

Assume the l-th TB is the TB in which message X is assigned to be transmitted. Let T_{preSL} denote the time between the moment the PTT button at the transmitting UE is pushed and the beginning of the sidelink period in which the message is transmitted. There are four components of the message transmission time T_{tx}: (1) T_{preSL}; (2) the control channel duration T_{PSCCH}; (3) the time offset of the l-th TB from the beginning of the current instance of the data channel PSSCH which equals $(l-1) \times N_{TRP} \times N$; and (4) the duration from the beginning of the l-th TB to the time that message X is sent out successfully within the l-th TB. Note that for a message to be transmitted with ProSe parameters $I_{TRP} = i$ and assuming the message is transmitted successfully in the n-th attempt initially, component (4) equals $(n-1) \times N_{TRP} + i + T_{delay}$. This is because for each failed transmission attempt, an additional N_{TRP} ms delay will be added, and within each round of transmission attempt, there is i ms time offset introduced for a message transmitted with TRP bitmap parameter $I_{TRP} = i$. Therefore, the average of component (4) shall be calculated by conditioning on the possible I_{TRP} values i and, given i, by additionally conditioning on the index number of the first successful transmission attempt n. Also, since we are looking at the delay associated with a successful message transmission, we use the law of total probability and condition the value of component (4) on the event that the message transmission succeeds in exactly n attempts for $n = 1, 2, \ldots, N$; the probability of this event is $(1-p_n)\prod_{m=0}^{n-1} p_m$. Furthermore, the probability of message transmission failure is the probability of N transmission attempt failures, which is $\prod_{n=1}^{N} p_n$. Thus, the message transmission success probability is $1-\prod_{n=1}^{N} p_n$, and the average value of T_{tx} is

$$\begin{aligned}
Avg\left(T_{tx}\right) &= E\left(T_{preSL}\right) + T_{PSCCH} + (l-1) \times N_{TRP} \times N + \frac{1}{1-\prod_{n=1}^{N} p_n} \sum_{i=0}^{(N_{TRP}-1)} Prob\left(I_{TRP}=i\right) \\
&\quad \cdot \left[\sum_{n=1}^{N}\left(\left((n-1) \times N_{TRP} + i + T_{delay}\right)\left(1-p_n\right) \prod_{m=0}^{n-1} p_m\right)\right] \\
&= E\left(T_{preSL}\right) + T_{PSCCH} + (l-1) \times N_{TRP} \times N + \sum_{i=0}^{(N_{TRP}-1)} Prob\left(I_{TRP}=i\right) \\
&\quad \cdot \left[\left(i+T_{delay}\right) + \frac{N_{TRP}}{1-\prod_{n=1}^{N} p_n} \sum_{n=1}^{N-1}\left(n\left(1-p_{n+1}\right) \prod_{m=1}^{n} p_m\right)\right]
\end{aligned} \tag{4}$$

where $p_0 = 1$.

2.3. Roundtrip Transmission Time. The roundtrip message transmission time over the sidelink, denoted as $T_{RTT}(X, Y)$, is defined as the duration from when message X is available at UE A until UE A receives the response message Y from the corresponding UE (denoted as UE B for simplicity) successfully over the sidelink channel, then

$$\begin{aligned}
T_{RTT}(X, Y) = T_{tx}(X) &+ T_B + T_{boundary} \\
&+ \left(T_{tx}(Y) - T_{preSL}(Y)\right)
\end{aligned} \tag{5}$$

where T_B is the time between when message X is received by UE B and the time when the response message Y is available at UE B as defined by the protocol (e.g., backoff timers); and $T_{boundary}$ is the time that a message has to wait before the beginning of the sidelink period in which the message is transmitted. This overhead time is due to the sidelink period boundary restriction.

Note that $T_{RTT}(X, Y)$ is not the simple summation of $T_{tx}(X)$, T_B, and $T_{tx}(Y)$. This is because the available timing of message Y is dependent on the available timing of message X and T_B, which is no longer the independent PTT push timing that is assumed for UE B when calculating $T_{tx}(Y)$.

The minimum $T_{RTT}(X, Y)$ that UE A may expect is

$$\begin{aligned}
&\min\left\{T_{RTT}(X, Y)\right\} \\
&\quad = 4 + \left\lceil \frac{T_{PSCCH} + T_{delay} + \min\left\{T_B\right\} + 4}{T_{SL}} \right\rceil T_{SL} \\
&\qquad + T_{PSCCH},
\end{aligned} \tag{6}$$

as shown in Figure 2(a) by assuming $T_B = 0$.

As can be seen from (6), $\min\{T_{RTT}(X, Y)\}$ and thus $T_{RTT}(X, Y)$ is always greater than T_{SL}.

The maximum $T_{RTT}(X, Y)$ that UE A may experience is

$$\begin{aligned}
&\max\left\{T_{RTT}(X, Y)\right\} = 3 \\
&\quad + \left(1 + \left\lceil \frac{T_{PSCCH} + N \times N_{TRP} + T_{delay} + \max\left\{T_B\right\} + 3}{T_{SL}} \right\rceil\right) \\
&\quad \cdot T_{SL} + T_{PSCCH} + N \times N_{TRP} + T_{delay}
\end{aligned} \tag{7}$$

as shown in Figure 2(b) by assuming $T_B = 0$.

The calculation of average $T_{RTT}(X, Y)$ starts from rearranging (5) into mutually indepenedent components:

$$\begin{aligned}
Avg\left(T_{RTT}(X, Y)\right) &= E\left(T_{tx}(X) + T_B + T_{boundary} + \left(T_{tx}(Y) - T_{preSL}(Y)\right)\right) \\
&= E\left(T_{preSL}(X) + \left(T_{tx}(X) - T_{preSL}(X)\right) + T_B + T_{boundary} + \left(T_{tx}(Y) - T_{preSL}(Y)\right)\right) \\
&= E\left(T_{tx}(X) - T_{preSL}(X) + T_B + T_{boundary}\right) + E\left(T_{tx}(Y)\right) \\
&= T_{SL} * E\left(K_{RTT}\right) + E\left(T_{tx}(Y)\right)
\end{aligned} \tag{8}$$

where K_{RTT} is one less than the number of sidelink periods from the sidelink period in which message X is transmitted successfully to the sidelink period in which message Y is transmitted successfully, so $K_{RTT} = (T_{tx}(X) - T_{preSL}(X) + T_B + T_{boundary})/T_{SL}$. Similar to component (4) in (4), the average of K_{RTT} shall be calculated considering both possible I_{TRP} values i and possible numbering of the first successful transmission attempts n. In addition, the randomness introduced by T_B needs to be included as well.

$$\begin{aligned}
E\left(K_{RTT}\right) &= \left[\sum_{n=1}^{N}\left(\left(\left(1-p_n\right)\prod_{m=0}^{n-1}p_m\right)\sum_{k=1}^{K_{max}(n)}\left(k * Prob\left(K_{RTT}=k \mid \right.\right.\right.\right.\\
&\qquad \left.\left.\left.\left.\textit{the first successful transmission is attempt } n\right)\right)\right)\right] \times \frac{1}{1-\prod_{n=1}^{N}p_n}\\
&= \frac{1}{1-\prod_{n=1}^{N}p_n}\left\{\sum_{n=1}^{N}\left[\left(\left(1-p_n\right)\prod_{m=0}^{n-1}p_m\right)\sum_{k=1}^{K_{max}(n)}k\sum_{i=0}^{(N_{TRP}-1)}Prob\left(I_{TRP}=i\right)\right.\right.\\
&\qquad \left.\left.\times Prob\left(\left(k-1\right)T_{SL} \le \left(T_{PSCCH}+4+\left(n-1\right)\times N_{TRP}+\mathrm{i}+T_{delay}+T_B < k * T_{SL}\right)\right)\right]\right\}
\end{aligned} \tag{9}$$

where $p_0 = 1$; and $K_{max}(n) = \lceil (T_{PSCCH} + n \times N_{TRP} - 1 + T_{delay} + \max\{T_B\})/T_{SL} \rceil$ is the maximum possible K_{RTT} for a given n. By combining (8) and (9), the average value of $T_{RTT}(X, Y)$ is

$$\begin{aligned}
&Avg\left(T_{RTT}\left(X,Y\right)\right)\\
&\quad = \left[\sum_{n=1}^{N}\left(\left(\left(1-p_n\right)\prod_{m=0}^{n-1}p_m\right)\sum_{k=1}^{K_{max}(n)}k\sum_{i=0}^{(N_{TRP}-1)}Prob\left(I_{TRP}=i\right)\times Prob\left(T_{BL}\left(n,k,i\right)\le T_B < T_{BU}\left(n,k,i\right)\right)\right)\right]\\
&\quad \times \frac{T_{SL}}{1-\prod_{n=1}^{N}p_n} + Avg\left(T_{tx}\right)
\end{aligned} \tag{10}$$

where $Avg(T_{tx})$ is calculated by (4);

$$\begin{aligned}
T_{BL}\left(n,k,i\right) &= \left(k-1\right)T_{SL} - T_{PSCCH} - 4\\
&\quad - \left(n-1\right)N_{TRP} - i - T_{delay};\\
\text{and } T_{BU}\left(n,k,i\right) &= k * T_{SL} - T_{PSCCH} - 4\\
&\quad - \left(n-1\right)N_{TRP} - i - T_{delay}.
\end{aligned} \tag{11}$$

2.4. Baseline Configuration Parameters for Evaluation. Analytical estimates and simulation results presented in Sections 3 and 4 assume the baseline configuration shown in Table 1 unless otherwise noted. Note that the PER of a single transmission attempt is set at 0.1 at most, since 0.1 is the common design assumption of LTE PER when Hybrid automatic repeat request (HARQ) Acknowledgement (ACK)/Negative Acknowledgement (NACK) is in use. Given that there is no HARQ ACK/NACK over the ProSe PSSCH meaning that all four HARQ transmissions always occur, the PER should not be set to a more aggressive value higher than 0.1.

We assume in the simulations that the timing of a PTT button push is uniformly distributed within a sidelink period T_{SL}, and $l = 1$ (i.e., the message is transmitted in the first TB of the sidelink, which is reasonable given the importance of public safety related communication). It is also assumed that ProSe TRP parameters are configured as $k_{TRP} = 1$ and $N_{TRP} = 8$. This means that only one of the 8 subframes in a TRP will be used to send data and $Prob(I_{TRP} = i) = 1/N_{TRP} = 1/8$. Then (4) is simplified as

$$\begin{aligned}
Avg\left(T_{tx}\right) &= 7.5 + \frac{T_{SL}}{2} + T_{PSCCH} + T_{delay}\\
&\quad + \frac{8\left(p_1 + p_1p_2 + p_1p_2p_3 - 3p_1p_2p_3p_4\right)}{1 - p_1p_2p_3p_4}
\end{aligned} \tag{12}$$

in units of ms.

We also assume that the MCPTT user only needs to push the PTT button to initiate the request to speak and is not required to follow up the request with any further actions, such as acknowledgement. In addition, since there are some variations in MCPTT message sizes (e.g., call announcement message is long) and the size of resource allocated to each UE (e.g., the resource allocated for a voice message may be small to start with), our modeling assumes that MCPTT messages are not piggybacked, in order to get the conservative

Table 1: Baseline configuration parameters used in the MCPTT access time evaluation.

Parameter Name	Symbol	Value
Sidelink period (ms)	T_{SL}	40, 60, 80, 120, 160, 240, 280, 320
PSCCH duration (ms)	T_{PSCCH}	8
# transmission attempts in one sidelink period	N	4
Time Resource Pattern parameters	N_{TRP}	8
	k_{TRP}	1
PER of the 4 transmission attempts in each sidelink period	PER	[0.1, 0.1, 0.1, 0]
Transmission + Propagation delays (ms)	T_{delay}	0.93

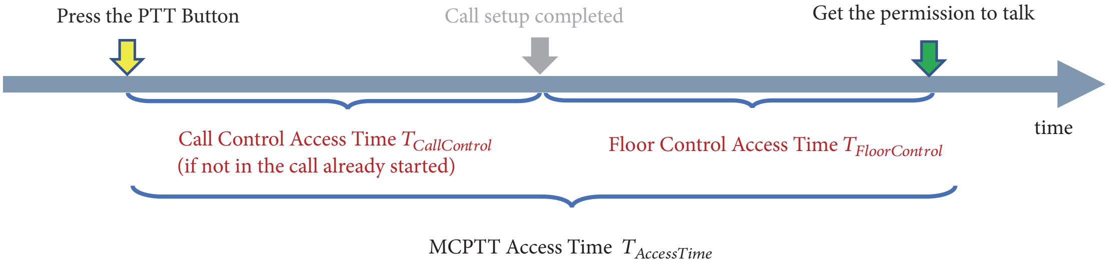

Press the PTT Button
Call setup completed
Get the permission to talk
time
Call Control Access Time $T_{CallControl}$
(if not in the call already started)
Floor Control Access Time $T_{FloorControl}$
MCPTT Access Time $T_{AccessTime}$

Figure 3: Components of MCPTT access time.

estimates/results. When piggybacking is allowed, the access time may be further reduced.

3. Modeling of MCPTT Access Time

3.1. Overview. According to 3GPP LTE specifications, "The MCPTT access time is defined as the time between when an MCPTT User requests to speak (normally by pressing the MCPTT control on the MCPTT UE) and when this user gets a signal to start speaking." [9]. Based on protocol details, the MCPTT access time ($T_{AccessTime}$) can be further divided into two components as illustrated in Figure 3: call control access time ($T_{CallControl}$) and floor control access time ($T_{FloorControl}$). Call Control Access Time ($T_{CallControl}$) is the time from when the MCPTT user presses the PTT button to when the MCPTT UE establishes a call successfully. This component occurs only when the call which the MCPTT UE wants to join/start does not exist yet. Floor control access time ($T_{FloorControl}$) is the time from when the MCPTT UE establishes a call successfully (if not already in a call) or when the MCPTT user presses the PTT button (if in a call already), whichever is later, to when the MCPTT UE becomes the floor arbitrator. In MCPTT off-network mode group call operation, the floor arbitrator is the current speaker who is the only group member that is allowed to speak to the group and that also arbitrates the floor in case other members of the group call (i.e., other floor participants) request that they be granted the floor so that they can talk.

MCPTT off-network mode supports the following three categories of calls: basic group call, broadcast group call, and private call. The analytical modeling of MCPTT access time associated with basic group call is analyzed first in Section 3.2, because the call control procedures and floor control procedures of the basic group call are very comprehensive. Then the analytical modeling of MCPTT access time associated with broadcast group call and private calls is presented in Sections 3.3 and 3.4, respectively, which reuses a subset of the modeling of basic group call procedures. We obtain simulation results assuming two users for all three call categories and compared these results with their analytical estimate counterparts in the corresponding sections. Simulation results presented throughout this paper are generated by running the ns-3 model for MCPTT over ProSe [17], especially the call control module and floor control module. Parameters configuration and simulation setup follow the description in Section 2.4, unless otherwise noted.

3.2. Basic Group Call. From the viewpoint of MCPTT access time performance evaluation, the various situations that an MCPTT client faces when pushing the PTT button to talk in a basic group call (with group ID *nn*) can always be categorized into one of the five scenarios summarized in Table 2.

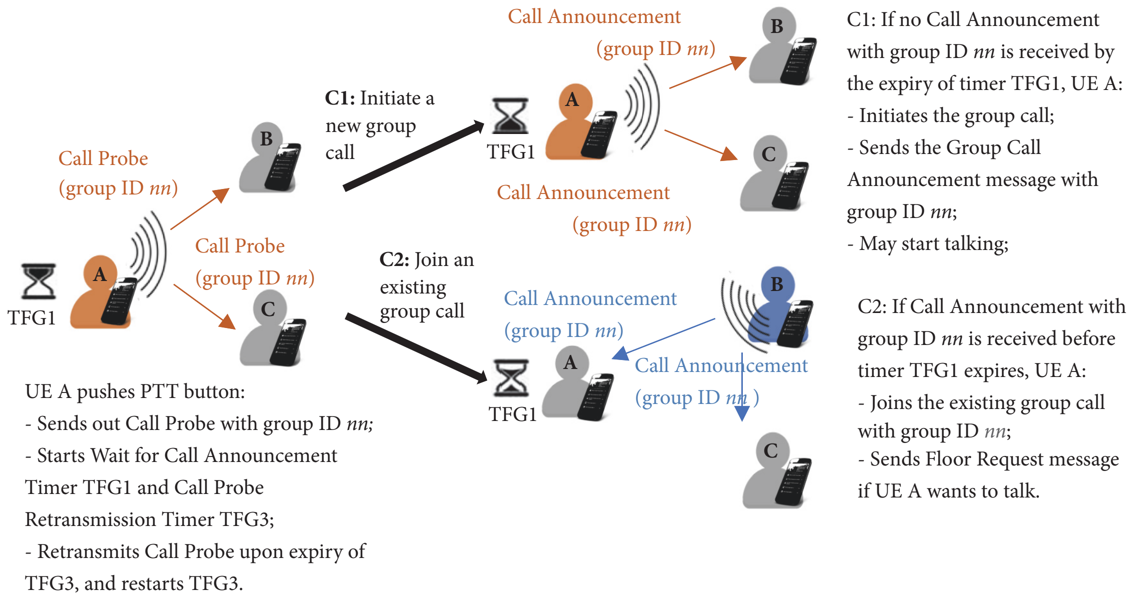
In Sections 3.2.1 and 3.2.2, we explain the call control and the floor control access procedures, and access time models are described. In Section 3.3 we derive the total access time $T_{AccessTime}$ experienced by an MCPTT client under Scenarios A through E from different combinations of $T_{CallControl}$ and $T_{FloorControl}$. Note that the value of $T_{FloorControl}$ may be 0 for certain scenario(s), as will be analyzed in Section 3.2.2.

3.2.1. Call Control Access Time $T_{CallControl}$. When an MCPTT client (UE A) wants to join/establish a basic group call with MCPTT group ID *nn* by pushing the PTT button, it has to go through either procedure "*C1: Initiate a new group call*" or procedure "*C2: Join an existing group call,*" which are shown in Figure 4. In the figure, the colors of the message name and the corresponding arrows are consistent with the color of the sending UE.

The call control access procedure we defined above has incorporated the call probe procedure, call setup procedure,

TABLE 2: Scenarios for basic group call.

Scenario	Group call *nn* exists?	UE A in group call *nn* already?	The floor arbitrator of group call *nn* exists?	$T_{AccessTime}$
A	No	No	No	$T_{CallControl} + T_{FloorControl}$
B	Yes	No	Yes	$T_{CallControl} + T_{FloorControl}$
C	Yes	No	No	$T_{CallControl} + T_{FloorControl}$
D	Yes	Yes	Yes	$T_{FloorControl}$
E	Yes	Yes	No	$T_{FloorControl}$



UE A pushes PTT button:
- Sends out Call Probe with group ID *nn*;
- Starts Wait for Call Announcement Timer TFG1 and Call Probe Retransmission Timer TFG3;
- Retransmits Call Probe upon expiry of TFG3, and restarts TFG3.

C1: If no Call Announcement with group ID *nn* is received by the expiry of timer TFG1, UE A:
- Initiates the group call;
- Sends the Group Call Announcement message with group ID *nn*;
- May start talking;

C2: If Call Announcement with group ID *nn* is received before timer TFG1 expires, UE A:
- Joins the existing group call with group ID *nn*;
- Sends Floor Request message if UE A wants to talk.

FIGURE 4: Call control access procedures.

and periodic group call announcement procedure specified in [18]. The decision to go through procedures C1 or C2 is made by UE A (the UE with the PTT button pushed) itself based on its own observation and may not be the correct decision. To be more specific, UE A always chooses C1 under Scenario A, which is the right decision. However, under Scenarios B or C, UE A might make the correct decision, C2, or the wrong decision, C1, depending on how soon UE A hears back from other UEs with respect to the configuration of timer TFG1.

When UE A executes the C1 procedure,

$$T_{CallControl}\left(\mathrm{C1}\right) = TFG1; \tag{13}$$

where $TFG1$ is the value of "wait for call announcement" timer $TFG1$.

When UE A executes the C2 procedure,

$$\begin{aligned} &T_{CallControl}\left(C2\right) \\ &\quad = T_{RTT}\left(\text{“}Call\ Probe\text{”}, \text{“}Call\ Announcement\text{”}\right); \end{aligned} \tag{14}$$

where T_{RTT} is given by (5) and where messages X and Y are the Call Probe and Call Announcement messages, respectively. In (5), $T_B = TFG2_{probe}(M)$, where M is the number of MCPTT UEs in the group call capable of sending out a Call Announcement message in response to receiving a Call Probe message from another UE. Here $TFG2_{probe}(M)$ reflects the time that it takes for at least one UE of the group to respond to a Call Announcement message after receiving the Call Probe message from UE A. It is a random variable that is the minimum of M random variables, each of which is uniformly distributed between 0 seconds and (1/12) seconds (83.3 ms). Therefore,

$$TFG2_{probe}\left(M\right) = \begin{cases} 0, & min \\ \dfrac{1000}{12\left(M+1\right)}, & avg \\ \dfrac{1000}{12}, & max \end{cases} \tag{15}$$

The minimum, maximum, and average of $T_{CallControl}(\mathrm{C2})$ can be obtained by plugging configuration values into (6), (7), and (10) accordingly.

The simulation study of call control access time in this section focuses on Scenarios B and C, i.e., Procedure C2. This is because there is no uncertainty in Scenario A (i.e., Procedure C1), since $T_{CallControl}$ always equals $TFG1$ when the MCPTT UE establishes a new group call.

Figure 5 shows that, for all sidelink periods T_{SL} defined by 3GPP for LTE ProSe, the analytical model provides good estimates of the minimum, average, and maximum for the

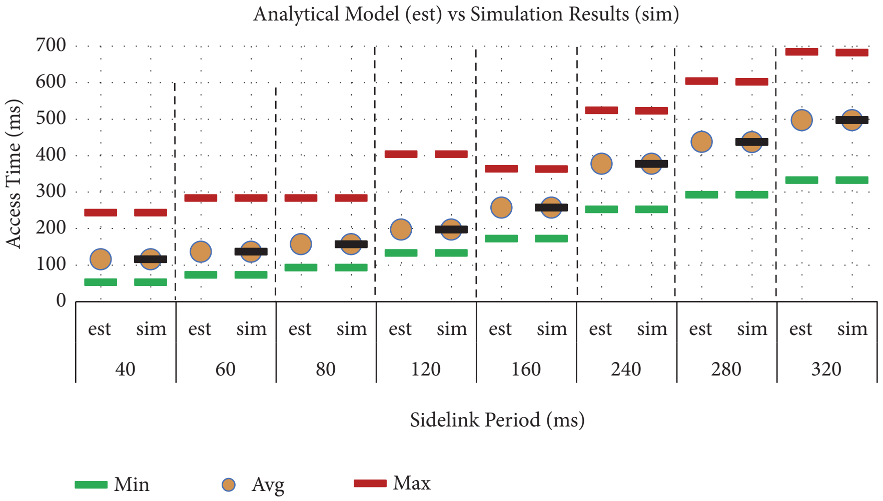


Figure 5: Call control access time for joining an existing group call with large value for $TFG1$ with 95% Confidence Intervals for the average values from the simulations.

call control access time experienced by an MCPTT client who goes through procedure C2. The 95% Confidence Interval for the average values obtained from simulation is also shown and is very tight. The minimum and maximum values are observed over all simulations and therefore do not have a Confidence Interval. This is true for all the figures shown in this document. As can be seen from the figure, $T_{CallControl}(C2)$ increases as T_{SL} increases and ranges from as small as 52 ms to more than 680 ms. The $TFG1$ timer value in the simulation is configured to max (1 sec, $6 * T_{SL}$) and the $TFG3$ timer value is configured to $6 * T_{SL}$; thus UE A will make the correct decision because neither timer will expire prematurely.

3.2.2. Floor Control Access Time $T_{FloorControl}$. When an MCPTT client wants to talk and thus pushes the PTT button on UE A, the UE will go through one of the "*F0: new call,*" "*F1: No Response,*" or "*F2: Response Received*" procedures depending on the feedback that it receives from other UEs.

After the MCPTT client (UE A) establishes a group call (i.e., C1 procedure), UE A sends out a Floor Granted message and may start talking, so the corresponding floor control access procedure is as follows.

F0. Send out Floor Granted message to grant the floor to oneself.

After an MCPTT client (UE A) joins an existing group call (i.e., C2 procedure), or if UE A is already a part of an ongoing group call, UE A sends out a Floor Request message. Depending on whether a response is received within a configured time window, UE A may perform the F1 or F2 procedure as illustrated in Figure 6. In the figure, the colors of the message name and the corresponding arrows are consistent with the color of the sending UE. Depending on the relative priority of the floor request and the queueing capability of the current floor arbitrator, procedure F2 may be further divided into three cases:

F2.1 (preemptive floor request). UE A's floor request has preemptive priority, so UE B must grant the floor to UE A immediately;

F2.2 (floor request denied). UE B is busy and does not support queueing requests, so UE B denies the floor to UE A;

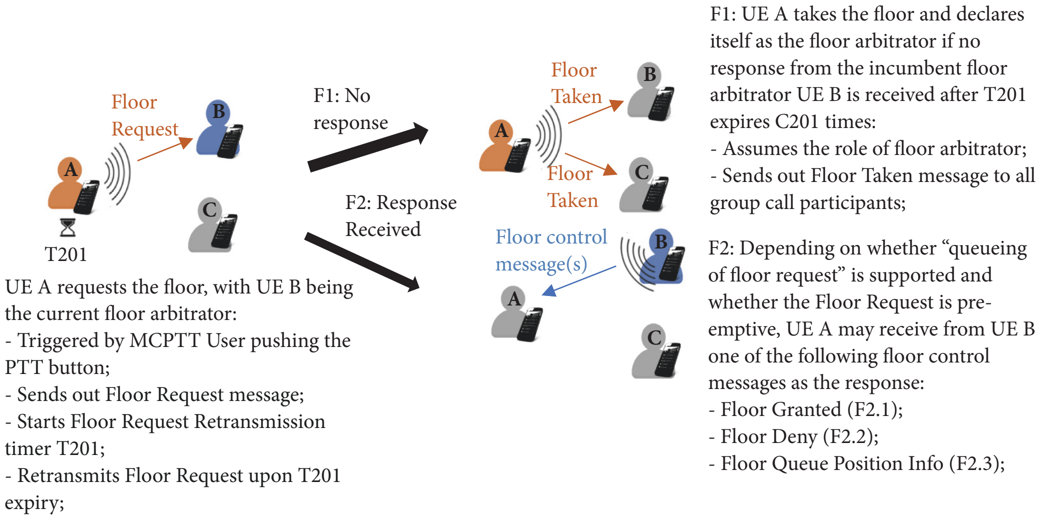
F2.3 (floor request queued). UE B is busy but supports queueing of requests, so UE A's floor request is placed in queue and will be handled later.

It is worth pointing out that the time it takes for UE A to receive a response is the same for F2.1, F2.2, and F2.3. However, the 3GPP definition of MCPTT access time focuses on the case of "when this user gets a signal to start speaking" [2]. Therefore, in the following analytical model and simulation study, among the three cases of procedure F2, we focus on procedure "F2.1: preemptive floor request". It is because procedure F2.2 or F2.3 will not grant the floor to UE A, and thus UE A's access procedures are not successful. The model and results can be easily generalized if the MCPTT access time definition is rewritten as "when this user gets a signal whether he/she can start speaking".

The floor control access procedure we defined above has incorporated several state transition procedures specified in [19]. The decision to go through procedure F0, F1, or F2 is made by UE A (the UE with the pushed PTT button) based on its own observation, and this decision may or may not be the correct one. To be more specific, UE A always chooses F1 under Scenario A, which is the right decision. However, under Scenarios B or D, UE A might make the correct decision (F2) or the wrong decision (F1), depending the time until UE A receives a response message from the floor arbitrator UE B and the configuration values of the $T201$ timer and $C201$ counter. The floor control access time of each of the three procedures are as follows:

When F0 procedure is executed,

$$T_{FloorControl}(F0) = 0; \tag{16}$$



UE A requests the floor, with UE B being the current floor arbitrator:
- Triggered by MCPTT User pushing the PTT button;
- Sends out Floor Request message;
- Starts Floor Request Retransmission timer T201;
- Retransmits Floor Request upon T201 expiry;

F1: UE A takes the floor and declares itself as the floor arbitrator if no response from the incumbent floor arbitrator UE B is received after T201 expires C201 times:
- Assumes the role of floor arbitrator;
- Sends out Floor Taken message to all group call participants;

F2: Depending on whether "queueing of floor request" is supported and whether the Floor Request is pre-emptive, UE A may receive from UE B one of the following floor control messages as the response:
- Floor Granted (F2.1);
- Floor Deny (F2.2);
- Floor Queue Position Info (F2.3);

Figure 6: Floor control access procedure.

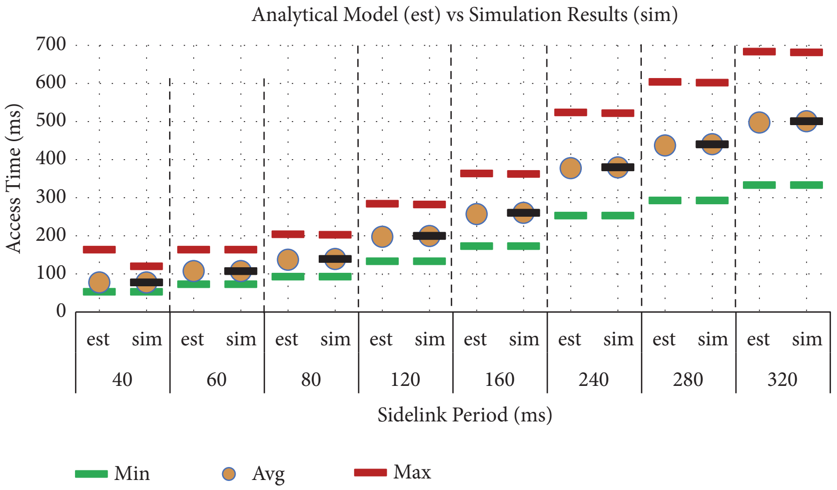


Figure 7: Floor control access time for preemptive floor request with 95% Confidence Intervals for the average values from the simulations.

When F1 procedure is executed,

$$T_{FloorControl}\left(F1\right) = C201 * T201; \qquad (17)$$

where $T201$ is the "Floor Request Retransmission" timer and $C201$ is the "Floor Request Retransmission" counter. The 3GPP default settings of these two parameters are T_{SL} and 3, respectively.

When the F2.1 procedure is executed,

$$\begin{aligned} & T_{FloorControl}\left(F2.1\right) \\ & \quad = T_{RTT}\left(\left(\text{“}Floor\ Request\text{”}, \text{“}Floor\ Granted\text{”}\right)\right); \end{aligned} \qquad (18)$$

again using (5), where $T_B = 0$. The minimum, maximum, and average of $T_{FloorControl}(F2.1)$ can be obtained by plugging configuration values into (6), (7), and (10) accordingly.

The simulation study of floor control access time in this section focuses on either Scenario B or D, i.e., procedure F2.1. This is because there is no uncertainty associated with $T_{FloorControl}$ for Scenarios A, C or E: $T_{FloorControl} = C201 * T201$ for Scenarios C & E (i.e., procedure F1), and $T_{FloorControl} = 0$ for Scenario A (i.e., procedure F0). Also, we focus on the case when UE A's floor request is of preemptive priority, i.e., F2.1, as explained in Section 2.4. In the simulation, it is assumed that the incumbent floor arbitrator UE B does not generate Real-time Transport Protocol (RTP) packets during UE A's access time. This assumption will not impact UE A's access time when UE A makes the right decision. We make this assumption to get a conservative estimate of the likelihood of UE A's making the wrong decision, which leads to an undesirable situation where there are multiple floor arbitrators. More detailed definition and analysis of the multiple floor arbitrator situation are given in Section 4.3.

Figure 7 shows that for all sidelink period lengths T_{SL} defined by LTE, the analytical model provides good estimates of the minimum, average and maximum floor control access times experienced by an MCPTT client who goes through procedure F2.1. The $T201$ timer value in the simulation is

Table 3: MCPTT access time $T_{AccessTime}$ for different scenarios.

Scenario	Correct decision	Wrong decision
A	$T_{CallControl}(C1) + T_{FloorControl}(F0)$	N/A
B	$T_{CallControl}(C2) + T_{FloorControl}(F2)$	$T_{CallControl}(C2) + T_{FloorControl}(F1)$ or $T_{CallControl}(C1) + T_{FloorControl}(F0)$
C	$T_{CallControl}(C2) + T_{FloorControl}(F1)$	$T_{CallControl}(C1) + T_{FloorControl}(F0)$
D	$T_{FloorControl}(F2)$	$T_{FloorControl}(F1)$
E	$T_{FloorControl}(F1)$	N/A

configured to $6 * T_{SL}$, so that UE A will make the correct decision.

Comparing Figure 7 with Figure 5, it shows that although the minimum call control access time $T_{CallControl}(C2)$ and floor control access time $T_{FloorControl}(F2.1)$ are very similar, the average and maximum values of $T_{FloorControl}(F2.1)$ are smaller than those of $T_{CallControl}(C2)$ for short sidelink periods (small T_{SL} values). This is because of the extra delay introduced by timer $TFG2$ to the round-trip time during the call control procedure. When T_{SL} is large, however, the impact of $TFG2$ becomes negligible.

3.2.3. Group Call MCPTT Access Time $T_{AccessTime}$. The analytical model of the overall access times for scenarios A through E can be obtained by the proper combination of the analytical models of $T_{CallControl}$ and $T_{FloorControl}$ derived in previous sections, as shown in Table 3. It is worth pointing out that for Scenario A, $T_{FloorControl}(F0)$ is 0 as was explained in Section 3.2.2.

It is worth pointing out that, for Scenarios B, C, and D, UE A may make the correct decision or the wrong decision, depending on the MCPTT parameters' configuration and the one-way or roundtrip message transmission time over the sidelink. When UE A makes the correct decision, the analytical models presented in Section 3 align quite well with the MCPTT access time observed in our simulations. When UE A makes wrong decisions, the simulation results deviate from the analytical estimates, which we analyze in Section 4, together with the discussion of potential negative impacts from incorrect decisions.

3.3. Broadcast Group Call. A broadcast group call is a special group call where the initiating MCPTT user expects no response from the other MCPTT users, so that when the initiator's transmission is complete, so is the call. Therefore, the UE that initiated the broadcast group call is the only MCPTT UE acting as the floor arbitrator. Consequently, only one scenario needs to be evaluated to examine the access time required to establish a new broadcast group call:

Scenario F. UE A establishes a new broadcast group call by pushing the PTT button.

The value of $T_{CallControl}$ under Scenario F is zero, because the UE can initiate a broadcast group call without waiting for timer expiration or messages from other UEs. Similarly, the value of $T_{FloorControl}$ is also zero, because the UE that initiates a broadcast group call becomes the floor arbitrator immediately. Consequently, the overall access time is:

$$T_{AccessTime} = 0; \tag{19}$$

3.4. Private Call. A private call takes place between two MCPTT UEs. For the private call scenarios, a MCPTT UE is in one of two possible states: it is in a private call already or needs to start a new private call; a UE cannot join an existing private call. Consequently, there are only two scenarios that need to be evaluated to obtain the access time of a private call:

Scenario G. UE A establishes a new private call by pushing the PTT button.

Scenario H. UE A is already in a private call and wants to talk by pushing the PTT button.

For Scenario G, the possible call control procedures that UE A may experience are depicted in Figure 8. In the figure, the colors of the message name and the corresponding arrows are consistent with the color of the sending UE. To establish a private call successfully, procedure CP2.1 is the only path, and then UE A becomes the floor arbitrator automatically. Therefore, the overall access time for Scenario G is

$$\begin{aligned} T_{AccessTime} &= T_{CallControl}(CP2.1) \\ &= T_{RTT}(\text{“}Private\ Call\ Setup\ Request\text{”}, \\ &\quad \text{“}Private\ Call\ Accept\text{”}); \end{aligned} \tag{20}$$

which uses (5) to evaluate T_{RTT}. The minimum, maximum, and average of $T_{CallControl}(CP2.1)$ can be obtained by plugging configuration values into (6), (7), and (10), respectively, with $T_B = 0$

Under Scenario G, when the time for UE A to receive a response is greater than $TFP1 * CFP1$, UE A will make the conclusion that the other UE (UE B) cannot be reached and stop trying to establish the call, i.e., procedure CP1.

For Scenario H, the floor control procedure that UE A may experience is very similar to the F2.1 procedure, and the overall access time for Scenario H is from (18)

$$\begin{aligned} T_{AccessTime} &= T_{FloorControl}(F2.1) \\ &= T_{RTT}(\text{“}Floor\ Request\text{”}, \text{“}Floor\ Granted\text{”}); \end{aligned} \tag{21}$$

The simulation study of private call access time in this section focuses on Scenario G, i.e., procedure CP2.1. This

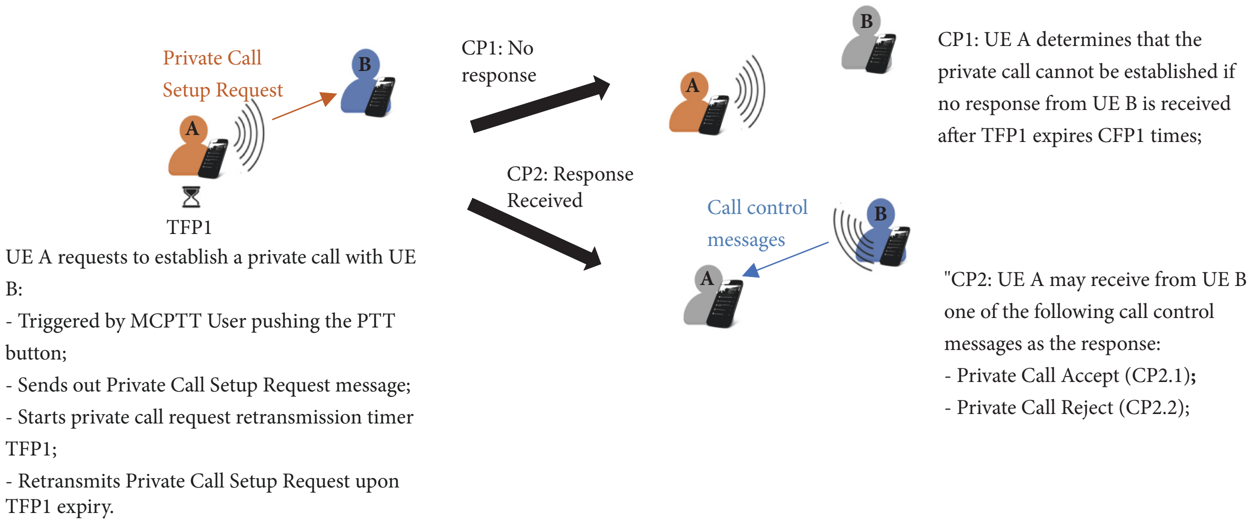

Private Call
Setup Request
A
B
TFP1
UE A requests to establish a private call with UE B:
- Triggered by MCPTT User pushing the PTT button;
- Sends out Private Call Setup Request message;
- Starts private call request retransmission timer TFP1;
- Retransmits Private Call Setup Request upon TFP1 expiry.
CP1: No response
CP2: Response Received
A
B
CP1: UE A determines that the private call cannot be established if no response from UE B is received after TFP1 expires CFP1 times;
Call control messages
A
B
"CP2: UE A may receive from UE B one of the following call control messages as the response:
- Private Call Accept (CP2.1);
- Private Call Reject (CP2.2);

Figure 8: Call control access procedure of private call.

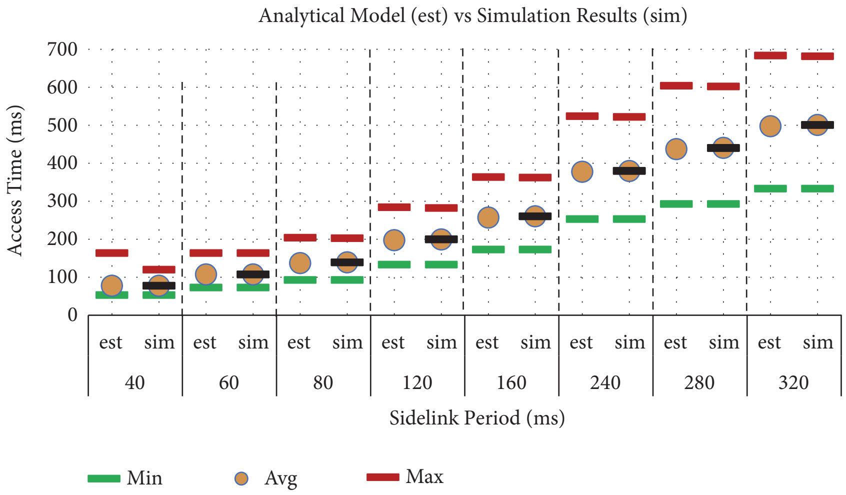


Figure 9: Access time of private call with 95% Confidence Intervals for the average values from the simulations.

is because the access time of Scenario H can use the results for F2.1 presented in Section 3.2.2. Figure 9 shows that, for all sidelink period lengths T_{SL} defined by 3GPP, the analytical model provides good estimates of the minimum, average, and maximum for access times experienced by an MCPTT client going through procedure CP2.1. We observe that the access time and the spread between the minimum and the maximum values increases as T_{SL} increases, for both analytical estimates and simulation results.

4. Impact of MCPTT Parameters on MCPTT Access Time

As already mentioned briefly in Section 3, several important observations can be made based on the analytical and simulation modeling of MCPTT access time.

Observation 1. An MCPTT UE needs more than one sidelink period to send a message and get the response message back over the sidelink.

As can be seen from (6), $\min\{T_{RTT}(X,Y)\}$ and thus $T_{RTT}(X,Y)$ is always longer than T_{SL}. However, the default 3GPP settings of several MCPTT retransmission timers, e.g., $TFG3$ and $T201$, is one T_{SL} only. Such a small timer value may lead to unnecessary retransmissions of certain MCPTT messages and may have negative impacts as illustrated in Sections 4.1 and 4.3.

Observation 2. An MCPTT UE may make wrong decisions about the existence of call and/or floor arbitrators, under some scenarios and with certain parameter configurations.

As summarized in Table 3, UE A's decision may go wrong for Scenarios B, C, and D. These wrong decisions may lead to undesirable situations such as a multiple calls situation or a

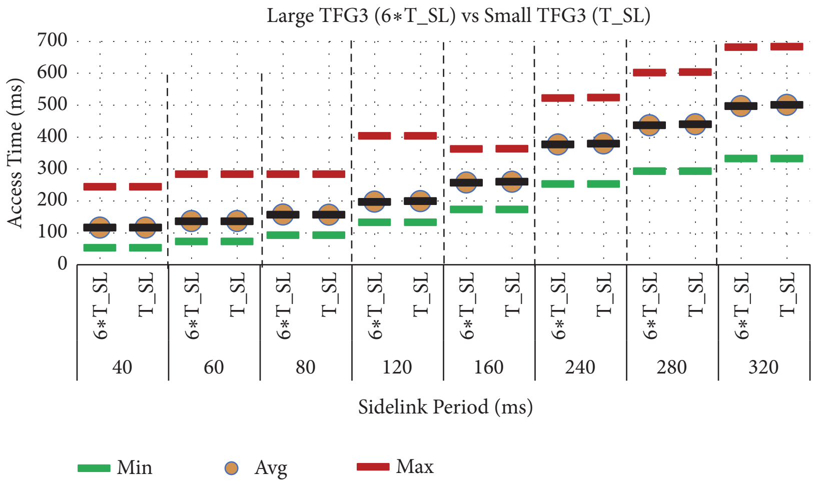


Figure 10: Impact of *TFG*3 Timer on call control access time with 95% Confidence Intervals for the average values from the simulations.

Table 4: Impact of *TFG*3 configuration.

	T_{SL} (ms)	40	60	80	120	160	240	280	320
Ratio of multiple occupancy to single occupancy	$TFG3 = T_{SL}$	0.510	0.667	0.670	0.576	0.573	0.524	0.530	0.534
	$TFG3 = 6 * T_{SL}$	0	0	0	0	0	0	0	0
Fraction of samples discarded	$TFG3 = T_{SL}$	0.132	0.140	0.092	0.059	0.052	0.024	0.025	0.026
	$TFG3 = 6 * T_{SL}$	0	0	0	0	0	0	0	0

multiple floor arbitrators situation, both of which are further analyzed in Sections 4.2 and 4.3, respectively.

Motivated by the above observations, in the remainder of this section, we study the impact of various MCPTT parameters to evaluate the benefits and drawbacks of different configurations on MCPTT access time performance.

4.1. Impact of "Call Probe Retransmission" Timer TFG3.

To evaluate the impact of *TFG*3, we look at the call control access time for Scenarios B and C (group calls). Figure 10 shows the results using the same assumptions as for Figure 5 but with *TFG*3 configured at the 3GPP default setting T_{SL} instead of $6 * T_{SL}$.

It appears that varying *TFG*3 from T_{SL} to $6 * T_{SL}$ produces no significant difference in the call control access time experienced by an MCPTT client. To further investigate the impact of *TFG*3, we define two other performance metrics called "multiple occupancy" and "single occupancy" to capture the impact of the half duplex effect, which can lead to packet losses at the receiving UE. "Multiple occupancy" refers to a situation in which multiple UEs try to send messages over the air during the same sidelink period, while "single occupancy" refers to a situation in which only one UE tries to send messages over the air during a sidelink period. The half-duplex effect will not occur with "single occupancy" but may occur with "multiple occupancy". As shown in Table 4, the negative impact of *TFG*3's configuration value's being too small can be seen in the higher ratio of "multiple occupancy" vs "single occupancy," which may lead to more frequent occurrences of the half duplex effect. This is mainly due to unnecessary retransmissions of the "Call Probe" message. The negative impact of *TFG*3's being too small is also shown through the nonzero ratio of samples discarded. Note that a sample is discarded if both SCI message transmissions in the control channel (PSCCH) fail or all four HARQ transmissions in the data channel (PSSCH) fail, due to either half-duplex effect or channel PER. Given PER = [0.1, 0.1, 0.1, 0] as a simulation assumption, all samples discarded are always the consequence of the half duplex effect. The fraction of samples discarded is dependent on the size of the resource pools allocated to sidelink communication. The positive impact of a small *TFG*3 is that the "Call Probe" message may be retransmitted sooner if it is lost.

It is worth pointing out that the occurrence of "multiple occupancy" and thus the half-duplex effect are expected to be more frequent when the number of participating UEs in the group call increases. Therefore, although we do not see any noticeable difference in access time when changing *TFG*3 value in the 2-UE system, the impact of *TFG*3 will become more obvious for multiple-UE systems. In addition, the occurrence of the half-duplex effect can be reduced by adapting resource pool configuration to the number of participating UEs (e.g., extending the pool in the time domain), which is an interesting research topic by itself.

4.2. Impact of "Wait for Call Announcement" Timer TFG1.

When we run the simulation with the same assumptions as for Figure 5 but with *TFG*1 configured at the 3GPP default setting of 150 ms instead of $\max(1s, 6 * T_{SL})$, we can see from

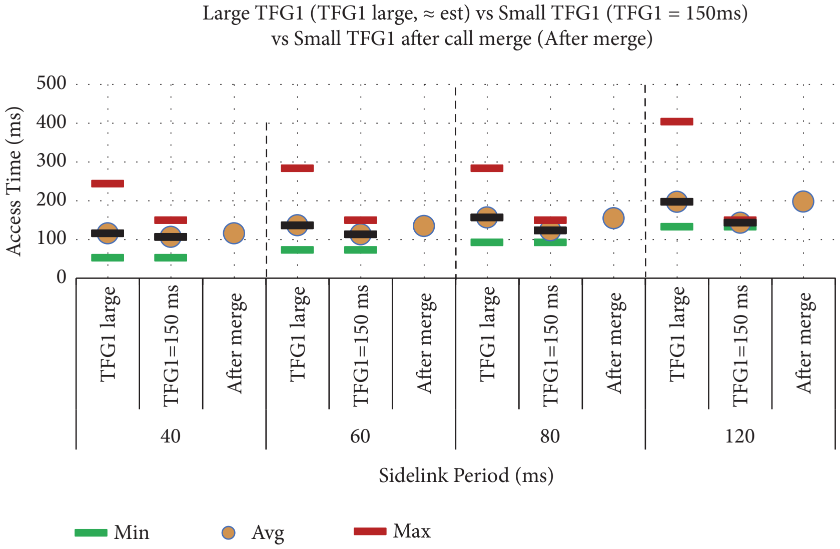


Figure 11: Impact of TFG1 on call control access time with 95% Confidence Intervals for the average values from the simulations.

Figure 11 that the maximum access times experienced by an MCPTT client are reduced and capped at 150 ms, and thus the average access time is reduced. As we explained in Section 3.2.1, when $TFG1$ expires, UE A assumes that the call of interest does not exist and will establish a new group call by itself, which marks the end of the MCPTT access time duration. However, given that the scenario simulated is B or C, i.e., group call nn exists already, these $TFG1$ expirations are premature, and the capped maximum access time might not be an improvement. Indeed, the MCPTT client, UE A, makes the wrong decision when establishing a new group call instead of joining the existing call. Consequently, the MCPTT client is not aware of the existing call, and multiple calls of the same group ID coexist. This undesirable situation of multiple calls with the same group ID will be referred to as the "multiple calls situation" for the remainder of this paper. The multiple calls situation will persist until the call merging procedure is triggered. After taking into account the extra delay introduced by the necessary call merging procedure, the average effective call control access time is very close to the results that we obtained when $TFG1$ is large, as seen in Figure 11, so the analytical model gives a good prediction of the effective call control access time after the call merge procedure. Simulation results with a sidelink period longer than 120 ms are not included in Figure 11 because UE A always makes the wrong decision in that case and, thus, the call merging procedure is always triggered to fix the multiple calls situation.

The disadvantages of the multiple calls situation include not only the extra delay due to the call merging procedure, but also the potential false impression/feedback to the MCPTT client. The MCPTT user, who is very likely a first responder on an urgent task, might think that no teammate is within communication range, although some teammates may be deployed nearby. Similarly, the MCPTT user may try to switch groups if he/she is notified wrongly that the group does not have an existing call. As another example, a first responder may decide to take actions at a higher risk, instead of waiting for support/backup from other teammates when in a time-sensitive or life-threatening situation. Therefore, although a smaller $TFG1$ configuration may reduce the wait before establishing a new call if the call does not exist, the 3GPP default setting of $TFG1 = 150$ ms is worth further discussion to reduce the occurrence of the multiple calls situation, and it may be desirable to take T_{SL} into account when configuring $TFG1$.

The negative impact of the multiple calls situation becomes more pronounced when the channel conditions degrade. Figure 12 compares the impact of channel loss on the call control access time when timer $TFG1$ is configured with the 3GPP default setting. When the channel PER of the fourth transmission attempt, $p4$, in the Sidelink Shared Channel increases from 0 (lossless) to 0.1 (lossy), $T_{CallControl}$ remains similar. This is because the potential long delay due to message loss is mostly absorbed by the expiration of $TFG1$. However, the value of $T_{CallControl}$ discussed above is only the nominal access time and does not consider that the MCPTT client might make the wrong decision to establish a new group call when $TFG1$ expires. When comparing the effective access time experienced by the MCPTT client, i.e., considering the extra delay introduced by the call merging procedure wherever applicable, the average effective $T_{CallControl}$ value has increased considerably due to the channel loss. The extra delay observed, which is quite large, is mainly due to the slow periodicity of the Call Announcement transmission cycle when the MCPTT UE is not responding to a Call Probe. When a Call Probe message is lost, or the responding Call Announcement message is lost, a UE is more likely to wait for the periodic Call Announcement to trigger a call merging procedure, which is slower.

Table 5 shows the probability of the "multiple calls situation" occurrence if $TFG1$ is set to 150 ms. Because of the potential packet loss due to the unnecessary retransmission of MCPTT messages, an error situation is more likely to occur

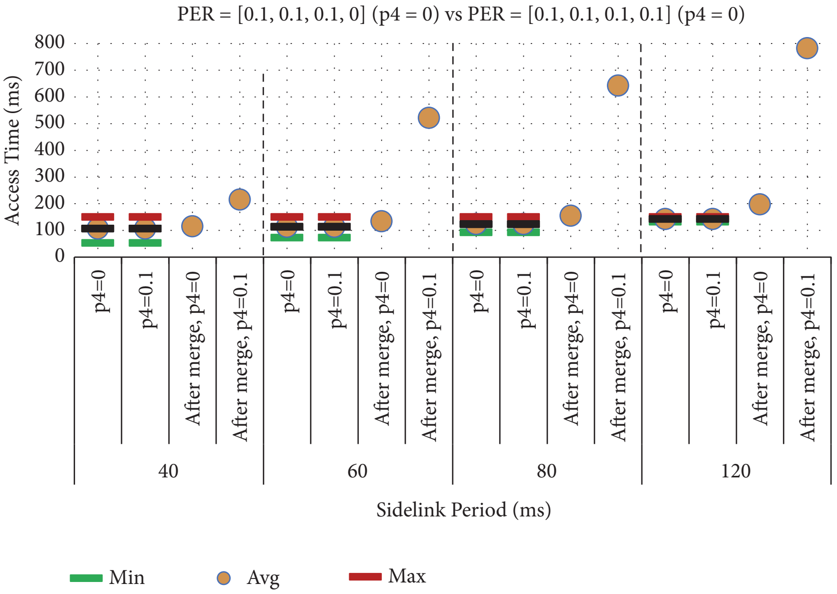

PER = [0.1, 0.1, 0.1, 0] (p4 = 0) vs PER = [0.1, 0.1, 0.1, 0.1] (p4 = 0)

Figure 12: Impact of channel loss on call control access time with TFG1 = 150 ms with 95% Confidence Intervals for the average values from the simulations.

Table 5: Probability of “Multiple Calls Situation” for various $TFG1$ configurations and channel conditions ($TFG1$ = 150 ms).

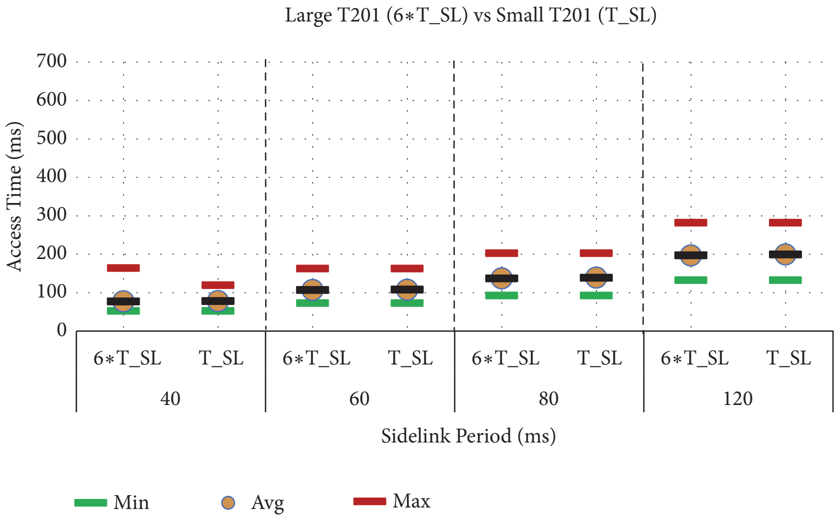
		T_{SL}(ms)	40	60	80	120
Probability of the occurrence of “multiple calls situation”	$PER = [0.1, 0.1, 0.1, 0]$	$TFG3 = 6 * T_{SL}$	0.169	0.384	0.504	0.894
		$TFG3 = T_{SL}$	0.176	0.378	0.513	0.904
	$PER = [0.1, 0.1, 0.1, 0.1]$	$TFG3 = T_{SL}$	0.186	0.362	0.525	0.917

when $TFG3$ is set to T_{SL} than when it is set to $6 * T_{SL}$. When the channel condition is worse, the occurrence rate of the “multiple calls situation” also increases, as shown in the last two lines of Table 5.

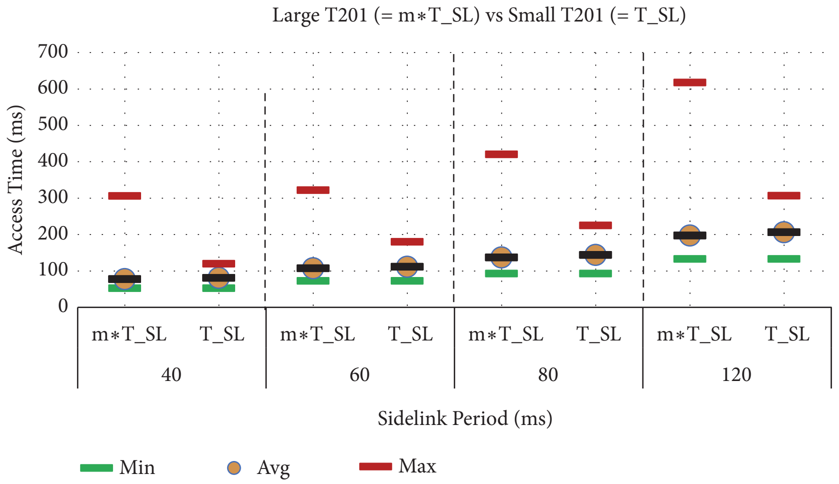
4.3. Impact of Floor Request Retransmission T201. The impacts of a smaller value for the Floor Request Retransmission Timer, $T201$, such as the 3GPP default setting T_{SL}, are multifold. On the positive side, if there is no floor arbitrator present, a smaller value for $T201$ enables the MCPTT user to declare the floor sooner, with the access time equal to $T201*C201$, than when $T201$ is larger (e.g., $6 * T_{SL}$) as shown in Figure 13(a), or when $T201$'s value is based on the provision of the near worst case as shown in Figure 13(b). In the case shown in Figure 13(b), $T201 = m * T_{SL}$, where m is the smallest integer which is greater than $\max\{T_{FloorControl}\ (F2.1)\}/T_{SL}$, which can be calculated based on (18). By comparing Figures 13(a) and 13(b), we can see that the channel loss makes the positive impact of a small $T201$ configuration more pronounced. Note that we obtained the results in Figure 13(a) by running the simulation with the same assumption as for Figure 7, except that we configured $T201$ at the 3GPP default setting T_{SL} instead of at $6 * T_{SL}$. We obtained the results of Figure 13(b) by running the simulation with the same assumption as for Figure 7, except that we set PER to [0.1, 0.1, 0.1, 0.1] instead of [0.1, 0.1, 0.1, 0], and we configured $T201$ at either T_{SL} or $m * T_{SL}$.

However, on the negative side, the decision made by the MCPTT client (e.g., assume the floor is idle) with a smaller $T201$ value might be premature, and it may lead to an undesirable situation in which there are multiple floor arbitrators, which we will refer to as the “multiple floor arbitrators situation”. Note that this situation is neither easy nor quick to resolve, given the current procedures. Table 6 shows the increase in ratio of “wrong decision” vs “correct decision” when comparing small $T201$ settings with large $T201$ settings. It is also clear from comparing Table 6(a) to Table 6(b) that the channel loss makes the negative impact of a small $T201$ configuration even more evident. Note that the results in Table 6 show the worst case outcome because we assumed that the incumbent floor arbitrator UE B does not generate RTP packets during UE A's access time. If the current floor arbitrator UE B is actively sending RTP packets, the counter C201 at UE A is reset every time an RTP packet is received. Therefore, it will take more time and will be less likely for UE A's $C201$ to reach the threshold and make a wrong decision when UE B is talking.

We encounter the “multiple floor arbitrators situation” when there is more than one floor arbitrator simultaneously in the same off-network group call, although the correct behavior is that there should be at most one floor arbitrator



(a) Access time when $T201 = 6 * T_{SL}$, PER = [0.1, 0.1, 0.1, 0] with 95% Confidence Intervals for the average values from the simulations



(b) Access time when $T201 = m * T_{SL}$, PER = [0.1, 0.1, 0.1, 0.1] with 95% Confidence Intervals for the average values from the simulations

Figure 13: Impact of $T201$ on floor control access time.

Table 6: Impact of $T201$ configuration—potential multiple floor arbitrators situation.

(a) PER = [0.1, 0.1, 0.1, 0]

	T_{SL} (ms)	40	60	80	120
Ratio of "premature $T201$ expiration case" (i.e., the wrong decision case) vs "the correct decision case"	$T201 = T_{SL}$	0.026	0.018	0.0007	0.0004
	$T201 = 6 * T_{SL}$	0	0	0	0

(b) PER = [0.1, 0.1, 0.1, 0.1]

	T_{SL} (ms)	40	60	80	120
Ratio of "premature $T201$ expiration case" (i.e., the wrong decision case) vs "the correct decision case"	$T201 = T_{SL}$	0.126	0.145	0.075	0.060
	$T201 = m * T_{SL}$	0	0	0	0

at any time. There are several circumstances that may lead to the "multiple floor arbitrators situation," such as a premature $T201$ timer expiration or the hidden node problem. The potential disadvantages of the multiple floor arbitrators situation are as follows:

(i) The lack of resolution procedure for "multiple floor arbitrators situation." This is in contrast to the "multiple group calls situation," for which the call merging procedure is already defined in the call control protocol as a remedy.

Table 7: Impact of $T201$ Configuration—potential half duplex effect.

	T_{SL} (ms)	40	60	80	120	160	240	280	320
Ratio of double occupancy vs single occupancy	$T201 = T_{SL}$	0.603	0.723	0.581	0.575	0.573	0.523	0.529	0.534
	$T201 = 6 * T_{SL}$	0	0	0	0	0	0	0	0
Ratio of samples discarded	$T201 = T_{SL}$	0.112	0.126	0.071	0.058	0.051	0.024	0.025	0.025
	$T201 = 6 * T_{SL}$	0	0	0	0	0	0	0	0

Table 8: Average MCPTT access time in milliseconds with 3GPP default settings.

T_{SL} (ms)	40	60	80	120	160	240	280	320
Scenario A	150	150	150	150	150	150	150	150
Scenario B	184.46	221.95	263.45	342.81	Call Merge procedure is always triggered			
Scenario C	226.43	293.82	364.47	503.17	Call Merge procedure is always triggered			
Scenario D	78.03	108.13	138.98	199.63	260.22	379.80	440.32	500.85
Scenario E	120	180	240	360	480	720	840	960

(ii) The MCPTT client who takes the floor prematurely is not aware of the situation and might start talking while not knowing that no one is listening;

(iii) The incumbent floor arbitrator is not aware of the situation either and does not know that one or more group members are not listening.

As was the case in the call control study, the negative impact of the Floor Request Retransmission Timer $T201$ being configured too small can be noticed in Table 7 and is mainly due to the unnecessary retransmissions of the Floor Request message.

When comparing the same performance metrics of the $TFG1 = T_{SL}$ case in Table 4 with the $T201 = T_{SL}$ case in Table 7, the call control procedure seems to perform slightly better than its floor control counterpart. This is because the $TFG2_{probe}$ timer adds a random delay to the transmission timing of the responding Call Announcement message. Consequently, the transmission of the Call Announcement message does not always compete with the retransmission of a Call Probe message for resources in the same sidelink period.

4.4. MCPTT Access Time with 3GPP Default Settings.

Table 8 summarizes the simulation results of the average MCPTT access time in milliseconds for different scenarios when the MCPTT parameters are configured according to the 3GPP default settings.

We obtained the values in the table assuming no hardware or software processing time. In addition, we obtained the access times when UE A makes the correct decision as described in Table 3. However, wrong decisions are not uncommon occurrences with the 3GPP default settings, which might lead to the "multiple floor arbitrators situation" and/or the situation of multiple calls with the same group ID.

5. Conclusions and Future Work

We developed the analytical models and simulation tools described in this paper to estimate and evaluate the performance and user experience of MCPTT off-network mode operations over ProSe in LTE, with a focus on access time when an MCPTT UE is out of coverage. We defined performance metrics and application scenarios for the access time analysis. Our analytical models provide good estimates of the minimum, average, and maximum for both call control and floor control access times experienced by an MCPTT client under various scenarios, as shown by our simulation results. We summarize our main observations with respect to parameter configurations in Table 9. These observations may be used to understand the tradeoff and provide configuration guidelines for MCPTT and ProSe.

The sensitivity analysis results show that preventive solutions, such as proper parameters configuration and/or modified procedures, need to be investigated to reduce the possibility of an MCPTT UE's making wrong decisions. In addition, 3GPP may re-evaluate the default setting of timers TFG1, TFG3, and T201 due to the undesirable occurrence of the 'multiple floor arbitrators situation,' the coexistence of multiple group calls with the same group ID, and the relatively high probability of the half duplex effect.

Our future work includes enhancing the access time analysis to consider more aspects related to ProSe, such as a probability model of the half duplex effect in the PSCCH. In addition, we will evaluate the MCPTT access time performance for the scenarios when multiple active UEs try to talk at nearly the same time, or there are multiple ongoing group calls competing for radio resources over ProSe.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Table 9: Impact of parameters configuration on MCPTT access time.

Parameter	Small	Big

T_{SL}	(i) Shorter access time;
	(ii) More overhead;
		(i) Longer access time;
		(ii) Less overhead;

$TFG3$	(i) Faster retransmissions of Call Probe; (ii) More unnecessary retransmissions of Call Probe (e.g., with 3GPP default configuration) are likely;	(i) Reduced unnecessary retransmissions of Call Probe; (ii) Necessary Call Probe retransmission may be slowed down;
$TFG1$	(i) Reduced nominal maximum and average access time; (ii) The call merge procedure is more likely to be triggered (e.g., with 3GPP default configuration), and result in increased effective maximum and average access time;	(i) Nominal maximum and average access time may be larger; (ii) Reduced unnecessary call merge procedure triggering, and thus reduced effective maximum and average access time;
$T201$& ($C201 * T201$)	(i) Faster determination of idle floor; (ii) The "multiple floor arbitrators situation" is more likely to occur (e.g., with 3GPP default configuration); (iii) More unnecessary retransmissions of Floor Request (e.g., with 3GPP default configuration) are likely.	(i) Slower determination of idle floor; (ii) The "multiple floor arbitrators situation" can be reduced ($C201*T201$); (iii) Reduced unnecessary retransmissions of Floor Request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

[1] National Public Safety Telecommunications Council (NPSTC), "Mission Critical Voice Communications Requirements for Public Safety," 2011.

[2] National Public Safety Council Telecommunications, "Public Safety Broadband Push-to-Talk over Long Term Evolution Requirements," Tech. Rep., 2013.

[3] A. Rebeiro-Hargrave, *OMA Push-to-Talk Architecture, Multimedia Group Communication*, Chichester, UK, pp. 23–57, 2AD, 2008.

[4] D. Ma, W. Huang, Q. Sun, W. Li, C. Yin, and Z. Shao, "Design and realization of the poc system with short session-setup-delay," in *Proceedings of the 2012 International Conference on Computer Science and Service System, CSSS 2012*, pp. 1075–1078, August 2012.

[5] E. Casini, N. Suri, M. Breedy, P. Budulas, J. Kovach, and R. Roy, "PeerTalk: A push-to-talk and instant messaging service for tactical networks," in *Proceedings of the 2013 IEEE Military Communications Conference, MILCOM 2013*, pp. 1476–1481, November 2013.

[6] M.-H. Tsai and Y.-B. Lin, "Talk burst control for push-to-talk over cellular," *IEEE Transactions on Wireless Communications*, vol. 7, no. 7, pp. 2612–2618, 2008.

[7] A. Kuwadekar and K. Al-Begain, "An evaluation of push to talk service over ims and lte for public safety systems," in *Proceedings of the 2014 6th International Conference on Computational Intelligence and Communication Networks, CICN 2014*, pp. 412–416, November 2014.

[8] Open Mobile Alliance (OMA), *Enabler Release Definition for Push to Communicate for Public Safety (PCPS)*, 2017.

[9] 3GPP, "Technical Specification Group Services and System Aspects; Mission Critical Push To Talk (MCPTT) over LTE; Stage 1 (Release 14); TS 22.179," Tech. Rep., 2016.

[10] J. Kim, O. Jo, and S. W. Choi, "State-based uplink-scheduling scheme for reducing control plane latency of MCPTT services," *IEEE Systems Journal*, 2018.

[11] R. Solozabal, A. Sanchoyerto, E. Atxutegi, B. Blanco, J. O. Fajardo, and F. Liberal, "Exploitation of mobile edge computing in 5G distributed mission-critical push-to-talk service deployment," *IEEE Access*, vol. 6, pp. 37665–37675, 2018.

[12] 3GPP, "Technical Specification Group Core Network and Terminals; Proximity services (ProSe) User Equipment (UE) to ProSe Function Protocol Aspects; Stage 3 (Release 14); TS 24.334," Tech. Rep., 2018.

[13] J. Wang and R. Rouil, "Assessing coverage and throughput for D2D communication," in *Proceedings of the 2018 IEEE International Conference on Communications, ICC 2018*, May 2018.

[14] D. W. Griffith, F. J. Cintron, and R. A. Rouil, "Physical sidelink control channel (PSCCH) in mode 2: performance analysis," in *Proceedings of the 2017 IEEE International Conference on Communications, ICC 2017*, May 2017.

[15] D. Griffith, F. Cintrón, A. Galazka, T. Hall, and R. Rouil, "Modeling and simulation analysis of the physical sidelink shared channel (PSSCH)," in *Proceedings of the 2018 IEEE International Conference on Communications, ICC 2018*, May 2018.

[16] A. M. Cipriano and D. Panaitopol, "Performance analysis of sidelink data communications in autonomous mode," in *Proceedings of the 2018 IEEE Wireless Communications and Networking Conference, WCNC 2018*, pp. 1–6, April 2018.

[17] National Institute of Standards and Technology, "Public Safety Communication modeling tools based on ns-3," 2018. https://github.com/usnistgov/psc-ns3.

[18] 3GPP, "Technical Specification Group Core Networks and Terminals; Mission Critical Push To Talk (MCPTT) call control; Protocol specification (Release 14); TS 24.379," Tech. Rep., 2018.

[19] 3GPP, "Mission Critical Push To Talk (MCPTT) media plane control; Protocol specification (Release 14); TS 24.380," Tech. Rep., 2018.

[20] P. Varin, Y. Sun, and W. Garey, “NISTIR 8236 Test Scenarios for Mission Critical Push-To-Talk (MCPTT) Off-Network Mode Protocols Implementation,” Tech. Rep., 2018.

[21] 3GPP, “Technical Specification Group Radio Access Network; Evolved Universal Terrestrial Radio Access (E-UTRA); Physical layer procedures (Release 14); TS 36.213,” Tech. Rep., 2018.