**3GPP TSG-RAN WG1 Meeting #110-Bis R1-22xxxx**

**e-Meeting, October 10th - October 19th, 2022**

**Agenda Item: 7.2**

**Source: Moderator (Qualcomm)**

**Title: Summary #1 of PRS Search Window Issue**

**Document for: Discussion and decision**

# Introduction

This email Discussion is related to the following 2 papers:

[1] [R1-2209841](https://www.3gpp.org/ftp/tsg_ran/WG1_RL1/TSGR1_110b-e/Docs/R1-2209841.zip) Discussion on the PRS search window Huawei, HiSilicon

[2] [R1-2209936](https://www.3gpp.org/ftp/tsg_ran/WG1_RL1/TSGR1_110b-e/Docs/R1-2209936.zip) Discussion on an issue on DL-PRS Search Window Definitions Qualcomm Incorporated

# Discussion

This issue was first identified in a RAN2 contribution during previous RAN2 meeting ([3]). During that meeting, it was identified ([4]) that it may be more appropriate to discuss it first in RAN1.

The issue comes from the fact that (a) the timing offsets at the TRP locations (SFN offsets) are defined positively (towards the "next subframe boundary") whereas the timing offset at the UE location ((expected) RSTD) can be positive or negative (towards the "closest subframe boundary"); and (b) the window centre calculations use a mixture of timings at TRP location and UE location.

If the issue is not fixed, the calculation of the DL-PRS search window center may result in an incorrect starting subframe of the DL-PRS of a neighbour TRP, and therefore, DL-PRS acquisition may fail.

## Solution 1 [1]

In the first solution proposed in [1], the suggestion is that both sfn-Offset and integerSubframeOffset could correspond to a value that is set by the network, so we remove how the rounding is down, with the intention that the network will set the 2 values in such a way to ensure that the PRS search window is what is supposed to be.

***Proposal 1: It is RAN1 understanding that sfn-Offset and integerSubframeOffset in NR-DL-PRS-SFN0-Offset is not required to be specified as round-down or round-to-the-closest, but should be set a value by the network to ensure that the UE assumption is correct when to determine the search window centre.***

* ***The definition of sfn-Offset and integerSubframeOffset could be changed to reflect that intention.***

***Proposal 2: Include the following text proposal in the LS to RAN2.***

|  |
| --- |
| ***nr-DL-PRS-SFN0-Offset***This field specifies the time offset of the SFN#0 slot#0 for the given TRP with respect to SFN#0 slot#0 of the assistance data reference TRP and comprises the following subfields:- ***sfn-Offset*** specifies the SFN offset at the TRP antenna location between the assistance data reference TRP and this neighbour TRP. The offset corresponds to the number of full radio frames counted from the beginning of a radio frame #0 of the assistance data reference TRP to the beginning of the closest subsequentradio frame #0 of this neighbour TRP.- ***integerSubframeOffset*** specifies the frame boundary offset at the TRP antenna location between the assistance data reference TRP and this neighbour TRP counted in full subframes. The offset corresponds to the number of full subframes counted from the beginning of a subframe #0 of the assistance data reference TRP to the beginning of the closest subsequent subframe #0 of this neighbour TRP.NOTE: Network shall set the value in accordance with the defined search window for the target device using *nr-DL-PRS-ExpectedRSTD* and *nr-DL-PRS-ExpectedRSTD-Uncertainty* |

## Solution 2 [2]

In the 2nd solution, a backward compatible solution to fix this issue is proposed by adding a new field to indicate an 1-ms offset on the calculation of the search window.

**Proposal:** Send an LS to RAN2 to ask them to add a *searchWindowOffset* parameter to the DL-PRS assistance data, which can take the values 0 or 1, and which should be added to *N* when calculating the center location of the DL-PRS search window.

The server knows the PRS and SFN offsets and can determine the excess value to be added to *N* when computing the search window. Therefore, it is proposed to add a *searchWindowOffset* to the DL-PRS assistance data, which can take the values 0 or 1 and solves the problems discussed in section 3 above:

NR-DL-PRS-AssistanceDataPerTRP-r16 ::= SEQUENCE {

 dl-PRS-ID-r16 INTEGER (0..255),

 nr-PhysCellID-r16 NR-PhysCellID-r16 OPTIONAL, -- Need ON

 nr-CellGlobalID-r16 NCGI-r15 OPTIONAL, -- Need ON

 nr-ARFCN-r16 ARFCN-ValueNR-r15 OPTIONAL, -- Need ON

 nr-DL-PRS-SFN0-Offset-r16 NR-DL-PRS-SFN0-Offset-r16,

 nr-DL-PRS-ExpectedRSTD-r16 INTEGER (-3841..3841),

 nr-DL-PRS-ExpectedRSTD-Uncertainty-r16

 INTEGER (0..246),

 nr-DL-PRS-Info-r16 NR-DL-PRS-Info-r16,

 ...,

 [[

 prs-OnlyTP-r16 ENUMERATED { true } OPTIONAL -- Need ON

 ]],

 [[

 searchWindowOffset-r16 ENUMERATED { m0, m1, ... } OPTIONAL -- Need ON

 ]]

}

| *NR-DL-PRS-AssistanceData* field descriptions |
| --- |
| […] |
| ***nr-DL-PRS-SFN0-Offset***This field specifies the time offset of the SFN#0 slot#0 for the given TRP with respect to SFN#0 slot#0 of the assistance data reference TRP and comprises the following subfields:- ***sfn-Offset*** specifies the SFN offset at the TRP antenna location between the assistance data reference TRP and this neighbour TRP. The offset corresponds to the number of full radio frames counted from the beginning of a radio frame #0 of the assistance data reference TRP to the beginning of the closest subsequent radio frame #0 of this neighbour TRP.- ***integerSubframeOffset*** specifies the frame boundary offset at the TRP antenna location between the assistance data reference TRP and this neighbour TRP counted in full subframes. The offset is counted from the beginning of a subframe #0 of the assistance data reference TRP to the beginning of the closest subsequent subframe #0 of this neighbour TRP, rounded down to multiples of subframes. |
| ***nr-DL-PRS-ExpectedRSTD***This field indicates the RSTD value that the target device is expected to measure between this TRP and the assistance data reference TRP. The *nr-DL-PRS-ExpectedRSTD* field takes into account the expected propagation time difference as well as transmit time difference of PRS positioning occasions between the two TRPs. The resolution is 4×Ts, with Ts=1/(15000\*2048) seconds. |
| ***nr-DL-PRS-ExpectedRSTD-Uncertainty***This field indicates the uncertainty in *nr-DL-PRS-ExpectedRSTD* value.The uncertainty is related to the location server′s a‑priori estimate of the target device location. The *nr-DL-PRS-ExpectedRSTD* and *nr-DL-PRS-ExpectedRSTD-Uncertainty* togetherdefine the search window for the target device.The resolution R is- Ts if all PRS resources are in frequency range 2,- 4×Ts otherwise,with Ts=1/(15000\*2048) seconds.The target device may assume that the beginning of the subframe for the PRS of this TRP is received within the search window of size- [*-nr-*DL*-PRS-ExpectedRSTD-Uncertainty*×R *;* *nr-DL-PRS-ExpectedRSTD-Uncertainty*×R] centerd at TREF*+*1 millisecond×(N+M)+*nr-DL-PRS-ExpectedRSTD*×4×Ts,where TREF is the reception time of the beginning of the subframe for the PRS of the assistance data reference TRP at the target device antenna connector, N can be calculated based on- *nr-DL-PRS-SFN0-Offset*- *dl-PRS-Periodicity-and-ResourceSetSlotOffset*- *dl-PRS-ResourceSlotOffset*and M is given by *searchWindowOffset*. |
| […] |
| ***searchWindowOffset***This field specifies the excess value to be added to *N* when computing the center location of the search window as described in *nr-DL-PRS-ExpectedRSTD-Uncertainty* field description. Enumerated value '*m0'* indicates M=0, '*m1*' indicates M=1. |

The above would be compatible with existing assistance data and measurement definitions. Legacy UEs would calculate the search window as currently defined (i.e., with *M*=0), which however, may fail in certain instances as shown. New UEs would take the *M* value into account when determining the beginning of DL-PRS subframes of neighbour TRPs. But all definitions of e.g., SFN offset, (expected) RSTD, etc. could remain unchanged.

# 3 Moderator Comments

In short, in both solutions 1 & 2, a change is suggested that would enable the LMF to provide values on the assistance data fields that will result in correct computation of the search window.

* In solution 1, the suggestion is to allow the network to assign *sfn-Offset* and *integersubframeOffset* without explicitly saying how the rounding is happening. In that case, the LMF could add or subtract a value, as needed to ensure that the search window is eventually computed correctly.
* In solution 2, the definitions of *sfn-Offset* and *integersubframeOffset* do not change, and the network adds an offset 0 or 1 in a new field (i.e. instead of “hiding” it inside the *integersubframeOffset* as it was in solution 1).

Moderator’s understanding is that, with regards to DL PRS search window computation, either solution could work; in one case, the LMF is adjusting the *integersubframeOffset* value without informing the UE, in the other case, it sends an additional offset to the UE without adjusting the *integersubframeOffset.*

However, Solution 1 changes the definition of the *integerSubframeOffset*, which appears (strictly speaking) non-backwards compatible. As noted in [2] (Figure 1), the current definitions for the *sfn-Offset*, *integerSubframeOffset*, and *subframeOffset* align to the "true" transmission offset with Tc granularity (for UE-based). With solution 1, this would now have a 1-ms ambiguity, since the rounding of the *integerSubframeOffset* is changed to "unspecified" in Solution 1 (i.e., unknown at the UE), which could create ambiguity for some scenarios in UE-based as shown in the example below.

In short, in solution 1, in UE-based, the UE cannot determine the actual RTD since the *integersubframeOffset* may or may not have an offset. In solution 2, in UE-based, the UE *can* determine the actual RTD of the network, by adding the *integersubframeOffset (1 msec granularity)* to the (fine) subframeOffset (Tc granularity).

Example:

* Assume 2 TRPs with a Transmission offset = 1.4ms
* A UE that has a Geometric RSTD = 0.55ms
* Expected RSTD(computed by network) = Reception offset mod (-0.5,0.5) = Transmission offset +geometric RSTD mod(-0.5,0.5) = -0.05ms
* Transmit RTDInfo = 0.4ms (fractional part of transmission offset)

Solution1:

* integersubframeoffset = 2
* search window center = 2+-0.05 = 1.95ms
* UE needs to obtain geometric RSTD which includes the transmission offset between the TRPs; however, this is not exactly known (it can be, integersubframeoffset + RTDInfo or integersubframeoffset + RTDInfo -1). So, a UE estimates geometric RSTD as 0.55 ms or -0.45ms. UE doesn’t know which one is the correct.

Solution2:

* integersubframeoffset = 1
* search window offset = 1 (new parameter added in the AD according to this Solution)
* search window center = 1+1+-0.05 = 1.95ms (same as Solution 1, since both solutions address the ambiguity with the search window center).
* Transmission offset = integersubframeoffset + RTDInfo = 1.4ms.
* Reception offset = 1.95ms
* Geometric RSTD measured = 0.55ms

# 4 Questions

## Question 4.1 Do you agree that there can be an issue that needs to be fixed?

|  |  |  |
| --- | --- | --- |
| **Company** | **Yes/No** | **Comments** |
|  |  |  |

**Question 4.2 If yes, which solution is preferred to fix the problem and why?**

|  |  |  |
| --- | --- | --- |
| **Company** | **Solution**  | **Comments** |
|  |  |  |

# References

[1] [R1-2209841](https://www.3gpp.org/ftp/tsg_ran/WG1_RL1/TSGR1_110b-e/Docs/R1-2209841.zip) Discussion on the PRS search window Huawei, HiSilicon

[2] [R1-2209936](https://www.3gpp.org/ftp/tsg_ran/WG1_RL1/TSGR1_110b-e/Docs/R1-2209936.zip) Discussion on an issue on DL-PRS Search Window Definitions Qualcomm Incorporated

[3] R2-2208119, “Issues with DL-PRS Search Window Definitions”, Qualcomm Incorporated

[4] R2-2208962, “Summary of [AT119-e][407][POS] Rel-15/16 LPP”, Qualcomm Incorporated