**3GPP TSG-RAN WG2 Meeting #115 electronic *R2-210xxxx***

**Online, August 16th – August 27th, 2021**

**Agenda Item: 9.2.3**

**Source: OPPO**

**Title: Summary of [AT115-e][037][IoT-NTN] User Plane Impact (OPPO)**

**Document for: Discussion and Decision**

# Introduction

This document aims to summarize the following offline discussion.

* [AT115-e][037][IoT-NTN] User Plane Impact (OPPO)

 Scope: Treat documents under 9.2.3. Identify potential agreements (e.g. confirm SI agreements), Open points, potential alternatives.

 Intended outcome: Report

 Deadline: CB Monday W2

# Discussion

Proposals from companies [1-10] will be discussed in the following sections.

## 2.1 MAC

### 2.1.1 RACH

In NTN, due to the large propagation delay, during random access procedure, after sending Msg1, UE could not receive Msg2 until a time interval of RTT has passed. In Rel-17 NR NTN WI, RAN1 has made the following agreements in RAN1#105 e-meeting:

|  |
| --- |
| Agreement:The starts of ra-ResponseWindow and msgB-ResponseWindow are delayed by an estimate of UE-gNB RTT. * The estimate of UE-gNB RTT is equal to the sum of UE’s TA and K\_mac.

Note 1: The UE’s TA is based on the RAN1#104bis-e agreement on Timing Advance applied by an NR NTN UE given by  $T\_{TA}=\left(N\_{TA}+N\_{TA, UE-specific}+N\_{TA,common}+N\_{TA,offset}\right)×T\_{c}$. The estimate of gNB-satellite RTT is equal to the sum of and K\_mac.  How to treat $N\_{TA}$ and $N\_{TA,offset}$ can be further discussed.Note 2: According to the RAN1#104bis-e agreement: When UE is not provided by network with a K\_mac value, UE assumes K\_mac = 0.Note 3: The accuracy of the estimated UE-gNB RTT with respect to the true UE-gNB RTT can be further discussed.Note 4: Other options of determining the estimate of UE-gNB RTT can be further discussed. |

Based on above RAN1 agreements, in NR NTN, an offset is used to delay to start of ra-ResponseWindow, and the offset is the estimated UE-gNB RTT.

In [1], [2], [3], [7], [9] and [10], it is proposed to introduce an offset to delay the start of the ra-ResponseWindow for IoT-NTN. Regarding the offset value, there are different options:

* **Option 1**: Derive the offset based on UE-eNB RTT
	+ **Option 1-1**: The offset is defined as max (current offset, UE-eNB RTT), where the current offset is fixed to 3 subframes for eMTC, and can be either 4 subframes or 41 subframes for NB-IoT as defined in TS36.321. [1]
	+ **Option 1-2**: The offset is an estimate of UE-eNB RTT, which aligns with NR NTN. [7] [9]
* **Option 2**: Postpone the discussion on offset to starting of RA response window until further agreements regarding RACH are made in RAN1. [3][5]

It seems that all the companies have a common understanding that the start of the ra-ResponseWindow should be delayed by an offset. Rapporteur would like to ask the following question:

**Question 1: If the start of the ra-ResponseWindow is delayed by an offset, which is your preferred option regarding the offset value?**

* **Option 1**: **Derive the offset based on UE-eNB RTT**
	+ **Option 1-1: The offset is defined as max (current offset, UE-eNB RTT), where the current offset is fixed to 3 subframes for eMTC, and can be either 4 subframes or 41 subframes for NB-IoT as defined in TS36.321.**
	+ **Option 1-2: The offset is an estimate of UE-eNB RTT.**
* **Option 2: Postpone the discussion on offset to starting of RA response window until further agreements regarding RACH are made in RAN1.**

|  |  |  |
| --- | --- | --- |
| **Company** | **Option** | **Additional comments** |
| OPPO | Option 1-1 | For NB-IoT and eMTC, there is already an offset for the start of ra-ResponseWindow. For eMTC, the offset is fixed to 3 subframes, while for NB-IoT, the offset can be either 4 subframes or 41 subframes, which depends on the NPRACH transmission duration.The UE-eNB RTT for different IoT NTN scenarios are as below.

|  |  |  |
| --- | --- | --- |
| Scenarios | Maximum UE-eNB RTT | Minimum UE-eNB RTT |
| GEO (Scenario A) | 541.46ms | 520.86 |
| LEO with satellite altitude of 600 km(Scenario B&C) | 19.53ms | 25.77 ms |
| LEO with satellite altitude of 1200 km(Scenario B&C) | 35.41ms | 41.77ms |
| MEO(Scenario D) | 160.1 ms | 186.9 ms |

It can be observed that for eMTC the current offset value is shorter than UE-eNB RTT for all the IoT NTN scenarios. For NB-IoT, the current offset value may be shorter or longer than UE-eNB RTT. Therefore, a straight way is to adjust the offset for the start of ra-ResponseWindow based on UE-eNB RTT, e.g., the offset for the start of ra-ResponseWindow can defined as max (current offset, UE-eNB RTT). |
| Xiaomi | Option 1-1 | In case X=41, the delay is not negligible, thus it deserves to optimize for this case, i.e. using maximum {X, UE-eNB RTT}. |
| CATT | See comments  | The current offset mentioned in option 1-1 is applied to PRACH repetition mechanism. The offset is introduced based on the decoding capacity of network. However, the UE-eNB RTT is propagation delay in NTN. Therefore, the offset is defined as max (current offset, UE-eNB RTT) is not correct.Thus, we agree with the following option:Option 3: An offset is defined as sum (current offset, UE-eNB RTT). is introduced at the start of RAR Window, where the current offset is fixed to 3 subframes for eMTC, and can be either 4 subframes or 41 subframes for NB-IoT as defined in TS36.321. |
| Huawei, HiSilicon | Option 2 |  |
| MediaTek | Option 1-1 | Agree with Oppo and Xiaomi. |
| Qualcomm | Option 2 | Let the RAN1 first confirm calculation of the UE-eNB RTT as in NR NTN. |
| Lenovo | Option 2 | We would like to have RAN1’s conclusion first. |
| Nokia | Option 1-2 or Option2 | In WID, RAN1 agreed to take the time and frequency synchronization agreements in NR NTN as baseline for IoT NTN. Option1-2 follow NR NTN agreement on the start of RAR window. Furthermore, what we understand about Option1-2 is an additional offset on top of current offset defined in TS36.321.For Option 1-1, the question is NW may not know the exact UE-gNB RTT before RACH thus don’t know when UE will monitor RAR.We are also fine to wait for RAN1 conclusion first if it is the majority view. |
|  |  |  |
|  |  |  |

**Rapporteur summary:**

TBA…

Regarding ra-ResponseWindowSize length, the following agreement was made in RAN2#112e for NR NTN:

Agreements:

1. If the start of the ra-ResponseWindow and msgB-ResponseWindow is accurately compensated by UE-gNB RTT, ra-ResponseWindow and msgB-ResponseWindow are not extended in LEO/GEO.

In [3], [7], [8] and [9], it is proposed that if the start of the RA Response window is accurately compensated by UE-eNB RTT and no extension of repetition is required, there is no need to extend the ra-ResponseWindowSize for IoT NTN.

**Question 2: Do companies agree that if the start of the RA Response window is accurately compensated by UE-eNB RTT and no extension of repetition is required, there is no need to extend the ra-ResponseWindowSize for IoT NTN?**

|  |  |  |
| --- | --- | --- |
| **Company** | **Agree / Disagree** | **Additional comments** |
| OPPO | Agree |  |
| Xiaomi | yes |  |
| CATT | Agree |  |
| Huawei, HiSilicon | Agree |  |
| MediaTek | Agree |  |
| Qualcomm | Agree |  |
| Lenovo | Agree |  |
| Nokia | Agree |  |
|  |  |  |

**Rapporteur summary:**

TBA…

In RAN2#115e, the following agreement has been made in NR NTN WI:

Agreements:

1. In the MAC specification section 5.1.5, delay the start of ra-ContentionResolutionTimer by the UE-gNB RTT (i.e. sum of UE's TA and K\_mac)

In [1], [2], [3], [7], [9] and [10], it is proposed to introduce an offset to delay the start of the mac-ContentionResolutionTimer for IoT-NTN. Regarding the offset value, in [1] and [9], it is further proposed that the offset value should be set to UE-eNB RTT. In [7], it is suggested that the exact meaning of the offset should follow NR NTN agreement. On the other hand, in [3] and [5], it is proposed to postpone the discussion on offset to starting of mac-ContentionResolutionTimer until further agreements regarding RACH are made in RAN1. In rapporteur’s understanding, since the start of mac-ContentionResolutionTimer is captured in TS36.321, this issue should be addressed in RAN2. However, it would be good to collect companies’ views.

**Question 3: If the start of the mac-ContentionResolutionTimer is delayed by an offset, which is your preferred option regarding the offset value?**

* **Option 1: use UE-eNB RTT as the offset value**
* **Option 2: Postpone the discussion on offset to starting of mac-ContentionResolutionTimer until further agreements regarding RACH are made in RAN1.**

|  |  |  |
| --- | --- | --- |
| **Company** | **option** | **Additional comments** |
| OPPO | Option 1 | It could align with conclusion in NR NTN. |
| Xiaomi | Option 1 |  |
| CATT | Option 1 |  |
| Huawei, HiSilicon | Option 2 |  |
| MediaTek | Option 1 |  |
| Qualcomm | Option 2 |  |
| Lenovo | Option 2 | We would like to have RAN1’s conclusion first. |
| Nokia | Option 1 | As RAN2-115 agreed that, for NR NTN, the offset to start ra-ContentionResolutionTimer is UE-gNB RTT, we think Option1 is agreeable to follow NR NTN agreements. |
|  |  |  |
|  |  |  |
|  |  |  |

**Rapporteur summary:**

TBA…

Regarding the mac-ContentionResolutionTimer length, considering that the offset for the start of mac-ContentionResolutionTimer can be accurately compensated by UE-eNB RTT, it is proposed in [9] that mac-ContentionResolutionTimer is not extended.

**Question 4: Do companies agree that there is no need to extend the mac-ContentionResolutionTimer for IoT NTN?**

|  |  |  |
| --- | --- | --- |
| **Company** | **Agree / Disagree** | **Additional comments** |
| OPPO | Agree with comment | If the start of mac-ContentionResolutionTimer is accurately compensated by UE-eNB RTT and no extension of repetition is required, there is no need to extend the mac-ContentionResolutionTimer for IoT NTN |
| Xiaomi | yes |  |
| CATT | Agree |  |
| Huawei, HiSilicon | Agree |  |
| MediaTek | Agree |  |
| Qualcomm | Agree |  |
| Lenovo | Agree |  |
| Nokia | Agree with modification | Same view as OPPO. |
|  |  |  |
|  |  |  |

**Rapporteur summary:**

TBA…

If we agree to use UE-eNB RTT as the start of some UP timers (e.g. ra-ResponseWindow, mac-ContentionResolutionTimer), the next issue is how to determine UE-eNB RTT.

In NR NTN WI, based on RAN1 agreement in RAN1#105e, the estimate of UE-gNB RTT is equal to the sum of UE’s TA and K\_mac, while K\_mac is needed when downlink timing and uplink timing are not aligned at gNB, in which case this parameter donotes the TA value pre-compensated by gNB and can be provided by gNB.

It is proposed in [1] that UE estimates UE-eNB RTT in a similar way as in NR NTN, i.e., the estimated UE-eNB RTT is the sum of UE’s TA and K\_mac. In [3], [7] and [9], it is suggested to wait for RAN1 on how UE obtains UE-NB RTT.

**Question 5: Please provide your views on how UE obtains UE-eNB RTT in IoT NTN?**

* **Option 1: UE determines UE-eNB RTT in a similar way as in NR NTN, i.e., the UE-eNB RTT is the sum of UE’s TA and K\_mac**
* **Option 2: wait for RAN1 on how UE obtains UE-NB RTT.**

|  |  |  |
| --- | --- | --- |
| **Company** | **Option** | **Additional comments** |
| OPPO | Option 1 | In our understanding, the method for UE-eNB RTT estimation in NR NTN could also apply to IoT NTN. However, if most companies prefer option 2, we are also ok.  |
| Xiaomi | Option 1 |  |
| CATT | Option 1 |  |
| Huawei, HiSilicon | Option 2 |  |
| MediaTek | Option 1 |  |
| Qualcomm | Option 2 | In NR NTN, it was RAN1 who decided to use UE’s TA and K\_mac for UE-gNB RTT. At least we can wait if RAN1 confirms same for IoT NTN. |
| Lenovo | Option 2 | We would like to have RAN1’s conclusion first. |
| Nokia | Option 2 | In WID, RAN1 may have minimum change on how to obtain UE-gNB RTT for IoT NTN based on NR NTN agreements. |
|  |  |  |
|  |  |  |
|  |  |  |

**Rapporteur summary:**

TBA…

For UE with TA pre-compensation capability, when scheduling Msg3’s transmission for the UE, network may not know the exact value of UE-specific TA, due to UE’s autonomous pre-compensation for Msg1 transmission. This can be solved by using the maximum propagation delay of the cell to schedule Msg3’s transmission.

In RAN2#112e meeting, the following agreement was made in NR NTN WI.

Agreements:

1. From RAN2 perspective, for UE with UE-specific pre-compensation as a baseline it is up to gNB implementation to ensure sufficient time on UE side for the Msg3 transmission.

In [3], it is proposed that the above agreement also applies to IoT NTN.

**Question 6: Do companies agree that from RAN2 perspective, for UE with UE-specific pre-compensation as a baseline it is up to eNB implementation to ensure sufficient time on UE side for the Msg3 transmission for IoT NTN?**

|  |  |  |
| --- | --- | --- |
| **Company** | **Agree / Disagree** | **Additional comments** |
| OPPO | Agree | It could align with conclusion in NR NTN. |
| Xiaomi | yes |  |
| CATT | Agree |  |
| Huawei, HiSilicon | Agree |  |
| MediaTek | Agree |  |
| Qualcomm | Agree |  |
| Lenovo | Agree |  |
| Nokia | Agree |  |
|  |  |  |
|  |  |  |

**Rapporteur summary:**

TBA…

TA report during RACH has been discussed in NR NTN WI. The motivation of introducing UE-calculated TA report during RACH is to let gNB know the absolute TA value of the UE, so that gNB could utilize UE’s absolute TA to schedule UL transmission for the UE, which could reduce unnecessary latency. So far, RAN2 has reached the following agreement on TA report.

Agreement in RAN2#104e:

1. If enabled by the network, the UE reports information about UE specific TA pre-compensation at the random access procedure (MSGA/MSG3 or MSG5) using a MAC CE. Actual content is FFS and also depends on further RAN1 input (we can revise this whole agreement if RAN1 come to a different conclusion in terms of what needs to be conveyed to the NW)

Agreement in RAN2#105e:

1. UE specific TA reporting during RACH procedure is enabled/disabled by SI (FFS for RACH in connected mode)

TA reporting was not discussed during the SI for IoT NTN. In [3], it is proposed to wait for RAN1 to decide whether reporting of the UE specific TA pre-compensation is supported in IoT NTN. However, rapporteur understands that RAN2 could address this TA reporting issue similar as for NR NTN. Anyway, it would be good to collect companies’ views.

**Question 7: Do companies agree to support TA report during RACH in IoT NTN?**

* **Option 1: Yes**
* **Option 2: No**
* **Option 3: wait for RAN1 progress.**

|  |  |  |
| --- | --- | --- |
| **Company** | **Option** | **Additional comments** |
| OPPO | Option 1 | We could reuse conclusion in NR NTN as baseline. |
| Xiaomi | Option 2 | For IOT, delay is not an issue, no need to report TA in RACH procedure. |
| CATT | Option 1 | In NR NTN, reporting UE-specific TA is to assist network for the following scheduling. Thus, this method maybe beneficial for scheduling in IOT NTN. |
| Huawei, HiSilicon | Option 3 |  |
| MediaTek | Option 1 | We could reuse conclusion in NR NTN as baseline. |
| Qualcomm | Option 1 | In NR NTN also, RAN2 made agreement on this. |
| Lenovo | Opyion 1 | Align with NR NTN. |
| Nokia | Option 3 | The topic is listed as RAN1 objective in IoT NTN WID (as below). No need to duplicate the discussion in RAN1 and RAN2."Signalling aspects in UE-specific TA maintenance and reporting, techniques to reduce the signalling load and determination of the UE-specific TA." |

**Rapporteur summary:**

TBA…

### 2.1.2 (UL) HARQ RTT Timer

SI agreements for HARQ-RTT-Timer and UL-HARQ-RTT-Timer enhancement in IoT NTN has been captured in TR36.763, as below.

|  |
| --- |
| As the challenges associated with the expiry of MAC timers in NR NTN [3] remain the same in IoT NTN, it is assumed that the same solutions as NR NTN for the start of DL HARQ RTT Timer and UL HARQ RTT Timer can be reused as a baseline to support IoT NTN. |

In NR NTN WI, RAN2 has made the following agreement on DL/UL HARQ RTT Timer:

Agreement in RAN2#112-e:

|  |
| --- |
| 1. For UE with pre-compensation capability (at least for the HARQ-feedback enabled case. FFS for HARQ-feedback disabled, if supported), drx-HARQ-RTT-TimerDL is offset by UE-specific RTT (UE-gNB delay) in LEO/GEO. FFS if offset is applied to: 1) the start of the timers or 2) the timer value range (i.e. existing values within value range increased by offset)
 |

Agreement in RAN2#113-e:

|  |
| --- |
| 1. For HARQ processes with DL HARQ feedback enabled, drx-HARQ-RTT-TimerDL length is increased by offset (i.e. existing values within value range increased by offset). RAN2 working assumption: offset is equal to UE-gNB RTT (if RAN1 decides something that requires to change this we can revisit it) (RAN2#113-e)
 |

Agreement in RAN2#114-e:

|  |
| --- |
| 1. The following options are supported for drx-HARQ-RTT-TimerUL in NTN per HARQ process: 1) Timer length is extended by offset; 2) Timer set to zero and/or 3) Timer disabled (i.e. not started). FFS if this is based on explicit configuration or not. We can also come back to see whether both 2 and 3 are needed.
2. RAN2 working assumption: Offset for drx-HARQ-RTT-TimerUL is equal to UE-gNB RTT (if RAN1 decides something that requires to change this we can revisit it).
 |

In TR36.763, it is described that “*It was concluded that from a physical layer perspective, there is no consensus on disabling HARQ feedback for NTN IoT in Rel-17*”. Since disabling HARQ feedback is not supported for IoT NTN, adaptation of HARQ RTT Timer and/or UL HARQ RTT timer are needed.

Unlike NR, for eMTC and NB-IoT, both HARQ RTT timer and UL HARQ RTT Timer are derived by a formula in TS36.321. In general, the formula can be expressed as below.

HARQ RTT Timer = TPDSCH to UCI + TUCI + Tprocessing + delta PDCCH

UL HARQ RTT Timer = Tprocessing + delta PDCCH

Where,

* TPDSCH to UCI is the time interval from last PDSCH repetion reception to HARQ-feedback transmission;
* TUCI is HARQ-feedback transmission duration;
* Tprocessing is UE processing delay after PUSCH or HARQ-feedback, which is defined as one or a few milliseconds;
* delta PDCCH is not a fixed value but an offset to align with the next NPDCCH occasion, which applies to NB-IoT only.

In [1], [2], [3], [7], [9] and [10], it is proposed to extend the HARQ RTT Timer length by an offset. Regarding the offset value, there are some different views. In [1], company thinks that the offset value should be max(UE-eNB RTT - Tprocessing, 0), since the UE processing delay after PUSCH or HARQ-feedback transmission and the waiting time of UE-eNB RTT can happen in parallel. In [2], [3], [7], [9] and [10], it is suggested to use UE-eNB RTT as the offset. On the other hand, in [5], it is proposed to delay the start of HARQ RTT timer and/or UL HARQ RTT timer with an offset (e.g. UE-eNB RTT) for IoT NTN.

Based on above, rapporteur would like to ask the following question:

**Question 8: For (UL) HARQ RTT Timer enhancement, which is your preferred option?**

* **Option 1: Add an offset, equal to max(UE-eNB RTT - Tprocessing, 0), to the formula calculating the (UL) HARQ RTT timer**
* **Option 2: Add an offset, equal to UE-eNB RTT, to the formula calculating the (UL) HARQ RTT timer.**
* **Option 3: delay the start of (UL) HARQ RTT timer with an offset of UE-eNB RTT.**

|  |  |  |
| --- | --- | --- |
| **Company** | **Option** | **Additional comments** |
| OPPO | Option 1 | Since the UE processing delay after PUSCH or HARQ-feedback transmission and the waiting time of UE-eNB RTT happen in parallel, we think it is reasonable to take the maximum value of the two. |
| Xiaomi | Option 2 | Processing time can be ignored since it is quite small. |
| CATT | Option 2 | We are confusion with option 1, maybe it is max (UE-eNB RTT -Tprocessing, Tprocessing)? In TS36.321, Tprocessing is constant in the formula, meanwhile, it cannot larger than UE-eNB RTT. As the formula to calculate RTT timer in the background is used for TDD where the timing between the PDSCH and the transmission of the associated HARQ feedback is not fixed. Therefore, the simpler solution (option 2) should be used in NTN.  |
| Huawei, HiSilicon | Option 2 |  |
| MediaTek | Option 2 |  |
| Qualcomm | Option 2 | It is simple. |
| Lenovo | Option 2 |  |
| Nokia | Option 2 | Follow NR NTN solution is the simple way forward. |

**Rapporteur summary:**

TBA…

### 2.1.2 SR-Prohibit timer

SI agreements for *sr-ProhibitTimer* enhancement in IoT NTN have been captured in TR36.763, as below.

|  |
| --- |
| The *sr-ProhibitTimer* will be modified for including larger values to support IoT NTN. Alignment to NR NTN can be considered. |

In Rel-17 NR NTN WI, RAN2 has agreed to extend the timer length of sr-ProhibitTimer, and the details are FFS.

In [1], [2], [3], [5], [7], [9] and [10], it is proposed to extend *sr-ProhibitTimer* to support IoT NTN, and the following options are raised for the details.

* Option 1: increase the sr-ProhibitTimer length by UE-eNB RTT, where the unit of this UE-eNB RTT should be aligned with the configured sr-ProhibitTimer [1][3]
* Option 2: sr-ProhibitTimer value range for eMTC over NTN is extended with INTEGER (8...4096) and INTEGER (8...128) for eMTC and NB-IoT, respectively. [5]
* Option 3: Postpone treatment of sr-ProhibitTimer values until the NR NTN details have been decided. [7][9][10]

All companies seem to consider that the extension of sr-ProhibitTimer is needed. Based on this, rapporteur would like to ask the following question:

**Question 9: If sr-ProhibitTimer is extended, which is your preferred option?**

* **Option 1: increase the sr-ProhibitTimer length by UE-eNB RTT, where the unit of this UE-eNB RTT should be aligned with the configured sr-ProhibitTimer**
* **Option 2: sr-ProhibitTimer value range for eMTC over NTN is extended with INTEGER (8...4096) and INTEGER (8...128) for eMTC and NB-IoT, respectively.**
* **Option 3: Postpone treatment of sr-ProhibitTimer values until the NR NTN details have been decided.**

|  |  |  |
| --- | --- | --- |
| **Company** | **Option** | **Additional comments** |
| OPPO | Option 1/3 | We think sr-ProhibitTimer should be increased by UE-eNB RTT. If majority companies prefer option 3, we are ok to postpone the discussion. |
| Xiaomi | Option 3 |  |
| CATT | Option 3 |  |
| Huawei, HiSilicon | Option 1 |  |
| MediaTek | Option 1 |  |
| Qualcomm | Option 1 | This timer just needs extension. |
| Lenovo | Option 1 |  |
| Nokia | Option 3. |  |

**Rapporteur summary:**

TBA…

### 2.1.4 PUR

In order to improve latency and reduce UE power consumption, PUR has been introduced in NB-IoT and eMTC.

In [1], [6] and [9], it is proposed that the offset for the start of *pur-ResponseWindowTimer* should be adjusted based on UE-eNB RTT for IoT NTN.

**Question 10: Do companies agree that the start of pur-ResponseWindowTimer should be delayed by the UE-eNB RTT for IoT NTN?**

|  |  |  |
| --- | --- | --- |
| **Company** | **Agree / Disagree** | **Additional comments** |
| OPPO | Agree | This is similar as ra-ResponseWindow. |
| Xiaomi | yes |  |
| CATT | Agree |  |
| Huawei, HiSilicon | Agree with comments | In our understanding, enhancements to PUR were considered as not essential and excluded from the WID.If enhancements are supported, then we agree with the proposal |
| MediaTek | Agree |  |
| Qualcomm | Agree | But it is to note that PUR can be supported only in GEO and RAN1 work may be needed. |
| Lenovo | Agree |  |
| Nokia | Agree |  |

**Rapporteur summary:**

TBA…

Regarding pur-ResponseWindowSize, it is proposed in [9] that it is not extended.

**Question 11: Do companies agree that pur-ResponseWindowSize is not extended for IoT NTN?**

|  |  |  |
| --- | --- | --- |
| **Company** | **Agree / Disagree** | **Additional comments** |
| OPPO | Agree with comments | If no extension of repetition is required, there is no need to extend the pur-ResponseWindowSize for IoT NTN |
| Xiaomi | yes |  |
| CATT | Agree |  |
| Huawei, HiSilicon | Agree with comments | In our understanding, enhancements to PUR were considered as not essential and excluded from the WID. |
| MediaTe | Agree |  |
| Qualcomm | Agree |  |
| Lenovo | Agree |  |
| Nokia | Agree with comments | Same view as Huawei. Enhancements to PUR is not in the scope of WID. Furthermore, if the start of pur-ResponseWindowSize can be accurately compensated by UE-eNB RTT, there is no need to extend the mac- pur-ResponseWindowSize for IoT NTN |

**Rapporteur summary:**

TBA…

### 2.1.5 SPS

UL SPS can be supported in both eMTC and NB-IoT. In NB-IoT, UL SPS is only supported for BSR.

In [3], it is proposed that SPS for eMTC and SPS-BSR for NB-IoT are supported without modification for IoT NTN.

**Question 12: Do companies agree that SPS is supported without modification for IoT NTN?**

|  |  |  |
| --- | --- | --- |
| **Company** | **Agree / Disagree** | **Additional comments** |
| OPPO | Agree |  |
| Xiaomi | yes |  |
| CATT | Agree |  |
| Huawei, HiSilicon | Agree |  |
| MediaTek | Agree |  |
| Qualcomm | Agree |  |
| Lenovo | Agree |  |
| Nokia | Agree with comment | Since short and sporadic transmissions is assumed for Rel-17 IoT NTN, we are not sure if SPS could be configured. |

**Rapporteur summary:**

TBA…

## RLC

Both AM and UM modes use the *t-Reordering* timer to control the RLC waiting interval for out-of-order MAC data before considering the missing data as lost and handing any received data off to the PDCP layer. The *t-Reordering* timer can be configured with fixed values between 0 and 1600ms. Large propagation delay might have impacts on *t-Reordering* timer. In the SI phase, it is agreed that the value range of the RLC t-Reordering timer will be extended to support IoT NTN.

In [2], [3], [5], [7], [9] and [10] it is proposed to extend the value range of t-Reordering. For the exact value, in [5], it is further suggested that the RLC t-Reordering timer value is extended with ENUMERATED (ms3200, ms6400) for IoT NTN, and in [3] and [9], it is proposed to FFS on the new values. On the other hand, it is stated in [8] that as the target data rates in NB-IoT are much lower than NR, and data transmission consists of a pretty small number of packets over a relatively long period of time, there is no need to extend RLC t-Reordering timer.

**Question 13: Do companies confirm the SI agreement that the value range of the RLC t-Reordering timer will be extended to support IoT NTN?**

|  |  |  |
| --- | --- | --- |
| **Company** | **Yes / No** | **Additional comments** |
| OPPO | Yes | In order to accommodate large propagation delay in NTNT, RLC t-Reordering timer need to be extended. the exact value can be FFS. |
| Xiaomi | yes | For NR NTN, It was agreed that: “The value range of t-Reassembly shall be extended. The following set of values are possibly added for t-Reassembly timer: {ms210, ms220, ms340, ms350, ms550, ms1100, ms1650, ms2200}”Similar to NR RLC t-Reassembly, LTE RLC t-Reordering should also be extended. |
| CATT | Agree |  |
| Huawei, HiSilicon | Yes |  |
| MediaTek | Preferably No |  |
| Qualcomm | Yes |  |
| Lenovo | Yes |  |
| Nokia | Yes |  |

**Rapporteur summary:**

TBA…

## PDCP

The transmitting PDCP entity shall discard the PDCP SDU when the *discardTimer* expires for a PDCP SDU or when a status report confirms the successful delivery. The *discardTimer* can be configured up to 1500ms for eMTC and up to 81920ms for NB-IoT, or can be switched off by choosing infinity. The *discardTimer* mainly reflects the QoS requirements of the packets belonging to a service.

The following note is given in TR36.763

|  |
| --- |
| NOTE: PDCP *discardTimer* enhancements can be considered during the Work Item phase provided the impact to the Technical Specifications is minimal. |

In [5], it is stated that the PDCP *discardTimer* for eMTC should be enlarged by 2 times, e.g. the extended PDCP *discardTimer* for eMTC can be ENUMERATED (ms3000, ms6000), and considering that the PDCP *discardTimer* for NB-IoT is already large enough, it is not necessary to be extended any more.

**Question 14: Do companies agree that the PDCP discardTimer should be extended to support IoT over NTN?**

|  |  |  |
| --- | --- | --- |
| **Company** | **Agree / Disagree** | **Additional comments** |
| OPPO | Disagree with comments | Since *discardTimer* mainly reflects the QoS requirements of the packets belonging to a service, we see no need to extend this timer unless new QoS requirement are defined in SA2 for IoT NTN. |
| Xiaomi | yes | If t-Reordering is agreed to be extended to 2200ms similar to t-Reassembly, considering that RAN2 has agreed that the values of PDCP discardTimer should be greater than the RLC t-Reassembly timer, the current maximum 1500ms PDCP discardTimer value needs to be extended. Although PDCP discardTimer extension is not essential, but since the change is small, RAN2 has decided it can be considered. Besides, the new defined 5QI for NTN is applicable to LTE-M connected to 5GS case. |
| CATT | Agree |  |
| Huawei, HiSilicon | Disagree with comments | In our understanding, enhancements to PDCP discardTimer were considered as not essential and excluded from the WID. |
| MediaTek | See Comment | It should be possible to configure PDCP Discard timer larger than RLC t-Reordering timer. |
| Qualcomm | Agree for eMTC | For NB-IoT, we agree this timer does not need to be extended.  |
| Lenovo | Postpone | Depends on whether SA2 define new QoS requirement. |
| Nokia | Disagree | Since there is no new QoS requirement for IoT NTN service, it is not necessary to extend the PDCP discardTimer. |

**Rapporteur summary:**

TBA…

In [8], it is mentioned that in IoT NTN the data rates are much lower, and data transmission will consist of a pretty small number of packets over a relatively long period of time. Hence, there is no need to extend PDCP t-Reordering timer and the existing range of the PDCP t-Reordering timer is sufficient.

**Question 15: Do companies agree that there is no need to extend PDCP t-Reordering for IoT NTN?**

|  |  |  |
| --- | --- | --- |
| **Company** | **Agree / Disagree** | **Additional comments** |
| OPPO | Disagree | In LTE, PDCP t-Reordering can only be used for LWA bearers and PDCP duplication. If these two features are not supported by eMTC and NB-IoT, there would be no impact on PDCP t-Reordering for IoT NTN. |
| Xiaomi | See comment | PDCP t-Reordering is only applicable to LWA, which is not valid for IOT NTN. |
| CATT | Agree |  |
| Huawei. HiSilicon |  | PDCP t-Reordering does not apply to IOT. At least it does not apply to NB-IoT. |
| MediaTek | Agree |  |
| Qualcomm | Disagree | Not applicable |
| Lenovo | Disagree | Does not apply. |
| Nokia | Agree |  |

**Rapporteur summary:**

TBA…

# 3. Summary and Proposals

This section summarizes the discussion and reports the following proposals:

*Proposals for easy agreements:*

*Proposals for further online discussion:*

# 4. References

1. R2-2107082 Discussion on UP impact for IoT over NTN OPPO discussion Rel-17 LTE\_NBIOT\_eMTC\_NTN
2. R2-2107320 User Plane Impact for IOT NTN CATT discussion Rel-17 LTE\_NBIOT\_eMTC\_NTN
3. R2-2107425 User plane for IOT NTN Huawei, HiSilicon discussion Rel-17 LTE\_NBIOT\_eMTC\_NTN
4. R2-2107614 Provision of ephemeris Apple discussion Rel-17 LTE\_NBIOT\_eMTC\_NTN
5. R2-2107766 User plane aspects of IoT NTN ZTE Corporation, Sanechips discussion Rel-17 FS\_LTE\_NBIOT\_eMTC\_NTN
6. R2-2107915 Further enhancement for PUR in IoT NTN Lenovo, Motorola Mobility discussion Rel-17
7. R2-2108117 Discussion on User Plane impact for IoT NTN Nokia, Nokia Shanghai Bell discussion Rel-17 LTE\_NBIOT\_eMTC\_NTN
8. R2-2108335 On User-Plane Timers in NB-IoT based NTN MediaTek Inc. discussion
9. R2-2108454 User plane aspects of NB-IoT and LTE-M in NTNs Ericsson discussion Rel-17 LTE\_NBIOT\_eMTC\_NTN
10. R2-2108529 User plane for IoT-NTN CMCC discussion Rel-17 LTE\_NBIOT\_eMTC\_NTN

# Contact information

|  |  |
| --- | --- |
| Company | Delegate contact |
| COMPANY\_NAME | NAME (email@address.com) |
| OPPO | Haitao Li(lihaitao@oppo.com) |
| Huawei, HiSilicon | odile.rollinger@huawei.com |
| MediaTek | Abhishek.Roy@mediatek.com |
| Qualcomm | bshrestha@qti.qualcomm.com |
| Lenovo | Min Xu (xumin13@lenovo.com) |
| Nokia | Ping.1.Yuan@Nokia-sbell.com |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |