**3GPP TSG-RAN WG1 #106-e R1-21xxxxx**

**e-Meeting, 16th – 27th August, 2021**

**Agenda item: 8.8.1.1**

**Source: Moderator (Sharp)**

**Title: FL Summary #1 on Enhancements on PUSCH repetition type A**

**Document for: Discussion and Decision**

# Introduction

For PUSCH enahancements the following objectives are described in the Coverage Enhancement WID.

* *Specification of PUSCH enhancements [RAN1, RAN4]*
  + *Specify the following mechanisms for enhancements on PUSCH repetition type A [RAN1]*
    - *Increasing the maximum number of repetitions up to a number to be determined during the course of the work.*
    - *The number of repetitions counted on the basis of available UL slots.*

This document is intended to facilitate view exchange and discussions on the enhancements on PUSCH repetition type A, for the following assigned email discussion.

[106-e-NR-R17-CovEnh-01] Email discussion regarding enhancements for PUSCH repetition type A – Toshi (Sharp)

* 1st check point: August 19
* 2nd check point: August 24
* Final check: August 27

# Open Issues summary

## Increasing the maximum number of repetitions

For increasing of the maximum number of repetitions, the following agreements have been made.

|  |
| --- |
| **In RAN1#104-e**  Agreements:  The maximum number of repetitions for DG-PUSCH is also applicable to CG-PUSCH.  Agreements:  Rel-17 PUSCH repetition Type A supports the increase of maximum number of repetitions with repetition factors configured in a TDRA list with a row index indicated either by the configured grant configuration or by TDRA field in a DCI.   * FFS: increasing the maximum number of repetitions with repetition factor configured in *PUSCH-Config* and/or *ConfiguredGrantConfig*.   **In RAN1#105-e**  Agreement:   * Down-selection in RAN1#106-e: * Alt 1: The maximum number of repetitions supported by Rel-17 PUSCH repetition Type A is 32, irrespective of counting method, * Alt 2: The maximum number of repetitions supported by Rel-17 PUSCH repetition Type A is: 32 for the counting based on physical slots; and 16 (i.e. no change from Rel-16) for the counting based on available slots.   Agreement:  In addition to {1, 2, 3, 4, 7, 8, 12, 16} and {32}, the following additional value set for repetition factor is supported in Rel-17.   * {20, 24, 28} |

At the same time, the following two remaining issues have been identified.

* Issue#1-1: Value of the maximum number of repetitions
* Issue#1-2: RRC parameters to be extended for supporting the increased maximum number
* Issue#1-3: DCI formats supporting the repetition factors indicated/configured via TDRA lists

### [Open] Issue#1-1: Value of the maximum number of repetitions

In Rel-15/16, RRC parameter *pusch-AggregationFactor* configures the number of repetitions for PUSCH, where the candidate value set of *pusch-AggregationFactor* = {2, 4, 8}. TDRA based dynamic repetition number indication introduced in Rel-16 is applied when configured, where the candidate value set of *numberOfRepetitions-r16* = {1, 2, 3, 4, 7, 8, 12, 16}. For CG-PUSCH, RRC parameter *repK* configures the number of repetitions, where the candidate value set of *repK* = {2, 4, 8}. For Type 2 configured PUSCH repetition, TDRA based dynamic repetition number indication with *numberOfRepetitions-r16* using activation DCI is also applicable.

In RAN1#105-e, although the majority supported the maximum number of 32, some companies wanted to first see companies’ views on assumptions for designing of the maximum value, e.g. whether the number of repetitions is counted based on contiguous slots or available slots, whether to consider both FDD and TDD or either of them and whether to consider both VoIP and eMBB or either of them.

When discussing how much the maximum repetition factor should be increased, the following three cases were raised by companies.

* Case 1: FDD or SUL
* Case 2: TDD with contiguous-slot-based counting
* Case 3: TDD with available-slot-based counting

Most of the companies believed that, once the increased maximum repetition factor is decided, it should be applicable to all the three cases, but there were still different views on which cases should assumed when evaluating if proposed values achieve sufficient PUSCH coverage. Some company said the value should be decided based on Case 1 while other companies argued it should be Case 2 or Case 3. This divergence came from different views on the “bundle” of two enhancements, (a) increasing the maximum number of repetitions and (b) the number of repetitions counted on the basis of available slots. The majority thought that the two enhancements are not bundled (i.e. can be configured separately/independently) while a few companies said that the two enhancements are always bundled. The most of the majority companies were also thinking that the maximum value should be extended to 32 by the enhancement (a) so that sufficient coverage can be achieved without the enhancement (b). Furthermore, the some of the companies who preferred “always-bundle” were also saying that the maximum value should be extended to 32 even with the enhancement (a).

After several rounds of email discussions, the following agreement was made in the online session.

|  |
| --- |
| Agreement:   * Down-selection in RAN1#106-e: * Alt 1: The maximum number of repetitions supported by Rel-17 PUSCH repetition Type A is 32, irrespective of counting method, * Alt 2: The maximum number of repetitions supported by Rel-17 PUSCH repetition Type A is: 32 for the counting based on physical slots; and 16 (i.e. no change from Rel-16) for the counting based on available slots. |

The companies’ views collected during the 2nd round discussion in RAN1#105-e are summarized as follows.

* Alt 1: The maximum number of repetitions supported by Rel-17 PUSCH repetition Type A is 32, irrespective of counting method,
  + - Qualcomm, CATT, vivo, Xiaomi, ZTE, Sharp, OPPO, China Telecom (1st choice), Samsung, Apple, LG, Nokia/NSB, Sharp
* Alt 2: The maximum number of repetitions supported by Rel-17 PUSCH repetition Type A is: 32 for the counting based on physical slots; and 16 (i.e. no change from Rel-16) for the counting based on available slots.
  + - Intel, Panasonic, China Telecom (2nd choice), Lenovo/Motorola Mobility, CMCC, NTT DOCOMO, Huawei, HiSilicon, Ericsson

Companies’ views according to the contributions for RAN1#106-e are summarized as follows.

* Alt 1: The maximum number of repetitions supported by Rel-17 PUSCH repetition Type A is 32, irrespective of counting method,
  + - (16 companies): vivo [2], Nokia/Nokia Shanghai Bell [3], ZTE [4], CATT [6], China Telecom [9], OPPO [12], Qualcomm [13], CMCC [14], LG Electronics [15], Sierra Wireless [18], InterDigital [19], Apple [20], Sharp [21], NTT DOCOMO [22], Xiaomi [23]
* Alt 2: The maximum number of repetitions supported by Rel-17 PUSCH repetition Type A is: 32 for the counting based on physical slots; and 16 (i.e. no change from Rel-16) for the counting based on available slots.
  + - (8 companies): Huawei/HiSilicon [1], Rakuten Mobile, [8] NEC [10], Lenovo/Motorola Mobility [11], Ericsson [16], Intel [17]

The arguments are almost the same as what has been raised in the prerivous meetings. The proponents of Alt 1 are saying that Alt 1 provides more flexibility and achieves better coverage. Also, they think Alt 1 is simpler in terms of configuration and the restriction by Alt 2 would cause even more specification impacts. The proponents of Alt 2 are saying that 32 repetitions with the counting based on available slots makes the overall duration for a set of repetitions too long, which leads to too much delay.

1st round (Issue#1-1)

Companies are encouraged to provide their views on whether to support 32 repetitions with the counting based on available slots.

|  |  |
| --- | --- |
| **Company** | **Comments** |
| vivo | It is not necessary to limit the maximum 32 repetitions only to repetition counted based on physical slots. Even if these two features are enabled at the same time, NW could configure a suitable number of repetitions, not always the maximum 32.  Furthermore, due to PUSCH dropping rules, the actual number of repetitions could be less than the configured one, even if it is counted based on available slots. Thus, Alt 1 is preferable. |
| Apple | We support the maximum repetition 32 is applied to both counting methods. There is no repetition based on available slot in Rel.16, thus no baseline of 16 repetitions there. It’s up to gNB whether to configure 32 repetitions for available slot based counting. |
| Ericsson | Support alt2.  Alt1 indicates that the 2 features will be bundled together, while the 2 features should be treated independently in our view and this is not discussed yet.  Another reason is that 16 repeated available slots in DDDSU case already mean 80 repetitions counted based on physical slots which is already long enough.  Furthermore, repetition compared to retransmission does not give more coverage.  It would be good to first discuss issue on whether the 2 features will be bundled together or will be treated independently instead. |
| Nokia/NSB | Even if two counting approaches are adopted by RAN1 for Rel-17, Alt. 1 is still a better option. Indeed, one argument from the proponents for having two counting approaches is that counting on consecutive slots is friendlier for implementation, which indirectly states that counting on available slots requires a higher implementation complexity. Generally speaking, one should only opt for a higher complexity approach if it could offer a higher benefit, which is the number of actual repetitions in our case. In this regard, Alt. 2 cannot guarantee that counting on available slots can always offer a higher number of actual repetitions compared to the counterpart. |
| Intel | We support Alt. 2.  We do not think we need to couple these two features together. In our view, 32 repetitions based on available slots in TDD indicate that for some TDD frame structure, PUSCH repetition is quite long, which is not desirable due to unnecessary delay. |

### [Open] Issue#1-2: RRC parameters to be extended for supporting the increased maximum number

In Rel-16, there are three RRC parameters which are used to configure repetition factors, *pusch-AggregationFactor*, *numberOfRepetitions*, and *repK* as shown below, where *numberOfRepetitions* supports up to 16 repetitions while the value ranges of *pusch-AggregationFactor* in *PUSCH-Config* and *repK* in *ConfiguredGrantConfig* are {2, 4, 8} and {1, 2, 4, 8}, respectively.

|  |
| --- |
| **TS38.214v16.6.0**  6.1.2.1 Resource allocation in time domain  *[Omitted]*  For PUSCH repetition Type A, when transmitting PUSCH scheduled by DCI format 0\_1 or 0\_2 in PDCCH with CRC scrambled with C-RNTI, MCS-C-RNTI, or CS-RNTI with NDI=1, the number of repetitions *K* is determined as  - if *numberOfRepetitions* is present in the resource allocation table, the number of repetitions K is equal to *numberOfRepetitions*;  - elseif the UE is configured with *pusch-AggregationFactor*, the number of repetitions *K* is equal to *pusch-AggregationFactor*;  - otherwise *K=1*.  *[Omitted]*  6.1.2.3 Resource allocation for uplink transmission with configured grant  *[Omitted]*  For PUSCH transmissions with a Type 1 or Type 2 configured grant, the number of (nominal) repetitions *K* to be applied to the transmitted transport block is provided by the indexed row in the time domain resource allocation table if *numberOfRepetitions* is present in the table; otherwise *K* is provided by the higher layer configured parameters *repK.* |

In RAN1#104-e, we discussed which parameter(s) should be extended to support the increased maximum repetition factor. Although the large majority originally supported extension of all the three parameters, several companies expressed that extension of *numberOfRepetitions* (i.e. the one associated with TDRA list) is sufficient. In RAN1#104-e, it was agreed that Rel-17 PUSCH repetition Type A supports the increase of maximum number of repetitions with repetition factors configured in a TDRA list with a row index indicated either by the configured grant configuration or by TDRA field in a DCI. There was still the sub-bullet saying that “FFS: increasing the maximum number of repetitions with repetition factor configured in *PUSCH-Config* and/or *ConfiguredGrantConfig*”. In other words, it should be decided whether the repetition factors semi-statically configured without using the TDRA list (of which the value range was {1, 2, 4 or 8} in Rel-15/16) also support the increase of maximum number of repetitions in Rel-17 or not.

In RAN1#105-e, whether repetition factor configured in *PUSCH-Config* and/or *ConfiguredGrantConfig* supports the increased maximum number of repetitions or not was discussed again in the 1st and 2nd round discussions. Companies’ views provided are summarised as follows.

* Support (8 companies): Intel, China Telecom, Samsung, LG, OPPO, Xiaomi (IEs are up to RAN2), Nokia/NSB
* Not support (13 companies): Qualcomm, ZTE, Apple, Ericsson, NTT DOCOMO, Sharp, CATT, CMCC, Lenovo/Motorola Mobility, Sierra Wireless, vivo, Xiaomi

As shown above, in RAN1#105-e there was no consensus to support the increased maximum number of repetitions by the repetition factor configured in *PUSCH-Config* and/or *ConfiguredGrantConfig*. Proponents can try to convince the other party in this meeting.

Companies’ views according to the contributions for RAN1#106-e are summarized as follows.

* The repetition factor configured in *PUSCH-Config* and/or *ConfiguredGrantConfig* supports the increased maximum number of repetitions
  + (7 companies): Nokia/Nokia Shanghai Bell [3], Samsung [5], OPPO [12], LG Electronics [15], Intel [17], Xiaomi [23]
* The repetition factor configured in *PUSCH-Config* and/or *ConfiguredGrantConfig* does NOT support the increased maximum number of repetitions
  + (6 companies): vivo [2], ZTE [4], CATT [6], CMCC [14], Sharp [21], NTT DOCOMO [22]

The arguments are almost the same as what has been raised in the prerivous meetings. The proponents are saying that, Rel-16 TDRA list based repetition is an optional feature and Rel-17 CovEnh should support the increased maximum repetition factor without using the TDRA list based scheme. Meanwhile, the opponents think the enhancement of the TDRA list based scheme is sufficient to support the increased maximum number of repetitions and there is no need to enhance the repetition factor configured in *PUSCH-Config* and/or *ConfiguredGrantConfig* on top of the TDRA list based scheme.

1st round (Issue#1-2)

Companies are encouraged to provide their views on the necessity/benefits to enhance the repetition factor configured in *PUSCH-Config* and/or *ConfiguredGrantConfig*.

|  |  |
| --- | --- |
| **Company** | **Comments** |
| vivo | We do not see the clear benefit to enhance the repetition factor configured in *PUSCH-Config* and/or *ConfiguredGrantConfig*. Number of repetition can be applied to both DG and CG-PUSCH.  Although type-A repetition with repetition number in TDRA is an optional feature, but CE UEs must support some additional capabilities for achieve better coverage. Hence, it is not an issue if maximum 32 repetitions is only supported for parameter ‘*numberOfRepetitions*’. |
| Apple | It’s not necessary to configure the repetition factor in *PUSCH-Config* and/or *ConfiguredGrantConfig*, indication of repetition in TDRA list is enough. If it is allowed the PUSCH-config to support the 32 repetitions, additional effort is required to determine which repetition number is applied if both are configured, i.e. *pusch-AggregationFactor* vs., *numberOfRepetitions*. |
| Ericsson | Enhancement of Type A PUSCH repetition based on the repetition factors signalled in TDRA list is enough. |
| Nokia/NSB | Please refer to our comment on Issue 1-3, which is an example for why it may be beneficial for at least increasing the maximum number of repetitions for *pusch-AggregationFactor.* |
| Intel | We support to enhance the repetition factor configured in *PUSCH-Config* and/or *ConfiguredGrantConfig.*  As defined in the NR, when numberofrepetitions is not presented in the TDRA table and if pusch-AggregationFactor is configured in PUSCH-Config, the number of repetitions is determined based on pusch-AggregationFactor. In this case, it is more desirable to also support pusch-AggregationFactor with the increased maximum number of repetitions and also repK. |

### [Open] Issue#1-3: DCI formats supporting the repetition factors indicated/configured via TDRA lists

In RAN1#106-e, ZTE [4] provided their proposal that DCI format 0\_0 should support the repetition number with increased maximum repetition number configured in TDRA lists. Note that, in Rel-15/16 specification, *numberOfRepetitions* can not be indicated by DCI format 0\_0 since *numberOfRepetitions* is only included in the TDRA table configured for DCI format 0\_1 and 0\_2.

1st round (Issue#1-3)

Companies are encouraged to provide their views on whether the repetition number with increased maximum repetition number configured in TDRA lists indicated by DCI format 0\_0 is also supported in Rel-17.

|  |  |
| --- | --- |
| **Company** | **Comments** |
| vivo | Support enhanced repetition for non-fallback DCI is enough. It is not necessary to further extend to fallback DCI. |
| Apple | It’s not necessary for DCI format 0\_0 to support the increased maximum repetition. DCI format 0\_0 is the fallback DCI, which may not support the advance features. |
| Ericsson | Similar to repetition enhancement in NR Rel-16, TDRA tables/lists configured for DCI format 0\_1 and 0\_2 are enough in Rel-17.  DCI format 0\_0 is used also by uplink transmission before RRC connection, e.g. the retransmission of Msg3 which cannot be repeated in legacy and we do not think we need to apply the Rel-17 enhanced type A PUSCH repetition to these channels which are out of the scope of this agenda in our understanding. |
| Nokia/NSB | We share similar view with other companies that further extending *numberofrepetitions* for DCI format 0\_0 is not needed, and it could be considered as out-of-scope of this AI. On the other hand, this issue shows the need of increasing the maximum number of repetitions for other parameter(s), e.g. *pusch-AggregationFactor*, in case *numberofrepetitions* is not (or cannot be) configured in the TDRA table, e.g., for DCI format 0\_0. |
| Intel | We do not support increased maximum repetition number configured in TDRA lists indicated by DCI format 0\_0. DCI format 0\_0 is mainly for fallback DCI, which is not intended to be used for coverage enhacnement. We should reuse the existing mechanism as defined in Rel-15/16. |

## The number of repetitions counted on the basis of available UL slots

For the number of repetitions counted on the basis of available UL slots, the following agreements have been made.

|  |  |
| --- | --- |
| **In RAN1#104-e**  Agreements:  Select one of the following alternatives, considering the aspect whether or not the determination of all the available slots should be done prior to the first actual transmission of the repetitions (other alternatives are not precluded)  -        Alt1: Whether or not a slot is determined as available for UL transmissions depends on RRC configurations (at least tdd\_ul\_dl configuration, FFS: other RRC configurations) and does not depend on dynamic signaling (at least SFI, FFS: other dynamic signaling e.g. CI, PUSCH priority for URLLC).  -        Alt2: Whether or not a slot is determined as available for UL transmissions depends on RRC configurations (at least tdd\_ul\_dl configuration, FFS: other RRC configurations) and also depends on dynamic signaling (at least SFI, FFS: other dynamic signaling e.g. CI, PUSCH priority for URLLC).  Agreements:  For defining available slots: a slot is determined as unavailable if at least one of the symbols indicated by TDRA for a PUSCH in the slot overlaps with the symbol not intended for UL transmissions   * FFS details   **Conclusion:**  Discuss further to select one of the following alternatives:   * Alt-a: The determination of all the available slots has to be done prior to the first actual transmission of the repetitions. * Alt-b: The determination of all the available slots does not have to be done prior to the first actual transmission of the repetitions. The timeline requirement is per repetition basis.   **In RAN1#105-e**  Agreement:   * RV cycling is based on available slot for the Type A PUSCH repetition enhancement with repetitions counted based on available slot in Rel-17   **Conclusion:**   * The following agreement in RAN1#104-e is applied to all slots including special slots.  |  | | --- | | Agreements:  For defining available slots: a slot is determined as unavailable if at least one of the symbols indicated by TDRA for a PUSCH in the slot overlaps with the symbol not intended for UL transmissions.   * FFS details |   Agreement:   * Each available slot identified by the UE is considered as a transmission occasion for PUSCH repetition.   + RV is cycled across transmission occasions, irrespective of whether PUSCH transmission in the transmission occasion is further omitted or not.   Agreement:   * If PUSCH symbol in a slot overlaps with flexible symbol(s) with SSB transmission, the slot is determined as not available during the counting of repetitions. As there is no PUSCH in the slot, no PUSCH omission applies to the slot.   Agreement:  Select one from the following (further refinement of the alternatives can be further discussed), for the procedure of Rel-17 PUSCH repetition Type A (other alternatives are not precluded)   * Alt 1-B consisting of two steps   + Step 1: Determine available slots for K repetitions based on RRC configuration(s) in addition to TDRA in the DCI scheduling the PUSCH, CG configuration or activation DCI   + Step 2: The UE determines whether to drop a PUSCH repetition or not according to Rel-15/16 PUSCH dropping rules, but the PUSCH repetition is still counted in the K repetitions. * Alt 1-B’ consisting of two steps   + Step 1: Determine K repetitions based on available slots, where the available slot is the UL slot and flexible slot indicated by *tdd-UL-DL-ConfigurationCommon*, or *tdd-UL-DL-ConfigurationDedicated*.   + Step 2: The UE determines whether to drop a PUSCH repetition or not according to Rel-15/16 PUSCH dropping rules, but the PUSCH repetition is still counted in the K repetitions.   + FFS: handling of dynamic signaling (e.g. UL CI, DCI for high priority channel), e.g., UE without CI capability * Alt 2-A consisting of a single step   + Step 1: Determine available slots for K repetitions based on RRC configuration(s) and dynamic signaling (e.g. SFI, UL CI, DCI for high priority channel) in addition to TDRA in the DCI scheduling the PUSCH, CG configuration or activation DCI * Alt 2-B consisting of two steps   + Step 1: Determine available slots for K repetitions based on RRC configuration(s) and dynamic SFI in addition to TDRA in the DCI scheduling the PUSCH, CG configuration or activation DCI     - FFS timeline for the dynamic signalling   + Step 2: The UE determines whether to drop a PUSCH repetition or not according to Rel-15/16 PUSCH dropping rules, but the PUSCH repetition is still counted in the K repetitions. |

At the same time, the following eleven remaining issues have been identified.

* Issue#2-1: Use of dynamic signaling for the determination of available slots
* Issue#2-2: How to consider semi-static flexible symbols for the determination of available slots
* Issue#2-3: Use of Type0-PDCCH CSS set configuration for the determination of available slots
* Issue#2-4: Use of Invalid UL symbol configuration for the determination of available slots
* Issue#2-5: Use of semi-static PUCCH repetition configuration for the determination of available slots
* Issue#2-6: Use of SMTC configuration for the determination of available slots
* Issue#2-7: Use of other RRC configurations for the determination of available slots
* Issue#2-8: Limitation of overall duration of PUSCH repetitions
* Issue#2-9: Inter-Slot Frequency Hopping Cycle
* Issue#2-10: Handling of a collision between PUSCH repetition and P-SRS
* Issue#2-11: Applicability of available slot based counting method to paired spectrum
* Issue#2-12: Configurations/indications enabling CovEnh functions

### [Open] Issue#2-1: Use of dynamic signaling for the determination of available slots

In Rel-16, transmission occasions for a PUSCH with repetition type A are derived based on K consecutive slots, and then transmissions at some occasions may be omitted according to TDD configuration, dynamic SFI, PUSCH priority, and Cancelation Indication. Rel-15/16 also support PUCCH with N-time repetition in which only slots having sufficient UL/flexible symbols for the allocated PUCCH resource are counted as part of N slots, where UL/flexible symbols are determined by only semi-static configurations (i.e. TDD configuration and SSB configuration).

In RAN1#104-e, there were two different directions proposed for the determination of ”available slots for PUSCH repetition”, one was to follow Rel-16 PUSCH dropping rule, and the other was to follow Rel-15/16 PUCCH repetition rule. Both of the rules refer to TDD configuration and SSB configuration. Therefore, it is straightforward that TDD configuration and SSB configuration are also used for the dermination of available slots in Rel-17. On the other hand, companies had different views on whether or not dynamic signal (dynamic SFI, PUSCH priority, and cancelation indication) is used for the determination of available slots. This aspect was described in the following agreement in RAN1#104-e. For Alt 1, we discussed which semi-static configurations should be considerd for the available slot determination. Many companies preferred to reuse Rel-15/16 PUCCH repetition rules, i.e. using TDD configuration and SSB configuration, while a few companies wanted to use more configuration, e.g. invalid UL symbol configuration or Type0-CSS / CORESET#0 configuration. For Alt 2, we discussed which dynamic signaling should be considerd for the available slot determination. Although not many companies provided views on it, all the companies proving their views preferred to reuse Rel-16 PUSCH dropping rule, i.e. to use all of SFI, PUSCH priority and Cancelation Indication.

|  |
| --- |
| Agreements:  Select one of the following alternatives, considering the aspect whether or not the determination of all the available slots should be done prior to the first actual transmission of the repetitions (other alternatives are not precluded)  -        Alt1: Whether or not a slot is determined as available for UL transmissions depends on RRC configurations (at least tdd\_ul\_dl configuration, FFS: other RRC configurations) and does not depend on dynamic signaling (at least SFI, FFS: other dynamic signaling e.g. CI, PUSCH priority for URLLC).  -        Alt2: Whether or not a slot is determined as available for UL transmissions depends on RRC configurations (at least tdd\_ul\_dl configuration, FFS: other RRC configurations) and also depends on dynamic signaling (at least SFI, FFS: other dynamic signaling e.g. CI, PUSCH priority for URLLC). |

In RAN1#105-e, further clarificaions on Alt 1 and Alt 2 were made in terms of relation between the available slots and actual transmissions of the repetitions. The most of Alt 1 proponents were assuming that, even if a given slot is determined as available, PUSCH transmission in the available slot is possibly dropped depending on the dynamic signaling. In other words, Alt 1 consists of two steps, the determination of available slots and the determination of PUSCH dropping. As for Alt 2, some proponents were assuming that Alt 2 does not require PUSCH dropping procedure and as such it consists of a single step. On the other hand, some other proponents were thinking that only dynamic SFI among dynamic signaling is considered for the determination of available slots and the other dynamic signaling such as cancellation indication and priority channel would be used for PUSCH dropping. As a results of the clarification discussions, Alt 1 and Alt 2 were further classified into four sub-alternatives and the following was agreed.

|  |
| --- |
| Agreement:  Select one from the following (further refinement of the alternatives can be further discussed), for the procedure of Rel-17 PUSCH repetition Type A (other alternatives are not precluded)   * Alt 1-B consisting of two steps   + Step 1: Determine available slots for K repetitions based on RRC configuration(s) in addition to TDRA in the DCI scheduling the PUSCH, CG configuration or activation DCI   + Step 2: The UE determines whether to drop a PUSCH repetition or not according to Rel-15/16 PUSCH dropping rules, but the PUSCH repetition is still counted in the K repetitions. * Alt 1-B’ consisting of two steps   + Step 1: Determine K repetitions based on available slots, where the available slot is the UL slot and flexible slot indicated by *tdd-UL-DL-ConfigurationCommon*, or *tdd-UL-DL-ConfigurationDedicated*.   + Step 2: The UE determines whether to drop a PUSCH repetition or not according to Rel-15/16 PUSCH dropping rules, but the PUSCH repetition is still counted in the K repetitions.   + FFS: handling of dynamic signaling (e.g. UL CI, DCI for high priority channel), e.g., UE without CI capability * Alt 2-A consisting of a single step   + Step 1: Determine available slots for K repetitions based on RRC configuration(s) and dynamic signaling (e.g. SFI, UL CI, DCI for high priority channel) in addition to TDRA in the DCI scheduling the PUSCH, CG configuration or activation DCI * Alt 2-B consisting of two steps   + Step 1: Determine available slots for K repetitions based on RRC configuration(s) and dynamic SFI in addition to TDRA in the DCI scheduling the PUSCH, CG configuration or activation DCI     - FFS timeline for the dynamic signalling   + Step 2: The UE determines whether to drop a PUSCH repetition or not according to Rel-15/16 PUSCH dropping rules, but the PUSCH repetition is still counted in the K repetitions. |

According to the 4th round discussion in RAN1#105-e, companies showed the following preferences.

* Alt 1-B consisting of two steps
  + Step 1: Determine available slots for K repetitions based on RRC configuration(s) in addition to TDRA in the DCI scheduling the PUSCH, CG configuration or activation DCI
  + Step 2: The UE determines whether to drop a PUSCH repetition or not according to Rel-15/16 PUSCH dropping rules, but the PUSCH repetition is still counted in the K repetitions.
  + Support (16 companies): CATT, vivo, Intel, Lenovo/Motorola Mobility (2nd preference), Qualcomm, Sharp, OPPO, LG, CMCC, WILUS, Ericsson, Nokia/NSB, ZTE, Xiaomi
* Alt 1-B’ consisting of two steps
  + Step 1: Determine K repetitions based on available slots, where the available slot is the UL slot and flexible slot indicated by *tdd-UL-DL-ConfigurationCommon*, or *tdd-UL-DL-ConfigurationDedicated*.
  + Step 2: The UE determines whether to drop a PUSCH repetition or not according to Rel-15/16 PUSCH dropping rules, but the PUSCH repetition is still counted in the K repetitions.
  + FFS: handling of dynamic signaling (e.g. UL CI, DCI for high priority channel), e.g., UE without CI capability
  + Support (1 company): Apple
* Alt 2-A consisting of a single step
  + Step 1: Determine available slots for K repetitions based on RRC configuration(s) and dynamic signaling (e.g. SFI, UL CI, DCI for high priority channel) in addition to TDRA in the DCI scheduling the PUSCH, CG configuration or activation DCI
  + Support (1 company): Samsung
* Alt 2-B consisting of two steps
  + Step 1: Determine available slots for K repetitions based on RRC configuration(s) and dynamic SFI in addition to TDRA in the DCI scheduling the PUSCH, CG configuration or activation DCI
    - FFS timeline for the dynamic signalling
  + Step 2: The UE determines whether to drop a PUSCH repetition or not according to Rel-15/16 PUSCH dropping rules, but the PUSCH repetition is still counted in the K repetitions.
  + Support (2 companies): Lenovo/Motorola Mobility (1st preference), ZTE

In the discussions in RAN1#105-e, one issue on Alt 2-A and Alt 2-B was raised for CG-PUSCH, which is an RV mismatch problem between the UE and the gNB. More specifically, it is assumed that 4 slots with FUUU provided by semi-static TDD configuration, and those 4 slots are allocated for CG-PUSCH resource. The gNB sends the dynamic SFI indicating UUUU, but the UE fails to detect it. In this case, the gNB assumes that the first available slot is the first slot, while the UE determines that the first available slot is the second slot. It is better to consider this issue when discussing the down-selection from the above alternatives.

Companies’ views according to the contributions for RAN1#106-e are summarized as follows.

* Alt 1-B consisting of two steps
  + Step 1: Determine available slots for K repetitions based on RRC configuration(s) in addition to TDRA in the DCI scheduling the PUSCH, CG configuration or activation DCI
  + Step 2: The UE determines whether to drop a PUSCH repetition or not according to Rel-15/16 PUSCH dropping rules, but the PUSCH repetition is still counted in the K repetitions.
  + Support (20 companies): vivo [2], Nokia/Nokia Shanghai Bell [3], ZTE [4], CATT [6], Rakuten Mobile [8], China Telecom [9], NEC [10], OPPO [12], Qualcomm [13], CMCC [14], LG Electronics [15], Ericsson [16], Intel [17], Sierra Wireless [18], InterDigital [19], Sharp [21], NTT DOCOMO [22], Xiaomi [23], WILUS [24]
* Alt 1-B’ consisting of two steps
  + Step 1: Determine K repetitions based on available slots, where the available slot is the UL slot and flexible slot indicated by *tdd-UL-DL-ConfigurationCommon*, or *tdd-UL-DL-ConfigurationDedicated*.
  + Step 2: The UE determines whether to drop a PUSCH repetition or not according to Rel-15/16 PUSCH dropping rules, but the PUSCH repetition is still counted in the K repetitions.
  + FFS: handling of dynamic signaling (e.g. UL CI, DCI for high priority channel), e.g., UE without CI capability
  + Support (5 companies): CMCC [14], LG Electronics [15], Ericsson [16], Apple [20], WILUS [24]
* Alt 2-B’ consisting of two steps
  + Step 1: Determine available slots for K repetitions based on RRC configuration(s) and dynamic SFI received earlier enough (e.g. N2 symbols before the first repetition occasion of PUSCH repetition) in addition to TDRA in the DCI scheduling the PUSCH, CG configuration or activation DCI
    - FFS timeline for the dynamic signalling
  + Step 2: The UE determines whether to drop a PUSCH repetition or not according to Rel-15/16 PUSCH dropping rules (including dynamic SFI received later), but the PUSCH repetition is still counted in the K repetitions.
  + Support (4 companies): Huawei/HiSilicon [1], Lenovo/Motorola Mobility [11]

Looking at the above proposals, all the alternatives meet the condition that all the available slots are determined prior to the first transmission of the PUSCH repetitions.

1st round (Issue#2-1)

Companies (especially Alt 2-A and Alt 2-B proponents) are encouraged to provide their views on the different understanding issue (e.g. RV mismatch issue) between the UE and the gNB due to DCI detection failure at the UE side.

Companies (especially Alt 1-B’ proponents) are encouraged to provide further clarifications on handling of dynamic signaling.

|  |  |
| --- | --- |
| **Company** | **Comments** |
| vivo | We support Alt 1-B. |
| Apple | We support Alt 1-B’. The UL cancellation and physical layer priority are optional feature defined in Rel.16. For Rel.17 coverage enhancement, it’s subject to UE capability whether to support these features as well.  For Alt 1-B, could proponent clarify whether flexible slot is considered as available slot with this option? |
| Ericsson | 1-B. |
| Nokia/NSB | We support Alt. 1-B for the simplicity and for the reasons mentioned in our Tdocs. |
| Intel | We support Alt 1-B as it is very clear on the procedure to determine the available slots and apply the dropping/cancellation for PUSCH repetition. |

### [Open] Issue#2-2: How to consider semi-static flexible symbols for the determination of available slots for CG-PUSCH

In RAN1#104-e, it was agreed to use at lease tdd\_ul\_dl configuration (i.e. *tdd-UL-DL-ConfigurationCommon* and *tdd-UL-DL-ConfigurationDedicated*) for the available slot determination. In RAN1#105-e, there were some discussions on how the semi-static flexible symbols should be considered for the available slot determination.

In RAN1#106-e, Panasonic [7] is proposing that, for CG-PUSCH when dynamic SFI moniroting is configured, semi-static flexible symbol should be considered as unavailable for PUSCH repetition as in Rel.15/16. Meanwhile, during the discussions in RAN1#105-e, several companies expressed their views that semi-static flexible symbol should be always considered as available for CG-PUSCH irrespective of the dynamic SFI moniroting configuration.

1st round (Issue#2-2)

Companies are encouraged to provide their views on whether semi-static flexible symbol should be considered as available or unavailable for PUSCH repetition for CG-PUSCH when dynamic SFI moniroting is configured.

|  |  |
| --- | --- |
| **Company** | **Comments** |
| vivo | To avoid the available slots impacted by dynamic signaling, semi-static flexible symbol should be considered as unavailable for CG-PUSCH repetition, when dynamic SFI monitoring is configured. |
| Apple | The flexible slot is considered as available slot, the Rel. 15/16 defined rules is applied to determine whether the flexible slot is dropped. |
| Ericsson | Dynamic signaling should not be considered to determine the available slot as we discussed in issue #2-1.  Whether CG PUSCH will be cancelled due to dynamic SFI/CI depends on the rules in legacy. |
| Nokia/NSB | This issue is closely related to Issue 2-1. The main concern which has been bringing up so far by several companies is the discrepancy in counting available slots if dynamic singaling is considered. This also applies for dynamic SFI. On the other hand, the benefit of counting the F slots is only in latency reduction, which may not be a critical issue in this feature. |
| Intel | We share similar view as other companies. flexible symbol can be considered as available for PUSCH repetition in the step 1 and dynamic SFI in the step 2 can be used to drop/cancel the CG-PUSCH. |

### [Open] Issue#2-3: Use of Type0-PDCCH CSS set configuration for the determination of available slots

Regardless of whether dynamic signaling is used for the determination of available slots or not, at least of some RRC configurations need to be used for the determination. According to the following agreement in RAN1#104-e, it was agreed to use at lease tdd\_ul\_dl configuration (i.e. *tdd-UL-DL-ConfigurationCommon* and *tdd-UL-DL-ConfigurationDedicated*) for the available slot determination for both Alt 1 and Alt 2.

|  |
| --- |
| Agreements:  Select one of the following alternatives, considering the aspect whether or not the determination of all the available slots should be done prior to the first actual transmission of the repetitions (other alternatives are not precluded)  -        Alt1: Whether or not a slot is determined as available for UL transmissions depends on RRC configurations (at least tdd\_ul\_dl configuration, FFS: other RRC configurations) and does not depend on dynamic signaling (at least SFI, FFS: other dynamic signaling e.g. CI, PUSCH priority for URLLC).  -        Alt2: Whether or not a slot is determined as available for UL transmissions depends on RRC configurations (at least tdd\_ul\_dl configuration, FFS: other RRC configurations) and also depends on dynamic signaling (at least SFI, FFS: other dynamic signaling e.g. CI, PUSCH priority for URLLC). |

In addition, the following agreement was made in RAN1#105-e, which means that SSB configuration (i.e. *ssb-PositionsInBurst*) is also referred to for the determination of available slots.

|  |
| --- |
| Agreement:   * If PUSCH symbol in a slot overlaps with flexible symbol(s) with SSB transmission, the slot is determined as not available during the counting of repetitions. As there is no PUSCH in the slot, no PUSCH omission applies to the slot. |

In RAN1#105-e, it was also discussed whether other RRC configurations should be used or not. The companies’ views on what other RRC configurations are used are summarized as follows. In this meeting, it is suggested discussing the above RRC configurations separately, as the background of each proposal seems different.

* No other RRC configurations
  + Supported by: CATT, Qualcomm, Apple, OPPO, LG, Ericsson
* CORESET0 with Type0-PDCCH CSS set
  + Supported by: Intel, Lenovo/Motorola Mobility, Sharp (study further), WILLUS, Xiaomi
* Invalid UL symbols for DL-to-UL switching purpose
  + Supported by: Intel, Xiaomi
* Semi-static PUCCH with repetitions
  + Supported by: WILUS
* SSB based measurement by SMTC
  + Supported by: vivo
* DL-to-UL switching for half duplex FDD redcap UE
  + Supported by: vivo (wait the conclusion in RedCap WI)
* All the RRC configurations that inpact on the PUSCH repetitions
  + Supported by: ZTE
* Revisit in RAN1#106-e
  + Nokia/NSB

Under Issue#2-4, whether the configuration of CORESET0 with Type0-PDCCH CSS set is used for the available slot determination or not is discussed.

In Rel-16, CORESET0 with Type0-PDCCH CSS set (for paired spectrum) is used to determine time domain resource allocation for PUSCH repetition Type B. On the other hand, any clear behaviour (including PUSCH dropping rule) has not been specified for the case PUSCH repetition Type A overlaps with CORESET0 with Type0-PDCCH CSS set.

Companies’ views according to the contributions for RAN1#106-e are summarized as follows.

* Should use CORESET0 with Type0-PDCCH CSS set for the available slot determination
  + Samsung [5], Intel [17], Xiaomi [23]
* No need to use CORESET0 with Type0-PDCCH CSS set for the available slot determination
  + ZTE [4], CATT [6], Panasonic [7], Qualcomm [13], CMCC [14], LG Electronics [15], Sharp [21]

1st round (Issue#2-3)

In the operation with a fixed TDD configuration, CORESET0 with Type0-PDCCH CSS is most likely to be mapped on fixed DL symbols. DG-PUSCH cannot be scheduled on the DL symbols, and CG-PUSCH is not transmitted on DL symbols even if configured. Therefore, for the fixed TDD configuration case, PUSCH repetitions do not collide with CORESET0 with Type0-PDCCH CSS.

In the operation with a dynamic TDD configuration, CORESET0 with Type0-PDCCH CSS may be mapped on semi-static flexible symbols. CG-PUSCH on the semi-static flexible symbols is not transmitted unless the UE detects the dynamic SFI which indicates the symbols as uplink. Therefore, as long as the gNB sends appropriate SFI, any collistion between CG-PUSCH and CORESET0 with Type0-PDCCH CSS does not happen. On the other hand, for DG-PUSCH repetitions, Rel-16 specifications is not specifing any collisiton handling. This means, the Rel-16 gNB has to schedule DG-PUSCH repetitions such that the DG-PUSCH repetitions never overlap with CORESET0 with Type0-PDCCH CSS on semi-static flexible symbols. For PUSCH repetitions counted on the basis of available slots, K available slots for a TB are likely to be non-contiguous and varies depending on the UL timing. Whether or not Rel-17 gNB can handle it in the same way as Rel-16, even with the available slot based counting, would be a discussion point.

Companies are encouraged to provide their views on whether or not it is possible for Rel-17 gNB to always schedule DG-PUSCH repetitions with the available slot based counting such that the DG-PUSCH repetitions never overlap with CORESET0 with Type0-PDCCH CSS on semi-static flexible symbols in the operation with a dynamic TDD configuration.

|  |  |
| --- | --- |
| **Company** | **Comments** |
| vivo | Not necessary, it is up to NW to avoid collision with type-0 CSS. |
| Apple | The similar issue was there since Rel.15, it’s up to gNB scheduling to avoid the collision. Same handling could be used for available slot-based counting. |
| Ericsson | No other configurations are needed for available slot determination. Omission rules on overlapping between CORESET0 with Type0-PDCCH CSS set and PUSCH are clear in current spec. and can be reused. |
| Nokia/NSB | The scheduler can avoid collision. In the worst case, Rel-15/16 rules for collision handling can be applied. This may result in some loss in the actual number of repetitions, which can be avoided by configuring a high number of repetitions in the first place. |
| Intel | We support CORESET0 with Type0-PDCCH CSS set to determine available slot in the first step.  In our view, it is similar to collision handling and invalid symbols for PUSCH repetition type B transmission where symbol(s) indicated by pdcch-ConfigSIB1 in MIB for a CORESET for Type0-PDCCH CSS set. Otherwise, it is unnecessary restriction at gNB scheduler to avoid the collision between CORESET0 and PUSCH repetition. |

### [Open] Issue#2-4: Use of Invalid UL symbol configuration for the determination of available slots

Under Issue#2-4, whether the configuration of invalid UL symbols for DL-to-UL switching gaps is used for the available slot determination or not is discussed.

Similar to CORESET0 with Type0-PDCCH CSS set, in Rel-16, invalid UL symbol configuration is used to determine time domain resource allocation for PUSCH repetition Type B. On the other hand, any clear behaviour (including PUSCH dropping rule) has not been specified for the case PUSCH repetition Type A overlaps with the invalid UL symbols.

Companies’ views according to the contributions for RAN1#106-e are summarized as follows.

* Should use the invalid UL symbols for DL-to-UL switching gaps for the available slot determination
  + Samsung [5], Panasonic [7], Intel [17], Xiaomi [23], WILUS [24]
* No need to use the invalid UL symbols for DL-to-UL switching gaps for the available slot determination
  + ZTE[4] , CATT [6], Qualcomm [13], CMCC [14], LG Electronics [15], Sharp [21]

1st round (Issue#2-4)

In the operation with a fixed TDD configuration, DL-to-UL gaps exist only in special slots, and are configured as flexible symbols. For CG-PUSCH, if the PUSCH allocation in a special slot is configured such as to overlap with DL symbols, the special slot is determined as not available. This can be done without introducing any additional handling of DL-to UL gaps. For DG-PUSCH, the scheduler has to avoid overlapping between DC-PUSCH and DL symbols in a special slot anyway. Further avoidance of DL-to-UL gap in the same special slot does not cause any additional complexity.

In the operation with a dynamic TDD configuration, DL-to-UL gaps may be mapped on semi-static flexible symbols and likely to be further indicated as dynamic flexible symbols. CG-PUSCH on the semi-static flexible symbols is not transmitted unless the UE detects the dynamic SFI which indicates the symbols as uplink. Therefore, as long as the gNB sends appropriate SFI, any collistion between CG-PUSCH and DL-to-UL gaps does not happen. On the other hand, for DG-PUSCH repetitions, Rel-16 specifications is not specifing any collisiton handling. Similar to CORESET0 with Type-PDCCH CSS, this means that the Rel-16 gNB has to schedule DG-PUSCH repetitions such that the DG-PUSCH repetitions never overlap with DL-to-UL gaps on semi-static flexible symbols. Whether or not Rel-17 gNB can handle it in the same way as Rel-16, even with the available slot based counting, would be a discussion point.

Companies are encouraged to provide their views on whether or not it is possible for Rel-17 gNB to always schedule DG-PUSCH repetitions with the available slot based counting such that the DG-PUSCH repetitions never overlap with DL-to-UL gaps in the operation with a dynamic TDD configuration.

|  |  |
| --- | --- |
| **Company** | **Comments** |
| vivo | Not necessary. It can be up to NW scheduler to avoid PUSCH repetition overlapping with the DL  -to-UL switching gap. |
| Apple | gNB could handle this by implementation. No need to introduce new rules for available slot determination. |
| Ericsson | No other configurations are needed for available slot determination. Omission rules on collision between invalid symbols and PUSCH are clear in current spec. and can be reused. |
| Nokia/NSB | Same answer as for Issue 2-3. |
| Intel | We support this DL-to-UL gaps to determine the available slots.  As mentioned above, in our view, it is similar to collision handling and invalid symbols for PUSCH repetition type B transmission. Otherwise, it is unnecessary restriction at gNB scheduler to avoid the collision between switching gap and PUSCH repetition. |

### [Open] Issue#2-5: Use of semi-static PUCCH repetition configuration for the determination of available slots

Under Issue#2-5, whether the configuration of semi-static PUCCH with repetitions is used for the available slot determination or not is discussed.

In Rel-15/16, overlapping of PUSCH repetition Type A and semi-static PUCCH with repetitions is handled by PUSCH dropping rules, i.e. all PUSCHs overlapping with PUCCH repetitions are dropped. Whether PUSCH resources collide with semi-static PUCCH repetitions or not is known by the UE prior to the start of PUSCH repetitions.

|  |
| --- |
| **TS38.213 v16.6.0**  9.2.6 PUCCH repetition procedure  *[Omitted]*  If a UE would transmit a PUCCH over a first number of slots and the UE would transmit a PUSCH with repetition Type A over a second number of slots, and the PUCCH transmission would overlap with the PUSCH transmission in one or more slots, and the conditions in clause 9.2.5 for multiplexing the UCI in the PUSCH are satisfied in the overlapping slots, the UE transmits the PUCCH and does not transmit the PUSCH in the overlapping slots. |

Companies’ views according to the contributions for RAN1#106-e are summarized as follows.

* Should use semi-static PUCCH repetition configuration for the available slot determination
  + ZTE [4]
* No need to use semi-static PUCCH repetition configuration for the available slot determination
  + CATT [6], Panasonic [7], Qualcomm [13], CMCC [14], LG Electronics [15], Intel [17], Sharp [21]

1st round (Issue#2-5)

Companies are encouraged to provide their views on whether the overlapping of PUSCH repetition Type A and semi-static PUCCH with repetitions is handled by PUSCH dropping rules in the same as Rel-15/16 or is handled by the available slot determination.

|  |  |
| --- | --- |
| **Company** | **Comments** |
| Apple | Rel.15/16 defined dropping rule is applied for colliding between PUCCH PUSCH repetition. |
| Ericsson | No other configurations are needed for available slot determination. Omission rules on collision between semi-static PUCCH repetition and PUSCH are clear in current spec. and can be reused. |
| Nokia/NSB | Same answer as for Issue 2-3. |
| Intel | We do not support this. |

### [Open] Issue#2-6: Use of SMTC configuration for the determination of available slots

Under Issue#2-6, whether a collision of PUSCH repetition with the SSB based measurement by SMTC configuration is take into consideration for the available slot determination or not is discussed.

In RAN1#104-e, RAN1#105-e and RAN1#106-e, vivo [2] proposed that SSB based measurement by SMTC configuration should be handling in the available slot determination step, based on the following descriptions in RAN4 specification.

|  |
| --- |
| **TS38.133**  9.2.5.3 Scheduling availability of UE during intra-frequency measurements  UE shall be capable of measuring without measurement gaps when the SSB is completely contained in the active bandwidth part of the UE. When any of the conditions in the following clauses is met, there are restrictions on the scheduling availability; otherwise, there is no scheduling restriction. Note that the SSB symbols indicated by the union set of *SSB-ToMeasure* from all the configured measurement objects on the same serving carrier which can be merged[2], if it is configured; otherwise, all *L* SSB symbols within the SMTC window duration defined in clause 4.1 of TS 38.213 [3] are included.  9.2.5.3.1 Scheduling availability of UE performing measurements in TDD bands on FR1  When the UE performs intra-frequency measurements in a TDD band, the following restrictions apply due to SS-RSRP or SS-SINR measurement  - The UE is not expected to transmit PUCCH/PUSCH/SRS on SSB symbols to be measured, and on 1 data symbol before each consecutive SSB symbols to be measured and 1 data symbol after each consecutive SSB symbols to be measured within SMTC window duration. If the high layer in TS 38.331 [2] signalling of *smtc2*is configured, the SMTC periodicityfollows *smtc2*; Otherwise SMTC periodicity follows *smtc1.*  When the UE performs intra-frequency measurements in a TDD band, the following restrictions apply due to SS-RSRQ measurement  - The UE is not expected to transmit PUCCH/PUSCH/SRS on SSB symbols to be measured, RSSI measurement symbols, and on 1 data symbol before each consecutive SSB to be measured/RSSI symbols and 1 data symbol after each consecutive SSB to be measured/RSSI symbols within SMTC window duration. If the high layer signalling of *smtc2*is configured in TS 38.331 [2], the SMTC periodicityfollows *smtc2*; Otherwise the SMTC periodicity follows *smtc1.* |

On the other hand, Sharp [21] expresses their views that, in Rel-15/16, when SSB based measurement by SMTC configuration would overlap with UL symbols, the network configures an appropriate measurement gap. Once the measurement gap is configured to the UE, the MAC layer does not transmit on UL-SCH during the gap, and therefore PHY layer does not need to handle any collision between PUSCH repetition and SSB measurement with SMTC.

Companies’ views according to the contributions for RAN1#106-e are summarized as follows.

* Should use SMTC configuration for the available slot determination
  + vivo [2], ZTE [4]
  + FFS: Panasonic [7]
* No need to use SMTC configuration for the available slot determination
  + CATT [6], Qualcomm [13], CMCC [14], LG Electronics [15], Intel [17], Sharp [21], WILUS [24]

1st round (Issue#2-6)

Companies are encouraged to provide their views on whether the overlapping of PUSCH repetition Type A and SMTC-based SSB measurement needs to be handled by the available slot determination procedure or not.

|  |  |
| --- | --- |
| **Company** | **Comments** |
| vivo | The SMTC configuration is also semi-statically provided by RRC signalling, both NW and UE are aligned on the SMTC configuration and the corresponding UE behaviour. We support to handle the overlapping of PUSCH repetition Type A and SMTC-based SSB measurement by the available slot determination procedure.  Regarding sharp’s view, the PUSCH dropping during SMTC in rel-16 is caused by scheduling restriction for intra-frequency measurement. While measurement gap is configured for inter-frequency measurement, it is different story. |
| Apple | It’s not necessary to consider SMTC configuration for available slot determination. |
| Ericsson | No other configurations are needed for available slot determination. Rules on collision between SMTC configuration and PUSCH are clear in current spec. and can be reused. |
| Nokia/NSB | Same answer as for Issue 2-3. |
| Intel | We do not think we need to consider SMTC configuration for the available slot determination |

### [Open] Issue#2-7: Use of other RRC configurations for the determination of available slots

Issue#2-7 discusses other RRC configurations than the ones discussed in Issues #2-3 to #2-6.

In RAN1#105-e ZTE proposed all the RRC configurations should be taken into consideration to determine available slots.

Companies’ views according to the contributions for RAN1#106-e are summarized as follows.

* Should use semi-static PUCCH with larger priority index for the available slot determination
  + ZTE [4]
* No need to use other RRC configurations for the available slot determination
  + CATT [6], Qualcomm [13], CMCC [14], LG Electronics [15], Intel [17], Sharp [21]

1st round (Issue#2-7)

Companies are encouraged to provide their views on the use of other RRC configurations for the determination of available slots.

|  |  |
| --- | --- |
| **Company** | **Comments** |
| vivo | Semi-static PUCCH with larger priority index can be handled by reusing Rel-15/16 PUSCH dropping rules. There is no need to consider the use of semi-static PUCCH with larger priority index and other RRC configuration for the available slot determination procedure. |
| Apple | Rel.15/16 defined dropping rules are applied if PUCCH and PUSCH are colliding. |
| Ericsson | No other configurations are needed for available slot determination. Rules in current spec. are clear and can be reused. |
| Nokia/NSB | Same answer as for Issue 2-3. |
| Intel | This should be considered in the second step. |

### [Open] Issue#2-8: Limitation of overall duration of PUSCH repetitions

In RAN1#104-e and 105-e, several companies proposed having a time window/limitation of overall time duration for a signle set of PUSCH repetitions so that an excessive delay can be avoided. Meanwhile, more companies thought that the network can control the overall time duration by setting an appropreate repetition factor.

* Alt 1: Count of available slots continues until reaching the indicated/configured repetition factor.
* Alt 2: Count of available slots continues until reaching the indicated/configured repetition factor or reaching the limitation of overall duration for a set of PUSCH repetitions, whichever comes first.
  + Supported by: OPPO, Intel, Samsung, Lenovo/Motrola Mobility

At the same time, it was also mentioned that whether this issue needs to be discussed depends on the outcome of Issue#2-1, because the overall duration is certainly deterministic and controlable if the available slots are determined by semi-static configurations only while the postponement of PUSCH repetition due to dynamic scheduling may lead to difficulty to predict when the repetitions finish. Therefore, this discussion has been deferred, since we have not yet concluded the discussion on use of dynamic signaling for the determination of available slots.

Companies’ views according to the contributions for RAN1#106-e are summarized as follows.

* For CG-PUSCH with repetitions, overall duration of PUSCH repetitions should not exceed the configured periodicity of the configured PUSCH (similar to Rel-15/16).
  + Huawei/HiSilicon [1], Qualcomm [13]
  + Should be discussed: Panasonic [7]
* For DG-PUSCH with repetitions, no need to introduce upper limit of overall duration of PUSCH repetitions
  + Panasonic [7]
* A PUSCH transmission scheduled in a non-available slot is postponed to a next available slot where the PUSCH transmission is counted – the postponement is done until the count reaches the configured/indicated number of repetitions N, or until the duration of the PUSCH transmission is K slots and the count is not larger than N.
  + Samsung [5], Lenovo/Motorola Mobility [11], Intel [17], InterDigital [19]

For DG-PUSCH, 4 companies are proposing introducing the cap of over all duration for a set of PUSCH repetitions.

For CG-PUSCH, Rel-15/16 has the following restriction on the repetition configuration. It would be discussed if similar limitation for overall duration of a set of PUSCH repetitions needs to be applied to Rel-17.

|  |
| --- |
| **TS38.214**  6.1.2.3.1 Transport Block repetition for uplink transmissions of PUSCH repetition Type A with a configured grant  *[Omitted]*  The UE is not expected to be configured with the time duration for the transmission of *K* repetitions larger than the time duration derived by the periodicity *P*. If the UE determines that, for a transmission occasion, the number of symbols available for the PUSCH transmission in a slot is smaller than transmission duration *L*, the UE does not transmit the PUSCH in the transmission occasion. |

1st round (Issue#2-8)

Companies are encouraged to provide their views on the follwoing proposals.

For DG-PUSCH with counting based on the available slots,

* Alt 1: Count of available slots continues until reaching the indicated/configured repetition factor.
* Alt 2: Count of available slots continues until reaching the indicated/configured repetition factor or reaching the limitation of overall duration for a set of PUSCH repetitions, whichever comes first.

For CG-PUSCH with counting based on the available slots,

* Overall duration of PUSCH repetitions should not exceed the configured periodicity of the configured PUSCH (similar to Rel-15/16).

|  |  |
| --- | --- |
| **Company** | **Comments** |
| vivo | For DG-PUSCH, support Alt 1.  For CG-PUSCH, support the same/similar way as Rel-15/16. |
| Apple | It’s not necessary to set the cap for repetition, if the delay is the concern, the gNB could schedule the PUSCH with less repetitions. Alt 1 is ok. |
| Ericsson | For DG-PUSCH, it’s up to gNB to configure the number of repetitions, there’s no need of a limitation.  For CG PUSCH, legacy specification is enough, no specification change is expected. |
| Nokia/NSB | For DG-PUSCH, support Alt 1 with same understanding as Apple.  For CG-PUSCH, support the same/similar way as Rel-15/16. |
| Intel | For DG-PUSCH, we support Alt. 1 as gNB can flexibly indicate the repetition number  For CG-PUSCH, we are fine with the proposal. |

### [Open] Issue#2-9: Inter-Slot Frequency Hopping Cycle

In RAN1#105-e, Qualcomm raises the issue related to inter-slot frequency hopping. In Rel-15/16, inter-slot frequency hopping cycle is determined on the basis of consecutive physical slots. More specifically, hop index in a slot is determined based on whether the slot index within a radio frame is odd or even.

|  |
| --- |
| **TS38.214v16.6.0**  6.3.1 Frequency hopping for PUSCH repetition Type A  *[Omitted]*  In case of inter-slot frequency hopping, the starting RB during slot  is given by:  ,  where  is the current slot number within a radio frame, where a multi-slot PUSCH transmission can take place,  is the starting RB within the UL BWP, as calculated from the resource block assignment information of resource allocation type 1 (described in Clause 6.1.2.2.2) and is the frequency offset in RBs between the two frequency hops. |

However, the hopping based on physical slot indices causes an uneven distribution of hops in TDD system. InterDigital also mentions the same issue. In order to resolve this issue, Qualcomm’s proposal was that, for inter-slot frequency hopping, hop index is determined based on indexing within the determined available slots.

Companies’ views expressed in RAN1#105-e are summarized as follows. Although the majority was thinking Rel-16 inter-slot frequency hopping works with the counting based on available slots, several companies were still thinking some more discussions were necessary.

* For PUSCH repetition Type A without joint channel estimation, inter-slot frequency hopping is based on physical slot index as in Rel-15/16.
  + ZTE, Apple, Lenovo/Motorola Mobility, Sharp, CATT, LG, Nokia/NSB, Xiaomi, Huawei, HiSilicon (12 companies)
* No need to make any agreement on inter-slot frequency hopping cycle
  + Samsung, CMCC, Panasonic, Intel (4 companies)
* Good to discuss inter-slot frequency hopping cycle issue with AI8.8.1.3
  + Ericsson, OPPO (2 companies)
* Modifications on inter-slot frequency hopping cycle should be considered
  + Qualcomm (1 company)

Companies’ views according to the contributions for RAN1#106-e are summarized as follows.

* For PUSCH repetition type A without joint channel estimation, inter-slot frequency hopping is based on physical slot index as in Rel-15/16.
  + ZTE [4], Ericsson [16]
* To support joint channel estimation, the frequency hopping pattern optimization can be discussed in the JCE topic.
  + Ericsson [16]
* For PUSCH repetition type A without joint channel estimation, both inter-slot frequency hopping based on physical slot index as in Rel-15/16 and hopping pattern for joint channel estimation can be supported.
  + Sharp [21]

For this meeting, there is no company proposing inter-slot frequency hopping cycle based on available slots.

1st round (Issue#2-9)

Companies are encouraged to provide their views on the follwoing proposal.

Proposed conclusion:

* For Rel-17 PUSCH repetition Type A without joint channel estimation, no new inter-slot frequency hopping cycle is introduced.

|  |  |
| --- | --- |
| **Company** | **Comments** |
| vivo | Support. |
| Apple | Ok with this conclusion. |
| Ericsson | Looks fine. |
| Nokia/NSB | Support. |
| Intel | We are fine with this proposed conclusion. |

### [Open] Issue#2-10: Handling of a collision between PUSCH repetition and P-SRS

In RAN1#105-e, Huawei proposed studying the case when PUSCH repetition Type A overlaps with SRS. In Rel-15/16, the specifications are not specifing any special handling of collisions between PUSCH and SRS, except for the case of overlaping between high priorirty PUSCH and low priority SRS. The reason is because the gNB can schedule PUSCH and SRS such that any collision between them does not happen. Huawei expressed their views that such a strict TDRA limitation leads to more resource waste of UL symbols when the PUSCH is repeated across more slots, and proposed the following option in order to avoid the wast of uplink resources.

* If symbols in the slot indicated by TDRA for a PUSCH repetition overlaps with the symbols still intended for other UL transmission ( but not for this PUSCH transmission), such as higher priority URLLC signal or periodic SRS or cancellation indication, non-overlapped UL symbols within the overlapped UL slot can be used for one PUSCH repetition to make a full utilization of uplink resources.

During the discussions in RAN1#105-e, the large majority expressed their views that this proposal (i.e. partial dropping of PUSCH) violates PUSCH repetition Type A nature, i.e. the same symbol allocation for all the repetitions.

This issue was discussed in the GTW session in RAN1#105-e, but no agreement/conclusion was made. Mr. chairman suggested revisiting this issue in RAN1#106-e.

Companies’ views according to the contributions for RAN1#106-e are summarized as follows.

* For collision between enhanced Type A PUSCH repetitions and other UL channels.
  + Reuse existing collision handling rules
    - Qualcomm [13]
  + Define a priority rule
    - Ericsson [16]
  + FFS
    - CMCC [14]

For this meeting, there is no company proposing the following proposal:

* If symbols in the slot indicated by TDRA for a PUSCH repetition overlaps with the symbols still intended for other UL transmission ( but not for this PUSCH transmission), such as higher priority URLLC signal or periodic SRS or cancellation indication, non-overlapped UL symbols within the overlapped UL slot can be used for one PUSCH repetition to make a full utilization of uplink resources.

1st round (Issue#2-10)

Companies are encouraged to provide their views on the follwoing proposal.

Proposed conclusion:

* Rel-17 PUSCH repetition Type A does NOT support the following partial PUSCH transmisssion:
  + If symbols in the slot indicated by TDRA for a PUSCH repetition overlaps with the symbols still intended for other UL transmission ( but not for this PUSCH transmission), such as higher priority URLLC signal or periodic SRS or cancellation indication, non-overlapped UL symbols within the overlapped UL slot can be used for one PUSCH repetition to make a full utilization of uplink resources.

|  |  |
| --- | --- |
| **Company** | **Comments** |
| vivo | Support the proposed conclusion. |
| Apple | OK with this conclusion. |
| Ericsson | In our understanding:   * If CI or a grant of other UL transmission from the UE with higher priority is received either prior to transmission of the first repetition or during the transmission of PUSCH repetitions, this indicates one of the available slots is to be preempted, the UE cancels the transmission of PUSCH repetition in this slot. * For the case that one particular slot is determined as an available slot for multiple time-overlapping UL channels or signals (type A PUSCH repetition enhancement option 2, enhanced A-SRS in Rel-17 FeMIMO topic, or enhanced SPS HARQ-ACK in Rel-17 URLLC topic), RAN1 may need to define the priority of these multiple time-overlapping UL transmissions enhanced in Rel-17. The UE only transmits the channel or signal with the highest priority in overlapping symbols in the slot.   It would be good to make this clear with minimum specification changes after we agree on how to determine available slot for Type A PUSCH. |
| Nokia/NSB | We are fine with the proposed conclusion. |
| Intel | We are fine with the proposed conclusion. We should maintain same resource allocation in time for PUSCH repetition type A |

### [Open] Issue#2-11: Applicability of available slot based counting method to paired spectrum

In the 4th round discussion in RAN1#105-e, Qualcomm suggested collecting companies’ views on the following proposal.

* For PUSCH Type A repetitions, counting based on available slots is only applicable to unpaired spectrum.

The companies’ views on the above proposal in RAN1#105-e are summarized as follows.

* + Support: CATT, Intel, Qualcomm, Apple, LG, Ericsson, Nokia/NSB, ZTE, Xiaomi
  + Defer the discussion until concluding what semi-static configurations to be used for the detemination of available slots: Sharp, Panasonic, WILUS
  + No need: CMCC

Companies’ views according to the contributions for RAN1#106-e are summarized as follows.

* For Rel-17 PUSCH repetition Type A, counting based on available slots is only applicable to unpaired spectrum.
  + Qualcomm [13], Sierra Wireless [18], Sharp [21]
* For Rel-17 PUSCH repetition Type A, counting based on available slots is applicable to unpaired and paired spectrum.
  + ZTE [4]

The discussion point would be what components are used for the determination of available slots for paired spectrum. The components which have been agreed so far to be used for the available slot determination are only *tdd-UL-DL-ConfigurationCommon*, *tdd-UL-DL-ConfigurationDedicated* and *ssb-PositionsInBurst*. All of there three components are valid only for unpaired spectrum, as there would no collision between DL and UL for paired spectrum, except for Half-duplex FDD discussed in RedCap WI.

1st round (Issue#2-11)

Companies are encouraged to provide their views on the follwoing proposal.

* For PUSCH Type A repetitions, counting based on available slots is only applicable to unpaired spectrum.

|  |  |
| --- | --- |
| **Company** | **Comments** |
| vivo | For unified design, there is no need to introduce this restrction on the applicability.  Besides, for Half duplex FDD redcap UEs, how to handle PUSCH repetition overlapping with SSB is still under discussion, so it seems too early to say the counting on available slots only apply to TDD spectrum. |
| Apple | OK with this proposal. |
| Ericsson | Seems fine, and as long as the available slot determination is clear, such proposal is obvious. |
| Nokia/NSB | We are fine with the FL’s proposal. |
| Intel | We are fine with the proposal. |

### [Open] Issue#2-12: Configurations/indications enabling CovEnh functions

As described in the WID, two enhancements for PUSCH repetition type A would be supported. In RAN1#105-e, it was discussed whether/how to activate those enhancements.

The first aspect is, when UE is capable of CovEnh enhancement and reported it to the Rel-17 gNB, whether the Rel-17 gNB still have a choice to configure the UE with legacy repetition scheme. The large majority answered ”Yes” to the question. As Rel-17 gNB may want to use the legacy repetition scheme or may not have the ability of the CovEnh ennhancement, most of the companies think that Rel-17 gNB should not be forced to use the CovEnh enhancement function, even if the UE is capable of it.

And the second aspect discussed was whether two enhancements can be applied at the same time.

The following proposals were provided by FL which covered the above two aspects, and were discussed in GTW session.

* Rel-17 supports the configurability of “the counting based on available slots” function.
* Rel-17 supports the configuration enabling “the increased maximum number of repetitions”.
* FFS: whether to support only one of the two functions at the same time or simultaneous use of the two functions by a single UE capable of both functions.

However, no consensus was made in GTW session, and it was suggested more focusing on the functions themselves rather than configurations of the functions. After the GTW, the above proposals were reformulated as below and then companies’ views on them were collected again for further discussions.

* “The counting based on available slots” is enabled via RRC signaling. If not enabled, the Rel-17 UE uses “the counting based on physical slots” (i.e. the same repetition counting as in Rel15/16).
* Rel-17 RRC parameter(s) relating to “the increased maximum number of repetitions” is provided via RRC signaling to a UE which performs PUSCH repetitions with “the increased maximum number of repetitions”. If not provided, the UE performs PUSCH repetitions subject to Rel-15/16 configuration.
* FFS:
  + Alt 1: A single UE can be configured with both “the counting based on available slots” and the Rel-17 RRC parameter(s) relating to “the increased maximum number of repetitions” at the same time.
    - Support: CATT, OPPO, ZTE, Xiaomi
  + Alt 2: A single UE can be configured with only one of “the counting based on available slots” and the Rel-17 RRC parameter(s) relating to “the increased maximum number of repetitions” but not both at a given time.
    - Support: vivo, Ericsson

Support all bullets: Intel, Sharp, CMCC, Nokia/NSB, Xiaomi

Support 1st bullet, no need 2nd bullet: Qualcomm

Revisit in RAN1#106-e: Samsung, Panasonic, LG, Nokia/NSB

Companies’ views according to the contributions for RAN1#106-e are summarized as follows.

* A single UE can be configured with both “the counting based on available slots” and the Rel-17 RRC parameter(s) relating to “the increased maximum number of repetitions” at the same time.
  + ZTE [4]
* One of three options (legacy repetition Type A and two Rel-17 enhancements) is configured to a UE
  + Ericsson [16]
* The enhancements are always tied to each other and are always enabled/disabled at the same time.
  + Nokia/Nokia Shanghai Bell [3], Panasonic [7]
* Dynamic switching between two enhancements should be supported
  + Lenovo/Motorola Mobility [11]

Looking at the proposals in companies’ contributions for this meeting, there are three alternatives in terms of configurations of two enhancements. Note that UE feature discussions for CovEnh would be done later and the discussions in this meeting should focus on configurations of enhancements and the associated behaviors.

1st round (Issue#2-12)

Companies are encouraged to provide their views on the follwoing alternatives.

* Alt 1:
  + “The counting based on available slots” is enabled via RRC signaling. If not enabled, the Rel-17 UE uses “the counting based on physical slots” (i.e. the same repetition counting as in Rel15/16).
  + Rel-17 RRC parameter(s) relating to “the increased maximum number of repetitions” is provided via RRC signaling to a UE which performs PUSCH repetitions with “the increased maximum number of repetitions”. If not provided, the UE performs PUSCH repetitions subject to Rel-15/16 configuration.
* Alt 2:
  + A single Rel-17 RRC parameter enabling both “The counting based on available slots” and “the increased maximum number of repetitions” is introduced. If not enabled, the Rel-17 UE uses “the counting based on physical slots” (i.e. the same repetition counting as in Rel15/16) and performs up to 16 repetitions subject to existing configuration.
* Alt 3:
  + A single Rel-17 RRC parameter indicating one of the following three combinations is introduced.
    - “The counting based on physical slots” and “the existing maximum number of repetitions”
    - “The counting based on physical slots” and “the increased maximum number of repetitions”
    - “The counting based on available slots” and “the existing maximum number of repetitions”

Companies are encouraged also to provide their views on whether to support dynamic switching between the counting based on available slots and the couning based on physical slots.

|  |  |
| --- | --- |
| **Company** | **Comments** |
| vivo | Support Alt 1. |
| Apple | Alt 3 is preferred. The “the existing maximum number of repetitions” can be removed from third sub-bullet, due to no repetition was agreed in Rel.16 for available slot counting. |
| Ericsson | Alt 3 is preferred though we want to include some details to make sure our understanding of the proposal is correct.  In our understanding, in Rel-17 we will have 3 types of Type A PUSCH repetitions:   * + - Repetition Type A0 (legacy):       * “The counting based on physical slots” and “the existing maximum number of repetitions” will use the RRC configured legacy TDRA list which is already in the specification     - Repetition Type A1:       * “The counting based on physical slots” and “the increased maximum number of repetitions” is the function 1 which requires RRC configured new TDRA list with increase number of repetitions for backward compatibility     - Repetition Type A2:       * “The counting based on available slots” and “the existing maximum number of repetitions” is the function 2 which will reuse legacy RRC configured TDRA list though the repetitions should be counted based on available slot.   In Rel-17, whether Type A or Type B is selected is based on RRC signaling *pusch-RepTypeIndicatorDCI-0-x-r16*, if Type A is selected, UE will determine whether Type A0/A1/A2 is used can be based on another *pusch-RepTypeIndicatorDCI-0-x-r17* signaling corresponding to related DCI format in Rel-17, see below example RRC signaling.  PUSCH-Config ::= SEQUENCE {  …  pusch-RepTypeIndicatorDCI-0-2-r16 ENUMERATED { pusch-RepTypeA, pusch-RepTypeB} OPTIONAL, -- Need R  pusch-RepTypeIndicatorDCI-0-1-r16 ENUMERATED { pusch-RepTypeA, pusch-RepTypeB} OPTIONAL, -- Need R  pusch-RepTypeIndicatorDCI-0-1-r17 ENUMERATED { pusch-RepTypeA0, pusch-RepTypeA1, pusch-RepTypeA2} OPTIONAL, -- Cond RepTypeA  pusch-RepTypeIndicatorDCI-0-2-r17 ENUMERATED { pusch-RepTypeA0, pusch-RepTypeA1, pusch-RepTypeA2} OPTIONAL, -- Cond RepTypeA  …  }  It should also be fine that we define type A0 as a default repetition type if the *pusch-RepTypeIndicatorDCI-0-x-r17* is not present, meaning that there will be only 2 values (Type A1 and A2) for this Rel-17 repetition type signaling (in this case, is it Alt 1?). |
| Nokia/NSB | Alt. 2. We support a unified solution in Rel-17 since the whole point of having two counting approaches is unclear to us. If increasing the maximum number of repetitions for counting on consecutive physical slots appears to be an acceptable solution for everyone, why don’t we simply use only that counting approach and introduce even a higher maximum number of repetitions? This would result in the same number of actual repetitions as counting on available slots with a limited maximum number of repetitions. |
| Intel | We support Alt. 1. It does not necessarily combine these two features for PUSCH coverage enhancement. |

# References

1. R1-2106495 Discussion on coverage enhancements for PUSCH repetition type A Huawei, HiSilicon
2. R1-2106611 Discussion on enhancement for PUSCH repetition type A vivo
3. R1-2106655 Enhancements on PUSCH repetition type A Nokia, Nokia Shanghai Bell
4. R1-2106739 Discussion on enhanced PUSCH repetition type A ZTE
5. R1-2106902 Enhancements on PUSCH repetition type A Samsung
6. R1-2106988 Discussion on enhancements on PUSCH repetition type A CATT
7. R1-2107116 Discussion on enhancements on PUSCH repetition Type A Panasonic Corporation
8. R1-2107121 Discussion on enhancements on PUSCH repetition type A Rakuten Mobile, Inc
9. R1-2107123 Enhancements on PUSCH repetition type A China Telecom
10. R1-2107140 Discussion on PUSCH repetition type A NEC
11. R1-2107190 Enhancements on PUSCH repetition type A Lenovo, Motorola Mobility
12. R1-2107256 Enhancements on PUSCH repetition type A OPPO
13. R1-2107359 Enhancements on PUSCH Repetition Type A Qualcomm Incorporated
14. R1-2107417 Discussion on enhancements on PUSCH repetition type A CMCC
15. R1-2107548 Discussions on PUSCH repetition type A enhancements LG Electronics
16. R1-2107559 PUSCH Repetition Type A Enhancement Ericsson
17. R1-2107602 Enhancements on PUSCH repetition type A Intel Corporation
18. R1-2107634 Design considerations for PUSCH repetition Type A Enhancements Sierra Wireless, S.A.
19. R1-2107650 Type-A PUSCH repetition for coverage enhancement InterDigital, Inc.
20. R1-2107753 Discussion on PUSCH repetition type A enhancement Apple
21. R1-2107799 Enhancements on PUSCH repetition type A Sharp
22. R1-2107872 Enhancements on PUSCH repetition type A NTT DOCOMO, INC.
23. R1-2107935 Enhancements on PUSCH repetition type A Xiaomi
24. R1-2108157 Discussion on enhancements on PUSCH repetition type A WILUS Inc.

# List of agreements

## Agreements in RAN1#104-e

Agreements:

Select one of the following alternatives, considering the aspect whether or not the determination of all the available slots should be done prior to the first actual transmission of the repetitions (other alternatives are not precluded)

-        Alt1: Whether or not a slot is determined as available for UL transmissions depends on RRC configurations (at least tdd\_ul\_dl configuration, FFS: other RRC configurations) and does not depend on dynamic signaling (at least SFI, FFS: other dynamic signaling e.g. CI, PUSCH priority for URLLC).

-        Alt2: Whether or not a slot is determined as available for UL transmissions depends on RRC configurations (at least tdd\_ul\_dl configuration, FFS: other RRC configurations) and also depends on dynamic signaling (at least SFI, FFS: other dynamic signaling e.g. CI, PUSCH priority for URLLC).

Agreements:

The maximum number of repetitions for DG-PUSCH is also applicable to CG-PUSCH.

Agreements:

For defining available slots: a slot is determined as unavailable if at least one of the symbols indicated by TDRA for a PUSCH in the slot overlaps with the symbol not intended for UL transmissions

* FFS details

Agreements:

Rel-17 PUSCH repetition Type A supports the increase of maximum number of repetitions with repetition factors configured in a TDRA list with a row index indicated either by the configured grant configuration or by TDRA field in a DCI.

* FFS: increasing the maximum number of repetitions with repetition factor configured in *PUSCH-Config* and/or *ConfiguredGrantConfig*.

**Conclusion:**

Discuss further to select one of the following alternatives:

* Alt-a: The determination of all the available slots has to be done prior to the first actual transmission of the repetitions.
* Alt-b: The determination of all the available slots does not have to be done prior to the first actual transmission of the repetitions. The timeline requirement is per repetition basis.

## Agreements in RAN1#105-e

Agreement:

* RV cycling is based on available slot for the Type A PUSCH repetition enhancement with repetitions counted based on available slot in Rel-17

Agreement:

* Down-selection in RAN1#106-e:
* Alt 1: The maximum number of repetitions supported by Rel-17 PUSCH repetition Type A is 32, irrespective of counting method,
* Alt 2: The maximum number of repetitions supported by Rel-17 PUSCH repetition Type A is: 32 for the counting based on physical slots; and 16 (i.e. no change from Rel-16) for the counting based on available slots.

**Conclusion:**

* The following agreement in RAN1#104-e is applied to all slots including special slots.

|  |
| --- |
| Agreements:  For defining available slots: a slot is determined as unavailable if at least one of the symbols indicated by TDRA for a PUSCH in the slot overlaps with the symbol not intended for UL transmissions.   * FFS details |

Agreement:

In addition to {1, 2, 3, 4, 7, 8, 12, 16} and {32}, the following additional value set for repetition factor is supported in Rel-17.

* {20, 24, 28}

Agreement:

* Each available slot identified by the UE is considered as a transmission occasion for PUSCH repetition.
  + RV is cycled across transmission occasions, irrespective of whether PUSCH transmission in the transmission occasion is further omitted or not.

Agreement:

* If PUSCH symbol in a slot overlaps with flexible symbol(s) with SSB transmission, the slot is determined as not available during the counting of repetitions. As there is no PUSCH in the slot, no PUSCH omission applies to the slot.

Agreement:

Select one from the following (further refinement of the alternatives can be further discussed), for the procedure of Rel-17 PUSCH repetition Type A (other alternatives are not precluded)

* Alt 1-B consisting of two steps
  + Step 1: Determine available slots for K repetitions based on RRC configuration(s) in addition to TDRA in the DCI scheduling the PUSCH, CG configuration or activation DCI
  + Step 2: The UE determines whether to drop a PUSCH repetition or not according to Rel-15/16 PUSCH dropping rules, but the PUSCH repetition is still counted in the K repetitions.
* Alt 1-B’ consisting of two steps
  + Step 1: Determine K repetitions based on available slots, where the available slot is the UL slot and flexible slot indicated by *tdd-UL-DL-ConfigurationCommon*, or *tdd-UL-DL-ConfigurationDedicated*.
  + Step 2: The UE determines whether to drop a PUSCH repetition or not according to Rel-15/16 PUSCH dropping rules, but the PUSCH repetition is still counted in the K repetitions.
  + FFS: handling of dynamic signaling (e.g. UL CI, DCI for high priority channel), e.g., UE without CI capability
* Alt 2-A consisting of a single step
  + Step 1: Determine available slots for K repetitions based on RRC configuration(s) and dynamic signaling (e.g. SFI, UL CI, DCI for high priority channel) in addition to TDRA in the DCI scheduling the PUSCH, CG configuration or activation DCI
* Alt 2-B consisting of two steps
  + Step 1: Determine available slots for K repetitions based on RRC configuration(s) and dynamic SFI in addition to TDRA in the DCI scheduling the PUSCH, CG configuration or activation DCI
    - FFS timeline for the dynamic signalling
  + Step 2: The UE determines whether to drop a PUSCH repetition or not according to Rel-15/16 PUSCH dropping rules, but the PUSCH repetition is still counted in the K repetitions.