3GPP TSG-RAN2 Meeting #121bis-e R2-2204394

eMeeting, 17~26 April 2023

Agenda Item: 7.5.4.1

Source: Qualcomm

Title: Summary of [AT121bis-e][212][XR] BSR solutions (Qualcomm)

Document for: Discussion and Decision

# **Introduction**

This report provides a summary of the following at-meeting email discussion:

* [AT121bis-e][212][XR] BSR solutions (Qualcomm)

Scope: Attempt to find out which among the BSR table solutions have most support and preclude those with least support (if possible). Should discuss pros and cons of each solution and determine which are acceptable to companies (and why). Can also discuss other general details (e.g. how the BSR tables are used).

Intended outcome: Discussion report in [R2-2304394](https://www.3gpp.org/ftp/TSG_RAN/WG2_RL2/TSGR2_121bis-e/Docs/R2-2304394.zip).

Deadline: Deadline 2

During the online discussion on Monday, three solutions for BSR table enhancements were discussed:

* [1] proposes that a basic set of BSR tables can be pre-defined to support common use cases. But it also allows network to RRC configure additional BSR tables on demand, e.g. based on UE’s traffic characteristics.
* [2] proposes that UE generates a new BSR table by applying a scaling factor to a pre-defined reference BSR table. The scaling factor is RRC configured by network.
* [3] proposes that UE can send up to two BSR MAC CEs in single PUSCH transmission for a pending BSR. The first BSR MAC CE indicates a coarse value of UE’s buffer size, and the second BSR MAC CE refines the value reported by the first BSR. The two BSRs may or may not use different BSR tables.

Although these three solutions share the same goal of reducing quantization errors of BSR, they do differ in various ways and have their own advantage and disadvantages. In the following, we first discuss their pros and cons, on aspects such as whether they are efficient in reducing quantization error (e.g. weighing their achievable levels of quantization error vs overhead they introduce), their impacts on network’s flexibility in scheduling and complexity of UE implementation, etc. In the second half of this discussion, we then discuss other general but related issues for new BSR tables.

# **Contact information**

|  |  |
| --- | --- |
| Company | Name (Email) |
| Qualcomm | Linhai He (linhaihe@qti.qualcomm.com) |
| Nokia | Chunli Wu (Chunli.wu@nokia-sbell.com) |
| ZTE | eswar.vutukuri@zte.com.cn |
| LGE | Hanseul Hong (hanseul.hong@lge.com) |
| NEC | Yuhua chen(Yuhua.chen@emea.nec.com) |
| CMCC | Kangyi Liu (liukangyi@chinamobile.com) |
| Ericsson | Richard Tano (richard.tano@ericsson.com) |
| Quectel | Lily.huang@quectel.com |
| Sony | Vivek.sharma@sony.com |
| Samsung | Weiping Sun(wp.son@samsung.com) |
| Apple | Ping-Heng Wallace Kuo (pingheng\_kuo@apple.com) |
| Vivo | Chenli (Chenli5g@vivo.com) |
| Intel | Rafia Malik ([rafia.malik@intel.com](mailto:rafia.malik@intel.com)) |
| Lenovo | Joachim Löhr (jlohr@lenovo.com) |
| MediaTek | Ming-Yuan Cheng (ming-yuan.cheng@mediatek.com) |
| KDDI | Hiroki TAKEDA(ho-takeda@kddi.com) |

# **Discussion**

One key difference between [3] and [1][2] is their overall approach in reducing quantization error. [3] uses more bits (up to two BSRs) to encode buffer size. Whereas [1][2] always sends only one BSR but UE may use a new BSR table with smaller quantization error.

**Q1. Which of the following two options do you prefer for reducing quantization error in BSR?**

* Option 1a. UE always sends only one BSR. UE may use either the legacy BSR table or a new BSR table with smaller quantization error. UE chooses which BSR table to use based on its buffer size, e.g. use a new BSR table if its buffer size is within the range of the new BSR table or use the legacy BSR table instead.
* Option 1b. UE may send up to two BSR MAC CEs in one PUSCH transmission. These two BSRs are coupled, i.e. the first BSR indicates a coarse value of UE’s buffer size, and the second BSR refines the value reported by the first BSR. *Without loss of generality, let us assume in this discussion that either of these two BSRs can be based on either the legacy or a new BSR table.*
* Option 1c. UE sends only one BSR MAC CE in one PUSCH transmission, but the UE may report the overall buffer sizes for one LCG with two buffer size values in the BSR MAC CE: the first buffer size value indicates a coarse value of the LCG’s buffer size, and the second BSR refines the first buffer size. *Without loss of generality, let us assume in this discussion that either of these two buffer size values can be based on either the legacy or a new BSR table.*

In addition, the rapporteur suggests companies to discuss the pros and cons of these two options in the comments, e.g. whether it is more efficient than the other in reducing quantization error, its impact on network’s flexibility in scheduling and complexity of network’s UE implementation, etc.

|  |  |  |
| --- | --- | --- |
| **Company** | **Your preference**  (Option 1a/b) | **Comments**  (e.g. Pros and cons of these two options) |
| Qualcomm | Option 1a | Regarding Option 1b:   * If only legacy BSR tables are used, the maximum quantization error of the first BSR is ~10%, then the use of 2nd BSR can reduce it down to 1%. According to SA4’s TR, at 4K and 90 fps, the range of burst size is 28~208 KB. 1% of that corresponds to 280B ~2KB, which is at most one full PDCP PDU. So we are not sure if such a fine resolution in reporting is necessary or not, especially considering the extra overhead it introduces. * If new BSR tables are introduced, then a single BSR can offer a sufficiently good performance. Using the example above again, if the new BSR table uses linear distribution of 256 code points, then the resulting quantization error is 109B~810B, which is smaller than that of using two BSRs.   Based on the above analysis, we can see that a single BSR with a properly designed new BSR table in Option 1a can offer sufficiently good performance or beats the performance of Option 1b that uses legacy tables. Moreover, Option 1a has less UL overhead and is easier for network to decode received BSRs. Therefore, in our view, it is a better option to consider. |
| Nokia | See comments | Maybe better to have separate discussions about whether to allow fallback to legacy table and whether to allow two BS for an LCG as they are different issues.  Fallback to legacy table is at least needed if the new table does not cover full range (depends on the answer to Q3).  Two BS for an LCG could be used to reduce quantization error on top regardless of which table is used since the quantization error always remains. |
| ZTE | 1b/1c | In our view, this provides a simple mechanism to ensure an upper bound on the quantization error regardless of the new table design.  Firstly a “properly designed” new BSR table considering just the most likely application data packet sizes may not be sufficient to minimize quantization errors because the buffered data also depends on past scheduled data and newly arriving data. This will unfortunately require optimisation across the entire BSR range. As such covering all ranges of data sizes (especially now that we go towards even higher data sizes) with a very fine granularity would be impractical.  The main point is that eventually there will be gaps in these tables and it can happen that UE’s buffered data falls in these gaps, there by leading to quantization errors. This quantization error can be reduced by increasing the number of tables, but there will still be worst case values which will result in high degree of over reporting.  So, we think the solution proposed here to reduce the quantization error, especially when the quantization error exceeds some threshold (chosen by the network), is a simple way of achieving the goal.  It should be noted that the second index need not be always present. But, if the error is high, by including very few bits, we can reduce it to almost zero. With this approach, we also think we don’t need to define a large number of additional tables. And we can also have a simple table design which just covers a few additional high data code points. |
| LGE | Option 1a | For option 1a, it is simpler if the new table is defined. If the new table is defined, the existing BSR operation could be reused, which simplifies the spec and UE operation.  For option 1b, if two BSR indices are used, the design of new table may not be needed, which simplifies the discussion of design new BSR table(s). However, it is not desirable with following reasons:   * Generating two BSR MAC CEs for each LCG causes additional UE complexity to generate BSR table(s) and transmit the corresponding BSR MAC CE(s) * Two BSR MAC CEs cause the additional overhead, since it needs multiple MAC subheaders. * It also changes the procedure text of BSR operation, since in the current text specifies that only one BSR MAC CE is transmitted for multiple BSR triggering events * it is ambiguous whether the transmission of BSR is allowed or not when UL grant(s) can accommodate one BSR MAC CE for transmission but is not sufficient to accommodate two BSR MAC CEs.   For option 1c, it looks better than option 2b since it does not need to tramsmit multiple MAC subheaders. However, given that new BSR table is defined with finer granularity, one Option 1a is simple and sufficient. |
| NEC | Either | Either option can equally work well, we can follow majority. Pros and cons are analysed as following:  Option1a:   * only one BS field per LCG * assuming to keep the 8bits BS field, generally, the overhead may be slightly increased comparing legacy BSR report due to potential BS table indication field and new eLCID(s) * likely introduce New BSR MAC CE with BS table indication * Must discuss how to pre-define or configure new BS table(s)   Option1b/1c (assume a BS threshold will be configured and used to trigger second BSR):   * When buffer size is lower than the threshold, all legacy MAC CE format/procedure/LCID can be reused without any change * When buffer size is higher than the threshold. the first BS can be same as legacy, the second BS is reported with same or different format. potentially, this means nearly doubled overhead. * Potentially, no need to introduce new BS table(s), if the second BSR also use the legacy table . (i.e., no need to discussion Q2-Q7) * Potentially achieve smaller quantization error but with more bits in total |
| CMCC | Option 1a | We think that with new BSR table, one BSR is sufficient for reducing quantization error. Therefore, there is no need to introduce a secondary BSR.  For Option 1b, we think using two BSR MAC CE will increase overhead since BSR can be very common. Considering Long BSR, it can be even worse. |
| Ericsson | 1a | Reporting one BSR table value is sufficient (for each LCG/delay group in LCG).  We see no reason to send multiple BSRs to provide the same information, e.g. buffer size, if this can be done by transmitting one BSR. |
| Quectel | 1a | Option 1a is simple. Although option 1b and 1c may lead less wireless overhead, the triggering condition and the configuration is too complex. |
| Sony | 1b/1c | We think both options of either one MAC CE or two MAC CEs could be supported. This may carry the value from legacy BSR table and a delta compared to this value. |
| Samsung | 1a first, then 1b/c | We prefer to discuss first whether the new BS tables are fixed or dynamically constructed. When constructed dynamically, potentially a single BS field is sufficient to reduce the quantization error to a satisfactory level, provided that the NW constructs a table with appropriate ranges. We also believe that 1b/c is some kind of additional optimization regardless of the new BS table issue, so we should focus on what we agreed on earlier, step by step. |
| Apple | 1c and 1a | Obviously both Option 1a and 1b are beneficial in terms of reducing quantization error, so we would like to comment on the cons of these options:   * For Option 1a, RAN2 may need to make a lot of efforts to decide how the new tables should be constructed, what are the value ranges etc. This would be time-consuming especially if RAN2 decides to introduce multiple new tables. * For Option 1b, the UE may need to generate two BSR MAC CEs for one MAC PDU, which increases UE complexity during LCP, as well as extra overheads such as LCID and LCG ID etc.   Therefore, we think Option 1c is a better compromise between these two, where we can enjoy the benefits of lower quantization error of Option 1b, while without having to break the “one BSR MAC CE per MAC PDU” rule we currently have in TS 38.321 like in Option 1a. Moreover, the efforts of defining new BS tables could be minimized as even legacy tables can provide good performance in our understanding.  The downside of Option 1c is that we may need to introduce new BSR formats to accommodate two BS values per LCG, but we think in Rel-18 introducing new BSR formats is anyway inevitable considering all the potential enhancements for BSR.  Note that we think the UE should only use BSR formats offering finer granularity of BS value when the quantization error is too large.  If Option 1c is not acceptable, then we are fine with Option 1a which is the baseline for legacy BSR anyway. |
| vivo | Option 1a | The pros of Option 1a compared to Option 1b:   * Option1a is more compatible with the existing BSR procedure, i.e. Option 1a may cause less changes to existing specification regarding BSR generation; * Option 1a needs smaller total size due to at least one MAC subheader can be saved.   In the meanwhile, it is not clear whether Option 1b has smaller quantization error than Option 1a.  Regarding option 1c, we need to define the trigger condition for each BSR. |
| Intel | Option 1a | We prefer Option 1a, to send a single BSR. The UE could use RRC configuration (e.g. per MAC entity) or a threshold condition based on the buffer size to determine when to use the new BS table (i.e. to determine which format for MAC CE to use) in order to provide finer granularity for larger BS value and reduce quantization error.  Option 1b could also work, however, we do not see a clear benefit for sending two BSR MAC CEs for the same buffer size. It also adds unnecessary complexity, e.g. the additional BSR MAC CE contains additional fine indexing for multiple LCGs at the same time, the BS calculation needs to change, and on top of that the new BS table(s) anyway have to be introduced. Overall, the solution seems to have higher signalling overhead and spec impact. |
| Lenovo | 1c/1b | We agree with ZTE, that a simple method to report the large quantization error value with several additional bytes is enough. We prefer 1c, 1b would be also acceptable.  As shown in our contribution R2-2303203, evaluating the typical data rates and frame rates, the quantization errors range from several hundred or several thousand. To avoid the need to introduce too many new BS tables, RAN2 should consider using a buffer size table that indicates the quantization errors.  @NEC  For option 1b/1c: we think the quantization error value may be compared against a threshold. It can help determine to include the second BS or not.  @LGE  For 1a: it depends on the number of new tables to be defined and when to (re)configure the tables. As mentioned before we think it will be difficult to have only a limited set of new buffer size tables which cover all the possible packet sizes to decrease the quantisation error. It increases UE complexity to maintain the different tables from time to time.  For 1c: Add a second BS field in one BSR MAC CE is not complex,  We can discuss the truncated BSR as a stage3 detail, e.g., reusing the legacy principle. |
| MediaTek | Option 1a | We prefer one single BSR as legacy, which introduce less impact and less overhead with similar quantization error (if configured properly) and less complexity |
| KDDI | Option 1a | Share the view with Qualcomm |

Summary

(to be added later)

There have been different proposals on how new BSR tables may be introduced. For example, they may be pre-defined in specifications, generated on demand based on parameters configured by RRC, or a combination of these two approaches.

**Q2. Which of the following option(s) do you prefer for introducing the new BSR table(s)?**

* Option 2a. They are pre-defined in the spec;
* Option 2b. They are generated on demand based on a pre-defined formula whose parameters are RRC configured by network;
* Option 2c. Option 2a + 2b, i.e. a basic set of BSR tables can be pre-defined in the spec to cover common use cases, but network can configure additional BSR tables using one of the methods in Option 2b.
* Option 2d. They are generated based on a reference BSR table and a scaling factor RRC configured by network.

You may choose more than one option from the above in your reply. If possible, please also include your analysis on the pros and cons of these four options in your comment.

|  |  |  |
| --- | --- | --- |
| **Company** | **Your preference**  (Option 2a/b/c/d) | **Comments**  (e.g. Pros and cons of these options) |
| Qualcomm | Option 2a or 2c | From UE’s perspective, Option 2a is the simplest for UE to implement and yet serves the purpose well. Since new BSR table(s) only need to cover the size range of common XR encoding rates and frame rate, which are known, BSR tables can be predefined accordingly.  We understand that Option 2b can provide more flexibility for network. And if done right, it may be able to achieve lower quantization errors too. However, given the fact that the target range for new tables are known, we are not sure how much gain (e.g. in term of capacity improvement) Option 2b can offer and whether that would justify the extra implementation effort by UE. And the worst concern for UE implementation is that it is uncertain how much computing cycles it needs to budget for dynamic BSR table generation, because we don’t know how often network may ask UE to generate a new BSR table.  Therefore, Option 2c can be a good compromise for UE and network, because pre-defined BSR tables can help handle most of the scenarios and UE only needs to generate a new BSR table occasionally.  Option 2d can be an alternative to 2c if all the parameters of the reference table (e.g. min, max, distribution of its code points) can scale in the same way when encoding/frame rate changes. But that assumptions needs to be fully vetted before it can be considered. |
| Nokia | Option 2b or 2d | More flexible to cover all typical data rate and frame rate than 2a, without too much UE complexity.  No simplification for UE implementation with 2c compared to 2b since the UE would anyway need to implement both. |
| ZTE | Option 2a | We prefer a single additional table with focus on larger data packet sizes. |
| LGE | Option 2a  Acceptable for 2b  No for 2c and 2d | Option 2a is preferred since it minimizes the UE complexity using the new BSR table. If the UL XR traffic range can be covered using the several BSR tables, defining one or more fixed tables seems sufficient.  Option 2b is acceptable if it is the data volume range of UL XR traffic is diversified. In addition, no new BSR table would be needed in the future releases in order to support other types of traffic. The additional UE complexity depends on the details of the additional BSR table(s) (e.g., distribution of code points as in Q5).  Option 2c and Option 2d is not preferable since there is no additional benefits compared to option 2b. If the new BSR table(s) need to handle various range of data volume, option 2b seems sufficient. |
| NEC | Option 2b | Option2a has futureproof issue, and less flexible  Option2c basically specify two solutions for one issue, which is not the normal way we work in 3gpp  Option2d, with scaling factor, it means UE need to generate new table same as option2b. moreover, UE implementation may need to deal with non-integer values due to the scaling |
| CMCC | Option 2b | We think Option 2b can provide the least quantization error. UE or NW may trigger a BSR table update when it finding the quantization error between BSR table and actual buffer are too large.  For Option 2a, we think it has least impact on UE. However, XR can have multiple and/or unstable data rates, it’s difficult to the find all pre-defined BSR tables that suits for all or most XR applications.  For Option 2d, it’s easy for implementation as well, but it has the same problem as Option 2a, i.e., a suitable reference BSR table for XR service may not exist. |
| Ericsson | 2b | Configured/generated tables are really the only solution that is needed. This option gives the largest gain (shown to be close to the ideal case in simulations) and is a clean and easy to understand solution which is matching legacy operation of BSR reporting. It will have a low cost since the generation is only done when changes are needed (rare occasion if a few tables are generated from the start and selected from). Thus the benefit of introducing new pre-defined tables is hard to justify. Pre-defined tables will never be able to cover all the ranges that is needed and at same time keep the granularity high enough to get good gains. The range is shown (in simulations) to be dependent both on the traffic sizes and on the transmission sizes. Both are factors that will change with time and can't be known beforehand when defining the tables.  Legacy tables will always be an option for all solutions (those are of course pre-defined).  For option 2c we have done some comparison simulations on this option (of dividing/scaling every step size) and it seems to not be as good as the configurable table solution:    As can be seen with higher indexes the granularity is still low with the scaling solution, so it doesn’t actually solve the problem with low granularity for large sizes. And with increasing bitrates/transmission sizes the problem will only become larger. |
| Quectel | 2a/2b or 2c | Considering the difference character of XR sessions, it is benefit to predefine some tables, and let some space for gNB configuration. |
| Sony | 2a/2d | We think 2a is straightforward |
| Samsung | Option 2b | Option 2a causes another non-trivial issue that how to determine the range the new tables should cover, and how to design code points. Also, it has future-proof issue.  Option 2b is sufficient to provide enough flexibility and scalability considering current XR traffic pattern and its evolution in the future. We can discuss the way to handle UE complexity issue, if needed  Option 2c seems to impose higher burden on UE implementation.  Option 2d cannot fully resolve the quantization issue when the table is scaled to higher volume range, since the quantization error (the BS interval between two adjacent code points) is also scaled. |
| Apple | Option 2a, but … | As mentioned by some companies, Option 2a is simplest for UE implementation.  If we are going to use semi-static BS tables (e.g. Option 2b/2c/2d), since it is the UE who can directly observe the quantization error, as well as tracking the UE application activities, we think some we should allow the UE to express some preference and recommendations about the BS table parameters. |
| vivo | Prefer option 2a. | Option 2a has the following pros compared to the BSR tables specially optimized for certain XR traffic:   * It can be expected to be simplest among these options. * Fixed table(s) for common use is not sensitive to the rate adaptation of the XR traffics.   Given fixed table(s) of finer granularity is used, the additional quantization error reduction by further refining the BSR table for specific scenario, if there is, could be very marginal. It is not worthwhile to pursue the marginal additional gain at the cost of considerable implementation complexity and standardization effort increase. |
| Intel | Option 2a (preferred) or Option 2b (with comment) | Option 2a: This is our preferred option to have fixed BS table(s). We believe this option is the most straightforward approach where finer granularity can be easily achieved e.g. by having more number of bits in the BS field size (e.g. 10 bits to have 1024 codepoints). Such fixed table could cater to most cases for XR traffic since there are only a limited number of frame rates that need to be supported anyway.  Option 2b: This option of semi-statically RRC configured table (e.g. max, min and step size parameters) is acceptable to us if it is majority view. This could offer some flexibility and may be more forward compatible to other use-cases for XR in future releases (even if not required for Release 18).  Option 2c: This does not seem like a compromise since both approaches need to be specified. RAN2 will need to further discuss how the UE shall implement/support both kinds of tables (i.e. fixed and semi-static) e.g. this could depend on new/different UE capabilities. We don’t see the benefit of adding this complexity even with the understanding that in most cases only the fixed table(s) will be used.  Option 2d: We believe this is a subset of Option 2b, since the scaling factor is a configurable parameter. |
| Lenovo | Option 2a  Acceptable for 2b | We prefer 2a for simplicity. 2b may be also acceptable given that benefits can be proven. |
| MediaTek | 2b | 2b is more futureproof, 2a can’t support future XR services when its data rate is out of the new BSR range. |
| KDDI | Option2c | We understand that Option 2a can address some of the existing XR use cases, but it’s better to have option 2b also to address potential use cases. |

Summary

(to be added later)

To either pre-define or RRC configure a new BSR table based on a formula, one needs to decide on three factors: the range of buffer sizes in a table, number of code points, and the distributions of code points within the range. Some of these factors may need to be considered together. For example, the choice in number of code points may affect the choice on the range of a table, and vice versa. Or the choice in the distribution of code points may depend on the choice in the range or number of code points, and vice versa. We discuss these issues in the following.

For the range, the rapporteur thinks that there can be at least two possible options: either reuse the same range of the legacy BSR table or define a narrower range, e.g. based on the sizes of data bursts produced based on common XR encoding rates and frame rates. In the first option, quantization error can be reduced through techniques such as use of more code points or more efficient distribution of code points.

**Q3. What range of buffer sizes should new BSR table(s) have?**

* Option 3a. Reuse the same range of the legacy BSR table;
* Option 3b. A narrower range, e.g. based on the sizes of data bursts produced based on commonly used XR encoding rates and frame rates
* Option 3c. It depends on other options. No need to impose anything for now.

|  |  |  |
| --- | --- | --- |
| **Company** | **Your preference**  (Option 3a/b/c) | **Comments** |
| Qualcomm | Option 3b | We think there are two possible dimensions in reducing quantization errors: reduce the range of a table vs increase number of code points. Between these two choices, we think increasing number of code points is less desirable, because it will increase UL overhead and make the design of new BSR MAC CE more complicated. On the other hand, reducing the range of a table has much less overall impact on the current BSR framework. |
| Nokia | Option 3b | Finer granularity with narrower range. No need to cover full range as legacy table can be used for smaller buffered data, otherwise if with 3a it would have worse granularity for some code points than legacy table if to have finer granularity for others? |
| ZTE | Option 3b | As shown in our contribution, the problem is higher towards higher BSR indices (when there is more buffered data). This is because the code points are sparser in this region. So, targeting these regions seems to make sense. On top, if we have some fail-safe mechanism to ensure that the quantization error never exceeds a given value (like including second index if it exceeds), then the design can simply focus on the higher end of the buffer sizes (and typical frame sizes for XR traffic etc). If we have no such fail-safe mechanism, some more detailed analysis may be needed to see how to optimise over entire range. |
| LGE | 3c | It depends on the result of Q2.  If Option 2a is agreed, it should be determined based on the characteristic of XR traffic.  If Option 2b is agreed, it depends on the network configuration. |
| NEC | Option 3b | Optin3a (cover 0 to infinite) does not work well:   * With the same bits, quantification error is reduced in a certain buffer size range, while increase in other buffer size range   Or much more bits is required for BS report |
| CMCC | Option 3b | We think that this issue depends on the characteristics of XR traffic. Since XR traffic consists mainly of periodic Data bursts with finite size, a BSR table whose range covers Data burst seems fine |
| Ericsson | 3b/3c | The range of the tables depend on how the NW configures them and thus what is suitable for the traffic. This is not something we can decide on here.  It is likely that the configured tables will be narrower than legacy tables, to make possible for higher precision on some specific range, but it all depends on the bits/code points used. |
| Quectel | 3b |  |
| Sony | 3b |  |
| Samsung | Option 3c | It is NW implementation issue, if BS table is constructed based on NW configuration. If it is agreed to pre-define fixed new BS table(s), then Option 3b is more desirable. |
| Apple | Option 3b/3c | In general we believe a narrower range is sufficient, but we think we should leave it open for the time being. |
| Vivo | Option 3c | Of cause, finer granularity could obtained for narrower range. But this issue depends on the output from Q1 and Q2. Let’s discuss this issue when there is conclusion regarding Q1 and Q2. |
| Intel | Option 3a (if option 2a is agreed for Q2) or option 3b/3c (if option 2b is agreed for Q2) | We think different options for the range of buffer sizes may be applicable depending on how the BS table is defined (which is discussed separately in previous Q2).  **Approach 1: option 3a if BS table is predefined in spec. (i.e. option 2a is agreed in Q2).** Our preference in this case is option 3a with the following reasoning:  *Upper limit:* RAN2 uses the same upper limit Bmax (>81Mbytes) as in current BS table for the additional BSR table. For BS value of 81 Mbytes, with 2 bursts buffered and 60 fps packet arrival rate, the supported throughput can be calculated as 81/2 Mbytes \* 8 bit / Byte \* 60 / second = 19.44 Gbps, which seems sufficient for XR traffic.  *Lower limit:* We keep same lower limit Bmin (0kbytes), however, more code points are needed for finer granularity. The benefit in this case is that based on a threshold condition as explained in Q1, the UE can use the same new BS table for all LCGs within a BSR, and per LCG configuration is not needed.  **Approach 2: option 3b/3c if BS table is configured semi-statically via RRC (i.e. option 2b is agreed in Q2).** We could accept option 3b/3c for the case of RRC configured semi-static table with the following reasoning:  *Upper limit*: Same as legacy, for the reason explained above in approach 1).  *Lower limit*: Can have a higher lower limit (e.g., 20Mbytes). In this case the BS field size could be the same as legacy e.g. 8 bits. However, per-LCG table selection can be configured using semi-static RRC parameters.  Per-LCG table selection in Approach (2), which appears to be the majority companies’ view, seems more efficient, but we would like to point out that it will likely increase the decoding complexity of the MAC PDU carrying the BSR. Additionally, a new mapping table for per-LCG association to a BS table may be needed. On the other hand, Approach (1) of having uniform configuration of BS table across all LCGs in the BSR, especially if new BS table only uses 1 or 2 additional bits for the BS field size would have a comparable (or potentially lower) overhead. In summary, comparing these two approaches, we believe **Option 3a is straightforward without introducing much signalling overhead** (as the same range of BS values is used for both the new and legacy BS tables).  In summary, our preference is option 3a (as explained in our response to previous question Q2), although we also understand that there might be large support to provide some flexibility in which case option 3b is also acceptable to us. |
| Lenovo | Option 3b | Agree with ZTE. As shown in our contribution, the quantization errors range from several hundred to several thousand bits, a narrower range with a maximum BS value set to 10000 bits to cover the quantization error is sufficient. |
| MediaTek | Option 3b |  |
| KDDI | Option 3b |  |

Summary

(to be added later)

For the number of code points, the rapporteur thinks that there can be at least two possible options (for both RRC configured and predefined tables): all new BSR tables have the same number of code points or different new BSR tables may have different number of code points. The first option would simplify the design and implementation of the enhanced BSR MAC CE, whereas the second option maximizes the flexibility in defining/configuring new BSR tables.

**Q4. Which of the following is your preferred option for the number of code points in a new BSR table?**

* Option 4a. All new BSR tables have the same number of code points;
* Option 4b. Different new BSR tables can have different number of code points (e.g. depending on their ranges);
* Option 4c. Other (Please provide details in your comment)

|  |  |  |
| --- | --- | --- |
| **Company** | **Your preference**  (Option 4a/b/c) | **Comments** |
| Qualcomm | Option 4a | This choice is a tradeoff between performance and complexity. Theoretically, Option 4b probably has a better performance in reducing quantization errors than Option 4a, because tables with large ranges can benefit from having more code points. However, if network can configure different LCGs to use different BSR tables, then having different BS field lengths for different LCGs can make the format of the new BSR MAC CE much more complicated 🡪 not desirable for UE implementation. |
| Nokia | Option 4a | 8-bit table(s) are enough. |
| ZTE | 4a | We think just one additional table would be sufficient. |
| LGE | 4a | We prefer to define same size of BS field in order to simplify the new BSR MAC CE design, given that each LCG may use different BSR table(s) (related to Q6). Furthermore, the number of code points of BS field should be same as legacy BSR table (i.e., 8 bits for long BSR format), given that some LCGs may use the legacy BSR table. |
| NEC | Option4a | 8bits BS table.  it is likely that a long or long truncated BSR MAC CE includes some BS fields encoded based on a legacy 8bits BS table, and some other BS field encoded based on a new BS table, it would be easier to define the MAC CE format if we keep all the BS fields with same length. |
| CMCC | Option 4a | We think that using the same bit length has less complexity for implementation.  Besides, we think 8 bit is enough for XR, so Option 4b is less preferred. |
| Ericsson | Option 4b | There are mainly two aspects coming into this, how many bits and how many indexes that are used. Both of these can vary, e.g. we can have the same amount of bits as today or even less. Then what indexes are used is up to NW when deciding to configure the table, it may even decide to not use all indexes.  More bits/code points give higher precision and/or larger range, which may lower the number of tables needed. But fewer code points reduce overhead and can allow for reporting other information. This depends on what BSR formats that is selected to be used (e.g. new formats with other length than legacy may be introduced). |
| Quectel | 4a and 4b | For the predefined table, the code point is fixed. But for the RRC configured BSR table, the number of code point and the exact value depends on the configuration. |
| Sony | 4a |  |
| Samsung | Option4a with comment | Our preference is, potentially, if there are multiple new BS tables, they may have the same number of code points. But, we fail to see any constraints that longer than 8-bit BS field should be precluded for those new BS tables. As long as the byte align is achievable, e.g., with mixed 2-byte and 1-byte BS fields, a new BSR format can be designed decently, given the signalling on which BS table is used for a certain LCG. |
| Apple | It depends | For now, we tend to think it depends on what the “new table” is used for. For instance, if the new table is used for the second BSR or the second buffer size value in Option 1b/1c in Q1, probably we need fewer than 8 bits to indicate the differential BS value.  Therefore we prefer to keep the Option 4b open. |
| vivo | Option 4b | If new fixed BSR table is defined, it is better to have more code points than the legacy table. Since new BSR format MAC CE would be introduced anyway, it is not necessary to stick to 256 codepoints.  Option 4a is also acceptable with less implementation complexity. |
| Intel | Option 4a | Since this is possible, we don’t think the complexity of option 4b is needed which adds more variability and/or need for multiple new BSR MAC CE formats. |
| Lenovo | Option 4a | Agree with ZTE. A 5bit BS table for the second BS would be sufficient in our view. |
| MediaTek | Option 4a |  |
| KDDI | Option 4a |  |

Summary

(to be added later)

For the distribution of code points, three options have been proposed in contributions: exponential (as in legacy, the ratio between a step size and its associated buffer size is a constant across all code points), linear (step size for each code point is a constant), and truncated Gaussian [2]. A sensible choice in the distribution of code point may depend on factors such as range and number of code points of a BSR table, as well as traffic characteristics (e.g. size distribution of data burst).

**Q5. Which of the following is your preferred option for the distribution of code points for new BSR table(s)?**

- Option 5a. Exponential distribution, i.e. The same as in legacy;

- Option 5b. Linear distribution, i.e. equal interval between any two consecutive code points;

- Option 5c. Truncated Gaussian distribution;

- Option 5d. Other (Please provide details in your comments).

You may choose more than one option from the above. In that case, please provide the criteria for each selected option in your comment.

|  |  |  |
| --- | --- | --- |
| **Company** | **Your preference**  (Option 5a/b/c/d) | **Comments** |
| Qualcomm | Option 5a and 5b | In our understanding, exponential distribution is better suited for a large range (e.g. across several orders of magnitude), whereas linear distribution is better suited for a small range. Hence different new BSR tables may benefit from using different distributions.  Our understanding on truncated Gaussian distribution is that it is just a model used in RAN1’s evaluation study. It needs to be vetted whether it matches well with actual XR traffic generated by different codec algorithms and how forward compatible it can be. |
| Nokia | Option 5b or 5c | Depends on the option adopted for question 2.  5b with linear distribution is simpler for 2b with formula-based calculation.  5c could match with the traffic distribution better for 2d with scaling on top of the reference table without the need to define the Gaussian formula. |
| ZTE | 5a | Some exponential distribution optimising for higher data sizes would be suitable. |
| LGE | Depends on Q2;  (5a/5b for Option 2a,  5b for Option 2b) | If the new BSR table is generated by UE using formula (i.e., Option 2b in Q2), it should follow the linear distribution, in order to minimize the additional UE complexity.  If the new BSR table is specified (i.e., Option 2a in Q2), we are okay with option 5a and 5b. However, if the new table can be used other than XR services (related to Q8), the option 5c is not needed, since the benefits of option 5c would be limited to video traffics. |
| NEC | Option 5b | Option5b is simple and enough. NW can configure the (min, max ) properly to guarantee the quantization error is low |
| CMCC | Option 5c | For Option 5c, there are serval studies and simulations show that the size of video frame follows truncated Gaussian distribution. In AR/MR, video frames consist most of uplink data.  And for other information in uplink (e.g., pose, controller, audio) are not really sensitive to quantization error, therefore the need for BSR enhancement is mainly driven by video frames.  So Truncated Gaussian distribution should be introduced.  Option 5b is also acceptable for us, it can be utilized for pose and control information, they are relatively fixed in size. |
| Ericsson | 5b/5d | Linear distribution seems to be the simplest choice (and is found in simulations to be working well) for generation and configuration. However, if there is shown that there is a benefit to have other distributions (and the complexity to generate those is not a concern) then such distribution may also be considered. |
| Quectel | 5b | Since the new table only prefer a limited scope, 5b is enough. |
| Sony | 5b |  |
| Samsung | Option 5b | It is sufficient to have exponentially distributed legacy BS table, which already covers entire range of buffer size. For new BS table(s), we should focus on a certain range of buffer size tailored for the application of interest. With this in mind, linear distribution is an efficient way to manipulate quantization level, and easy to implement compared to other options. |
| Apple | Option 5b | We would like to keep it simple. |
| Vivo | Option 5a | For XR, there are much more P-frames than I-frames, and the total traffic volume for P-frame could be very much larger than that for the I-frames. With exponential distribution, the smaller burst corresponding to P-frames gets better granularity, which can reduce the quantization error in statistics.  Again, this is also related to the questions we discussed above in Q1 and Q2. |
| Intel | Option 5a (if option 2a is agreed for Q2) or option 5b (if option 2b is agreed for Q2) | Different options can be considered depending on the outcome of previous questions. For example:  Option 5a (preferred): if same range of BS values as legacy is used with additional bits i.e. extended BS field size, for example using 10 bits rather than 8 bits.  Option 5b: can be used if BS values of new table is over shorter range than legacy when using semi-statically configured BS table(s) in Q2.  Option 5c: We are not sure of the benefit of using a gaussian distribution since range of BS values in the new table(s) will be deterministic (predefined or semi-statically configured). |
| Lenovo | 5a/5b | We prefer 5a. If there’s shown there is gain when using 5b for the second BS in 1c/1b, then it may also be considered. |
| KDDI | 5b | If we go with a narrower range, then simple option 5b is enough. |

Summary

(to be added later)

There are a number of contributions on the granularity for using new BSR table(s). Most of them have proposed that network can configure on a per LCG basis which BSR table(s) UE should use, e.g. LCG #1 may use the legacy BSR table but LCG #2 may use one of the new BSR tables, and so on. On the other hand, it is also possible that in some solutions, it may be simpler for all LCGs in a BSR MAC CE to use the same BSR table.

**Q6. Which of the following is your preferred granularity for using new BSR table(s)?**

- Option 6a. Network can configure which BSR table(s) (either legacy or new) an LCG should use;

- Option 6b. All LCGs in a BSR MAC CE use the same BSR table;

- Option 6c. Other (Please provide details in your comment)

|  |  |  |
| --- | --- | --- |
| **Company** | **Your preference**  (Option 6a/b/c) | **Comments** |
| Qualcomm | Option 6a | Different LCGs can have different data rates and different burst sizes. It hence makes sense to configure BSR table on a per LCG basis. |
| Nokia | Option 6a | Per LCG makes sense as not all the LCGs are for XR and different LCHs/LCGs might have different data rate. |
| ZTE | Option 6c | We prefer that any new BSR mechanism would be per LCG (same as legacy). Perhaps this could be agreed as an independent agreement regardless of other enhancements.  But considering that the buffer size cannot be predicted accurately, which BSR table(s) is used should be selected based on the Buffer size to be reported.  And the table used is identified by the LC-ID. |
| LGE | Option 6a with comment | Regarding the granularity of BSR table, it should be configured per LCG since each LCG has different range of data volume.  However, we think network can also configure to use “both legacy and new BSR tables” for an LCG. Then, depending on the size of the buffered data, UE can decide the appropriate BSR table. |
| NEC | See comment | This question is a bit confusing.  Two possible aspects needs discussion:   1. whether new BS table is configured per LCG or per UE(same for all LCG ) 2. whether UE has to switch to use new BS table for all LCG once configured   **For aspect 1: 6b.** we are not sure if there will be two LCG requires new and different BS tables (i.e., support two XR service at the same time). so per UE may be enough, but per LCG (6a) is acceptable  **For aspect 2: 6c, UE choose**  Option6a cannot work, since the buffer size may fall out of the new BS table’ coverage, legacy BS table has to be used.  Option6b cannot work, since some LCG will never need new BSR table  Option 6c: UE use new BS table if the buffer size fall in the size range (min,max) of new BS table, otherwise ,use legacy BS table. |
| CMCC | Option 6a | When different QoS flows are mapped into different LCGs, BSR table configured per LCG can provide the least quantization errors compared to Option 6b. |
| Ericsson | 6a/6c | NW configures which table(s) that are applicable for each LCG. When reporting the BS, the UE should select the table and index which results in lowest inaccuracy (i.e. lowest difference between the max and min value of the index). If the configured tables do not contain an index covering the current buffer size, then UE should use legacy tables and BSR.  This also means that the same table cannot be used for all LCGs as they may have different buffer sizes and thus fit different table ranges. An example of what happen if you don’t use the suitable table for the buffer range is shown below, where an LCG is configured to only use a new configured table. It is clearly worse than using the adaptive approach, i.e. selecting the legacy table when outside of the new table range.  image |
| Quectel | 6a | The gNB shall configure it based on the XR session feature. |
| Sony | 6a |  |
| Samsung | 6c | NW configures which LCG(s) can (is permitted to) use new BS table(s). The selection of suitable BS table for the configured LCG(s) when reporting buffer size should be determined per LCG based on actual buffer size. |
| Apple | Option 6a and Option 6c | In general we agree BSR table should be LCG-specific to accommodate different traffics.  However, even for XR traffics, when the amount of buffered data is low enough, legacy BS tables could be sufficient. So we think table selection may also depend on the volume of buffered data. |
| vivo | Option 6a | The motivation to introduce the new BSR table is to reduce the quantization error for large data burst of XR traffic. For other traffic/signalling, there is no quantization error issue identified with the legacy BSR table. In such sense, it is preferred that the new BSR table can be configured for the LCG which requires better granularity. |
| Intel | Option 6b (if option 2a is agreed for Q2) or option 6a (if option 2b is agreed for Q2) | Our preference is option 6b (all LCGs in a BSR MAC CE use the same BSR table) considering that our preference in Q2 is to use a new table that extends current BS field size by few bits only.  As also explained in Q3 above, per-LCG table selection will likely increase the decoding complexity of the MAC PDU carrying the BSR and a new mapping table for per-LCG association to a BS table may even be needed. We believe uniform use of fixed new BS table (over legacy range and additional bits for BS field) across all LCGs in a BSR may be more straightforward with comparable (or potentially less) signalling overhead in comparison to per-LCG configuration.  We are okay with Option 6a if RAN2 decides (in Q2) to use RRC configured table(s) that could be of variable size instead of fixed new table(s). |
| Lenovo | 6a | A configuration per LCG is sensible since not all the LCGs are used for XR traffic/services and different LCHs/LCGs might have different data rates. It is OK for us the further discuss whether network can also configure to two BS tables for an LCG and UE decides the appropriate BSR table. |
| MediaTek | Option 6a |  |
| KDDI | 6a/6c | We share the view with Ericsson, it’s better to specify a mechanism which enables UEs to select the appropriate table among the multiple configured BSR tables. |

Summary

(to be added later)

In legacy, short BSR and long BSR use different BSR tables, because they use different number of code points. If we are going to introduce new BSR tables, then we need to discuss whether/how new BSR tables should be designed for them.

**Q7. Which of the following is your preferred option for introducing new BSR table(s) for short/long BSR?**

- Option 7a. Only long BSR need to have new BSR table(s);

- Option 7b. Only short BSR needs to have new BSR table(s);

- Option 7c. Both short BSR and long BSR can have their own new BSR table(s), which are defined/configured separately;

- Option 7d. The same set of new BSR table(s) are used by both short BSR and long BSR.

- Option 7e. Introduce new BSR formats to accommodate new BSR table(s).

|  |  |  |
| --- | --- | --- |
| **Company** | **Your preference**  (Option 7a/b/c/d) | **Comments** |
| Qualcomm | Option 7a | First, we think it is useful to keep the 2B short BSR MAC CE. For small bursts, the current short BSR table is sufficient, because its distribution of code points has fine resolution at the low end.  To cover the case where a single LCG has a large burst, we think RAN2 should agree that UE is allowed to use long BSR in that case. Then new BSR table(s) designed for long BSR can be used for to provide better resolution for large bursts, if needed. |
| Nokia | Option 7a | Long BSR provides finer granularity since it provides a lot more code points compared to short BSR, leading to minimizing the quantization error. |
| ZTE | 7a |  |
| LGE | Option 7a with comment | We think that the new BSR table for long BSR should be defined first.  Not sure about the new table for short BSR. We discuss later. |
| NEC | Option 7d or 7a with Comment | We can increase BS field bits for short BSR, as same as for long BSR. Then same set of BS table(s) can be used by both short BSR, long BSR, short/long truncated BSR.  7a is also fine with us, but it would mean short/short truncated BSR will not use new BS table and keep the quantization error as today |
| CMCC | Option 7d | We think Short and Long BSR should have the same performance on quantization errors, therefore there is no need to just configure new BSR table for short or long BSR.  Besides, Long BSR for reporting single LCG is less preferred since it has to transfer an 8-bit LCG map instead of 3-bit LCG ID. |
| Ericsson | 7e | There is no need to limit this. New tables can be used for any BSR format. New BSR formats can be introduced with different lengths. |
| Quectel | 7e |  |
| Sony | 7a |  |
| Samsung | Comment | We think what we should discuss here are two separate issues:   1. Whether we need to define new BS tables for a) 5-bit BS field, b) 8-bit BS field, c) longer than 8-bit BS field. 2. Whether we should use longer than 5-bit BS field when only one LCG having data.   For 1), we don’t think it is necessary to define new BS table for 5-bit BS field.  For 2), we think it is worth considering using longer than 5-bit BS field when only one LCG having data. |
| Apple | 7a and 7e | While we think long BSR should be considered, we must point out that many potential BSR enhancements would need RAN2 to define new BSR formats anyway. Thus, we think new BSR formats should be taken into account together.  (In our understanding, if new BSR table is used in existing long BSR, then it is still considered as a new BSR format due to e.g. new LCID.) |
| vivo | Option 7d. | Short BSR could be more frequently used than the long BSR since typically only single LCG has data when the UE only has XR traffic. Considering this, it is preferred that both short BSR and long BSR should use the same new BSR table with more codepoints. This would result in the size increase of short BSR. But it is a minor cost considering the large data volume of XR traffic. |
| Intel | Option 7a | Since the main motivation is to reduce quantization error, we believe long BSR is sufficient as short BSR carries minimal information for a single LCG, with maximum BS value <1Mbytes in the legacy BS table, which may not be the usual case for XR traffic. |
| Lenovo | 7d/7e/7a | Supporting second BS field in 1C needs to define new format. The quantization error can happen in case of a single LCG having XR large bursts, therefore it’d better to also support it for short BSR. |
| MediaTek | Option 7a |  |
| KDDI | Option 7a |  |

Summary

(to be added later)

Last but not least, there was discussion near the end of the online session on whether new BSR table(s) is available only to XR UEs or to any UEs. Let us continue that discussion here to collect more views.

**Q8. Do you think new BSR table(s) is available only to UEs supporting XR services or to any UEs?**

- Option 8a. Only UEs supporting XR services;

- Option 8b. Any UEs

|  |  |  |
| --- | --- | --- |
| **Company** | **Your preference**  (Option 8a/b) | **Comments** |
| Qualcomm | Option 8b | We do not see any strong reasons why a new BSR table cannot be used by UEs not supporting XR services. Moreover, use of new BSR tables is fully under network control, i.e. if network does not want a UE to use a new BSR table, it can simply not enable or configure that BSR table for the UE. |
| Nokia | Option 8a for now | We can start with 8a when designing the parameters/values for the new table(s).  It can be discussed later if need to apply to other UEs. |
| ZTE | 8b | As normal, we assume the UE will indicate support for these and if supported, the network can configure the UE to use these whenever it is appropriate. Whether an XR service is running at this point or not may be irrelevant (what matters is the configuration that the UE receives). |
| LGE | Option 8b | It is up to the network configuration. |
| NEC | Discuss later | We can discuss this later stage when we discuss capability. This would be a question not only for BSR enhancement but also for other enhancements |
| CMCC | Option 8b | For traditional service like live streaming and electronic games, new BSR table is more suitable compared to legacy BSR table, therefore new BSR table should be available to all UEs |
| Ericsson | 8b | The feature may mostly be useful for XR services, but it is not necessarily limited to XR and can be used for other services if such benefit is seen. It is depending on UE capabilities and network decision to configure new tables. |
| Quectel | 8b |  |
| Sony | 8b | Based on UE capability |
| Samsung | Option 8b | Our preference is based on the understanding that ‘any UEs’ means any UEs supporting new BS table capability. The fundamental question should be whether we allow the capability of supporting new BS table to be a standalone capability apart from XR service. |
| Apple | 8b | This is anyway UE capability discussion. |
| vivo | Option 8b | Given fixed BSR table for common use is defined, it is not necessary to restrict the table for XR traffic only. One capability indicator can be defined to indicate if the UE supports the new BSR table or not, and it depends on the NW to configure whether/how to use the new BSR table. |
| Intel | Option 8b | Even though new BS table(s) may only be used by XR services, we don’t think such service-based restriction is needed from RAN2 point of view. |
| Lenovo | 8b | It is not necessary to restrict this to only UEs supporting XR services. The new table can be enabled to use by NW according to UE capability. |
| MediaTek | Option 8b |  |
| KDDI | Option 8b |  |

Summary

(to be added later)

# **Conclusions**

(To be added later)

# References

1. R2-2302515, BSR enhancements for XR, Qualcomm Incorporated.
2. R2-2303862, BSR enhancements for XR, Nokia, Nokia Shanghai Bell.
3. R2-2302851, BSR enhancements for XR, ZTE Corporation, Sanechips.