

RLC Window Size and Acknowledgement with FLO

1. INTRODUCTION

The RLC protocol performance is ruled among other things by its window size. Typically, a good adequation between the window size, the maximum rate at which blocks are sent/received, and the available radio resources the protocol can use is needed to ensure a good operation of this protocol. In GPRS, a window size of 64 is available which shows its limitations when a large amount of timeslots is in use leading to quick and frequent window stalling. In EGPRS, this problem was overcome by defining a window size respective to the amount of timeslots allocated. The downside of a large window size however is its inherent need for a large acknowledgement bitmap. Ideally the RLC receiver should in fact be able to send an acknowledgement bitmap covering the whole receive window. But this bitmap may not fit within one RLC/MAC control block. Therefore, a rather complex technique using bitmap compression, dropping of channel measurements, partial bitmap reporting as well as polling was introduced in EGPRS.

The introduction of FLO in GERAN should allow for a good RLC performance while avoiding GPRS limitations with multislot scenarios and EGPRS complexity for acknowledgements. This contribution therefore analyses these two correlated issues of window size and acknowledgement for FLO.

2. (E)GPRS RLC LIMITATIONS

The RLC protocol used for FLO should tackle the limitations of (E)GPRS RLC protocol and therefore should:

- Avoid loss of throughput caused by frequent window stalling due to too small window size when multislot is used, by defining a larger window size than in GPRS, following EGPRS
- Fit both acknowledgement information *and* channel quality report within one RLC/MAC control block to avoid EGPRS problem of dropping channel quality measurements
- Avoid complex bitmap compression/decompression techniques if not necessary
- Avoid complex polling mechanism

3. SIMULATION MODEL AND ASSUMPTIONS

3.1 General

In order to assess both window size effect and acknowledgement on RLC performance during data transfer, the most challenging case is a situation where a TBF operating in RLC acknowledged mode is using all the resources on the allocated timeslots, and has an uninterrupted flow of data (i.e. there is always incoming RLC SDUs to transmit). Such a case was simulated under the following assumptions

- GPRS RLC protocol with pre-emptive retransmissions
- CS-3 link performance in TU3iFH, no link adaptation
- 2, 4, 6 and 8 timeslots
- In order to fit channel measurements in every acknowledgement message, a maximum bitmap size of 116 bits is estimated to be available (see appendix 1) and was used in the simulation.
- Window sizes of 64, 116, 256 and 512
- Polling period of 24 RLC data blocks
- Round-trip delay of 120ms and 200ms
- No incremental redundancy nor independently coded RLC/MAC headers

3.2 Acknowledgement bitmap construction

It is assumed that bitmap compression techniques are not applied when RLC acknowledgement messages are constructed. Therefore, in case the bitmap size in the RLC acknowledgement message is shorter than the RLC window size (receive window), special consideration need to be taken. To this end two partial bitmap approaches have been simulated that are described below.

Either of the approaches 1 and 2 are under control of the RLC receiver. The RLC transmitter only need to poll for an acknowledgement (i.e. one polling bit) without a need for extended polling as in EGPRS.

3.2.1 Partial bitmap approach 1

In this first approach, the SSN (Starting sequence number) is determined as follows:

- If $[V(R)-V(Q)] \bmod SNS \leq \text{bitmap size}$, then $SSN=V(R)$ and the bitmap covers a certain amount (determined by the bitmap size) of RLC data blocks having a BSN smaller than the SSN
- If $[V(R)-V(Q)] \bmod SNS > \text{bitmap size}$ then, $SSN=[V(Q)+\text{bitmap size}] \bmod SNS$ and the bitmap covers the RLC data blocks having a BSN within the interval $[V(Q), SSN-1]$

Since the RLC data block pointed by the $V(Q)$ state variable is included in the bitmap in every case, the BOW (beginning of window) bit can be set to '1'. This means that the RLC transmitter can assume an implicit acknowledgement to the RLC data blocks which have a BSN within the interval $[V(A), SSN - \text{bitmap size} - 1]$ if $V(A)$ is not explicitly covered by the bitmap.

3.2.2 Partial bitmap approach 2

In this second approach the RLC transmitter updates the PBSN parameter¹ as specified in 3GPP TS 44.060 and fills the bitmap as follows:

- If $[V(R)-V(Q)] \bmod SNS \leq \text{bitmap size}$, then $SSN=V(R)$ and the bitmap covers a certain amount (determined by the bitmap size) of RLC data blocks having a BSN smaller than the SSN.
- If $[V(R)-V(Q)] \bmod SNS > \text{bitmap size}$, then the mobile station determines whether a first partial bitmap or a next partial bitmap is transmitted as follows:
 - If $[PBSN-V(Q)] \bmod SNS < WS$ and if $[V(R) - (PBSN + PBSN_Threshold)] \bmod SNS < WS$, then a next partial bitmap is sent
 - Otherwise a first partial bitmap is sent

PBSN_Threshold is a threshold value intending at identifying a limit for the difference between PBSN and V(R), beyond which a next partial bitmap is generated

If a first partial bitmap is sent, then $SSN = [V(Q) + \text{bitmap size}] \bmod SNS$ and the bitmap covers the RLC data blocks having a BSN within the interval $[V(Q), SSN-1]$

If a next partial bitmap is sent, its characteristics are:

- If $[V(R) - (PBSN + 1)] \bmod SNS > \text{bitmap size}$ then $SSN = [PBSN+1+\text{bitmap size}] \bmod SNS$ and the bitmap covers the RLC data blocks having a BSN within the interval $[PBSN+1, SSN-1]$
- If $[VR) - (PBSN+1)] \bmod SNS \leq \text{bitmap size}$, then $SSN = V(R)$ and the bitmap covers a certain amount (determined by the bitmap size) of RLC data blocks having a BSN smaller than the SSN

The BOW is set to '0' when a next partial bitmap is transmitted. It is set to '1' otherwise.

3.2.3 Network-controlled partial bitmap approach

In this approach, the network controls the acknowledgement procedure by requesting either a first partial bitmap or a next partial bitmap (in an alternate manner). This follows a similar (but not the same) approach to the one used in EGPRS with the differences that channel measurements are *always* reported, and compression is *not* used:

The mobile station fills the reported bitmap under the following rules:

- If $[V(R) - V(Q)] \bmod SNS \leq \text{bitmap size}$, then $SSN = V(R)$ ² and the bitmap covers a certain number (determined by the bitmap size) of RLC data blocks having a BSN smaller than the SSN
- If $[V(R) - V(Q)] \bmod SNS > \text{bitmap size}$ and the network requested a next partial bitmap and
 - if $[PBSN - V(Q)] \bmod SNS < \text{RLC window size}$, and $[V(R) - (PBSN + 1)] \bmod SNS < \text{RLC window size}$, then a next partial bitmap is transmitted

¹ PBSN represents the Partial Bitmap Sequence Number variable stored at the receiver which helps to determine the starting sequence number (SSN) for the next partial bitmap to be transmitted (3GPP TS 44.060 [2])

² Note that in EGPRS SSN is calculated respective to V(Q) –it is relative to V(R) in GPRS. This doesn't change the results as the reported bitmap is the same anyway. It prevents the use of the EOW bit.

- If $[V(R) - V(Q)] \bmod \text{SNS} > \text{bitmap size}$ and the network requested a first partial bitmap then the mobile station transmits a first partial bitmap

The SSN of the reported bitmap is determined as follows:

- If the mobile station transmits a first partial bitmap, then $\text{SSN} = [V(Q) + \text{bitmap size}] \bmod \text{SNS}$ and the bitmap covers the RLC data blocks having a BSN within the interval $[V(Q), \text{SSN}-1]$
- If the mobile station transmits a next partial bitmap, then
 - If $[V(R) - (\text{PBSN}+1)] \bmod \text{SNS} > \text{bitmap size}$, then $\text{SSN} = \text{PBSN} + 1 + \text{bitmap size}$, and the bitmap covers the RLC data blocks having a BSN within the range $[\text{PBSN} + 1, \text{SSN} - 1]$
 - If $[V(R) - (\text{PBSN} + 1)] \bmod \text{SNS} \leq \text{bitmap size}$, then $\text{SSN} = V(R)$ and the bitmap covers a certain amount (determined by the bitmap size) of RLC data blocks having a BSN smaller than the SSN

The BOW bit is set '0' when a next partial bitmap is transmitted and to '1' otherwise.

Complexity-wise, the network controlled partial bitmap approach is not identical to EGPRS solution as measurements are always included –i.e. polling is not the same although the network must decide for either first or next partial bitmap, RRBP is not used. Both approaches (2 and NC) are comparable in term of additional complexity to the system.

4. SIMULATION RESULTS

4.1 Window size equals bitmap size

These first simulations show how the window size affects the RLC performance. For this purpose, it is assumed that the whole RLC window fits within the acknowledgement message. The results are available in appendix on figures 1, 2, 3 and 4 for 2, 4, 6 and 8 timeslots respectively, with a RTD (round-trip delay) of 200ms. The same simulations were performed with a RTD of 120ms, and are presented on figures 5 to 8.

It can be seen that too low, the RLC window size may restrict the throughput considerably. The higher the mobile station's multislot capability and the longer the round-trip delay on the RLC layer, the larger RLC window size is needed.

4.2 Partial bitmap approach 1

In this case, the bitmap size was limited to the value of 116 which corresponds to the estimated space in the RLC acknowledgement message. Then the RLC window size was varied from 116 to 512 for 2 to 8 timeslots. The results are shown on figures 9 to 12, respectively.

Based on the results the RLC window size of 256 or 512 gives clearly higher RLC link throughput than the RLC window size of 116 even if the bitmap size was not higher than 116. The partial bitmap approach 1 is not, however, able to give as high a throughput as the idealistic reference case where the bitmap size equals to the RLC window size of 512.

4.3 Partial bitmap approach 2

The same simulations as in 4.2 were run using partial bitmap approach 2. The RLC link throughput was once again measured for various RLC window sizes, C/I values and multislot capabilities. The results for 2-, 4-, 6- and 8-TS mobiles are presented in figures 13, 14, 15 and 16, respectively.

It can be seen from the results that the partial bitmap approach 2 performs pretty well with large RLC window sizes even with high multislot capabilities. In other words, the partial bitmap approach 2 applied with the RLC window size of 512 gives almost as high a throughput as the idealistic reference case where the bitmap size equals to the RLC window size and has a value of 512.

4.4 Network controlled partial bitmap approach

The same simulations as in 4.2 were run using network controlled partial bitmap approach. The RLC link throughput was measured for various RLC window sizes, C/I values and multislot capabilities. The results for 2-, 4-, 6- and 8-TS mobiles are presented in figures 17, 18, 19 and 20 respectively.

Based on the results the Network Controlled Partial Bitmap Approach performs pretty well and provides almost as high a throughput as the idealistic reference case where the bitmap size equals to the RLC window size.

The performances of the various partial bitmap approaches are compared with each other on figures 21 – 24. From these figures it can be seen that the Partial Bitmap Approach 2 gives slightly higher throughput than the Network Controlled Partial Bitmap Approach and that the Partial Bitmap Approach 1 gives somewhat lower throughput than the other partial bitmap approaches especially when the mobile supports high multislot capability.

5. CONCLUSIONS

This contribution has addressed the performance of the RLC protocol layer with respect to various parameter values. Various partial bitmap approaches are presented that enable the use of an RLC window size that is larger than the size of the bitmap that can be included in the RLC acknowledgement message. The performance of these partial bitmap approaches has been studied in the simulations.

Based on the simulation results the following conclusions can be reached:

- Too low an RLC window size may restrict the throughput considerably. The higher the mobile's multislot capability and the longer the round-trip delay on the RLC layer, the larger RLC window size is needed.
- The partial bitmap approach 1 applied with the RLC window size of 256 or 512 gives clearly higher RLC link throughput than the RLC window size of 116 in a situation where the bitmap size was restricted to the value of 116.
- The partial bitmap approach 2 applied with the RLC window size of 512 and the bitmap size of 116 gives almost as high a throughput as the idealistic reference case where the bitmap size equals to the RLC window size and has a value of 512.
- The Network Controlled Partial Bitmap approach where the first and next partial bitmaps are requested in alternate manner performs well also and gives almost as high a throughput as the Partial Bitmap Approach 2.

These conclusions imply the following recommendation:

- A window size of 256 or 512 is needed, 512 bringing almost ideal results, hence 512 is preferred

- Bitmap compression is not needed. Partial bitmap approaches are sufficient, the partial bitmap approach 2 performs best and brings close to ideal results with a window of size 512.
- No extended polling is needed. A polling bit is sufficient, as is already the working assumption.

REFERENCES

- [1] 3GPP TS 45.902, FLO TR
- [2] 3GPP TS 44.060, RLC/MAC protocol

APPENDIX 1 – Available bitmap size

The format of an RLC/MAC control block when FLO is used is defined in TR 45.902.

It consists of 23 octets, of which the first 2 bits are the RLC/MAC header (uplink), the rest (182 bits) being the content.

This content is assumed to be composed of:

- 6 bits: message tyoe
- 5 bits: TFI
- n bits: Ack/Nack description
- 45 bits: channel quality report (max. assuming BEP)

Given this, this leaves 126 bits for the Ack/Nack description of which 1 bit is used for the Final Ack Indication and 8 (or more) bits are used by the SSN. This leaves 117 bits available for the reported bitmap. 116 bits was selected under the assumption that a larger SSN would be needed.

APPENDIX 2 – Simulation results

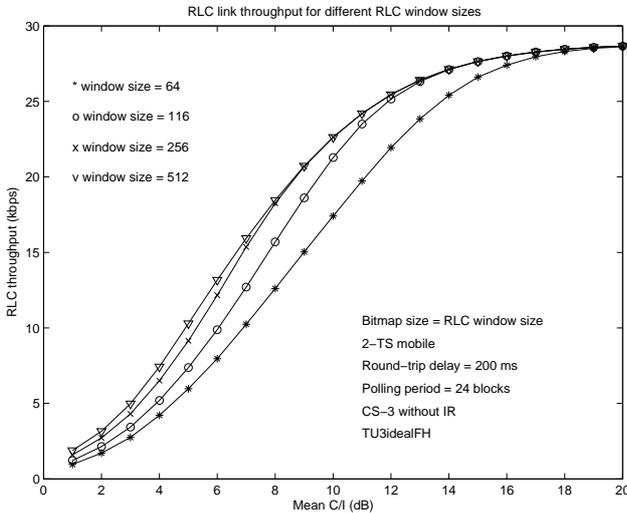


Figure 1. RLC link throughput for different RLC window sizes. Bitmap size = RLC window size. Round-trip delay = 200 ms. 2-TS mobile.

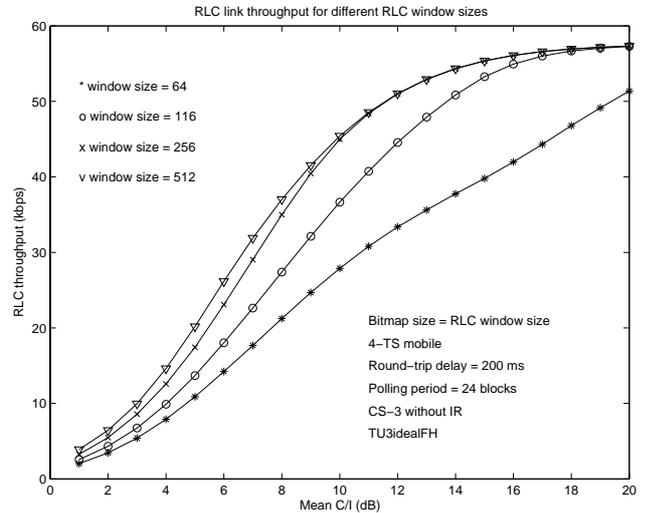


Figure 2. RLC link throughput for different RLC window sizes. Bitmap size = RLC window size. Round-trip delay = 200 ms. 4-TS mobile.

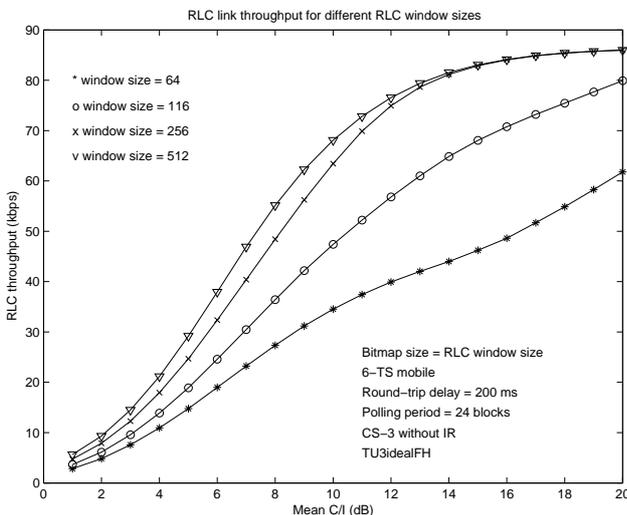


Figure 3. RLC link throughput for different RLC window sizes. Bitmap size = RLC window size. Round-trip delay = 200 ms. 6-TS mobile.

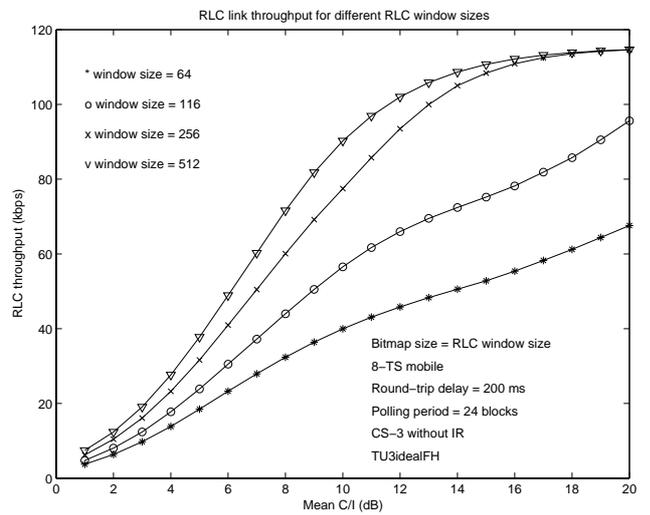


Figure 4. RLC link throughput for different RLC window sizes. Bitmap size = RLC window size. Round-trip delay = 200 ms. 8-TS mobile.

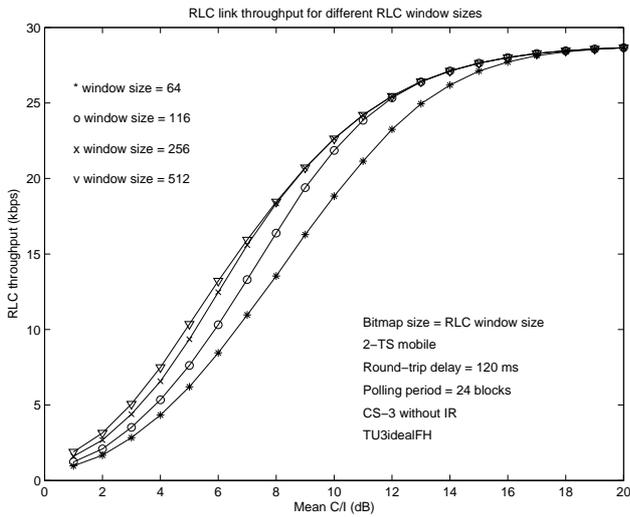


Figure 5. RLC link throughput for different RLC window sizes. Bitmap size = RLC window size. Round-trip delay = 120 ms. 2-TS mobile.

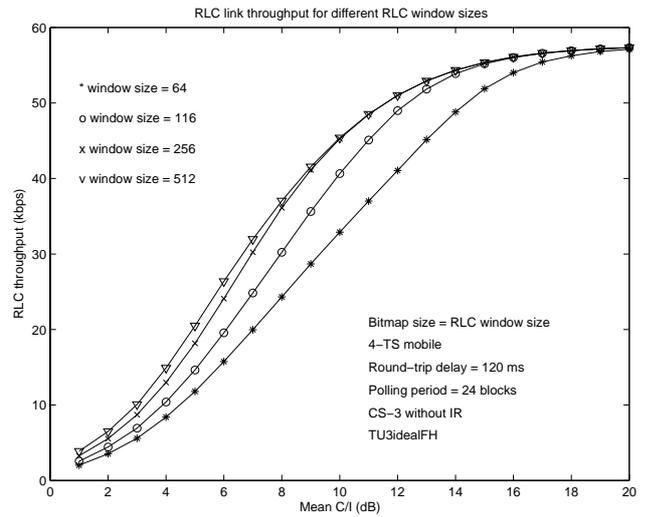


Figure 6. RLC link throughput for different RLC window sizes. Bitmap size = RLC window size. Round-trip delay = 120 ms. 4-TS mobile.

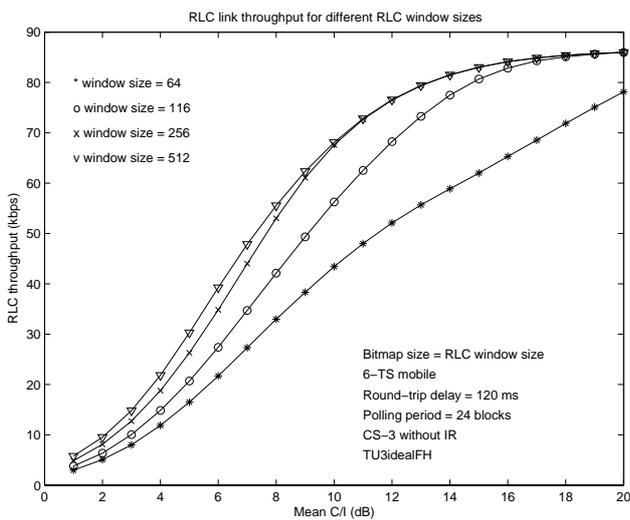


Figure 7. RLC link throughput for different RLC window sizes. Bitmap size = RLC window size. Round-trip delay = 120 ms. 6-TS mobile.

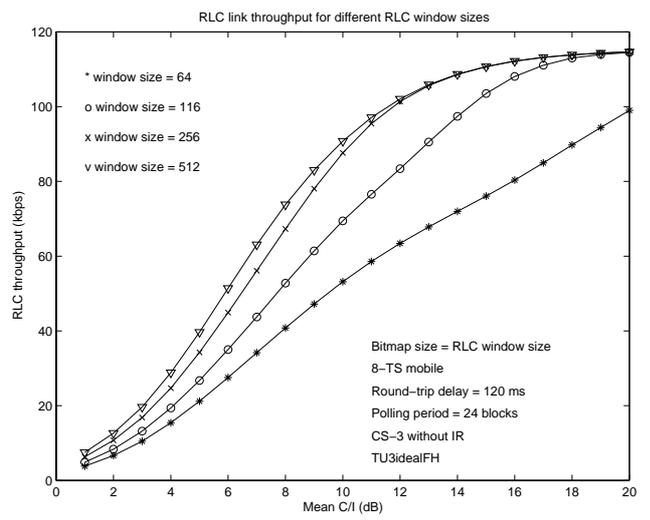


Figure 8. RLC link throughput for different RLC window sizes. Bitmap size = RLC window size. Round-trip delay = 120 ms. 8-TS mobile.

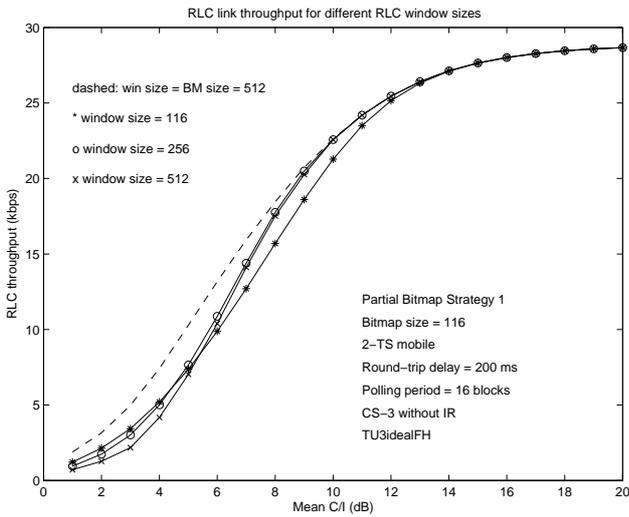


Figure 9. RLC link throughput for different RLC window sizes. Partial bitmap approach 1 applied with bitmap size of 116. Round-trip delay = 200 ms. 2-TS mobile.

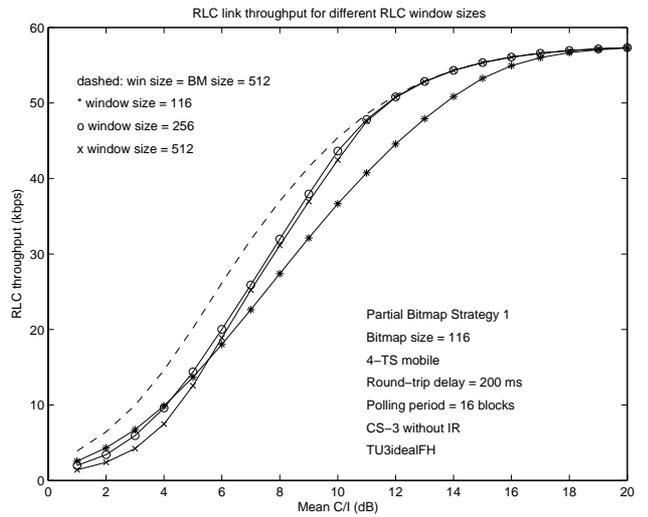


Figure 10. RLC link throughput for different RLC window sizes. Partial bitmap approach 1 applied with bitmap size of 116. Round-trip delay = 200 ms. 4-TS mobile.

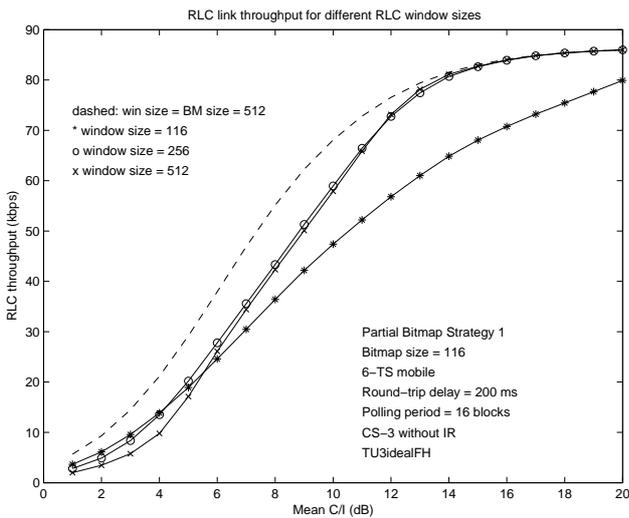


Figure 11. RLC link throughput for different RLC window sizes. Partial bitmap approach 1 applied with bitmap size of 116. Round-trip delay = 200 ms. 6-TS mobile.

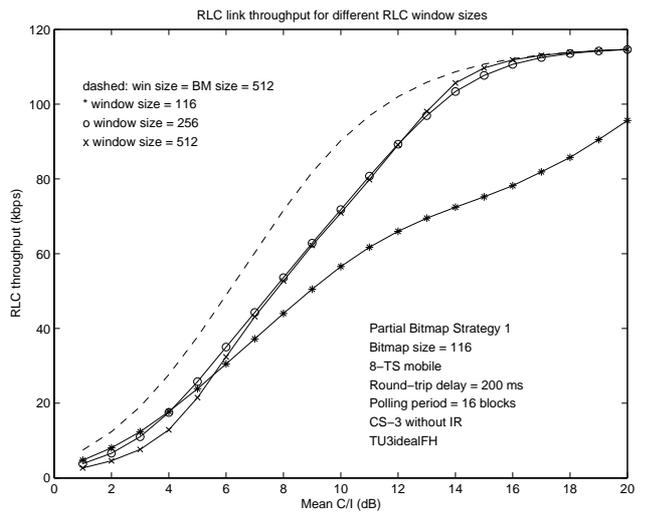


Figure 12. RLC link throughput for different RLC window sizes. Partial bitmap approach 1 applied with bitmap size of 116. Round-trip delay = 200 ms. 8-TS mobile.

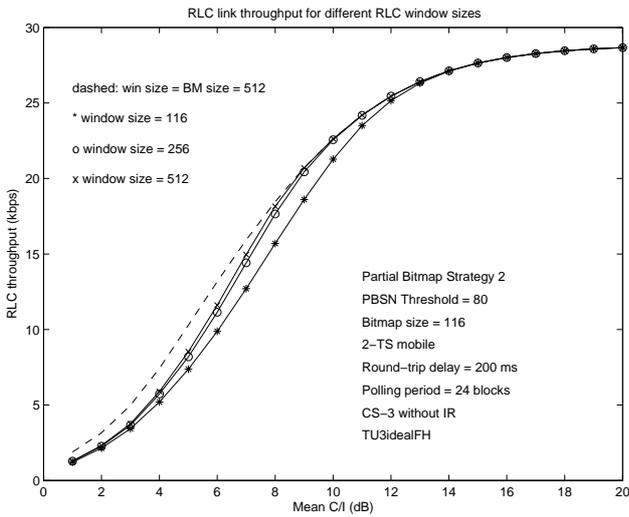


Figure 13. RLC link throughput for different RLC window sizes. Partial bitmap approach 2 applied with bitmap size of 116. Round-trip delay = 200 ms. 2-TS mobile.

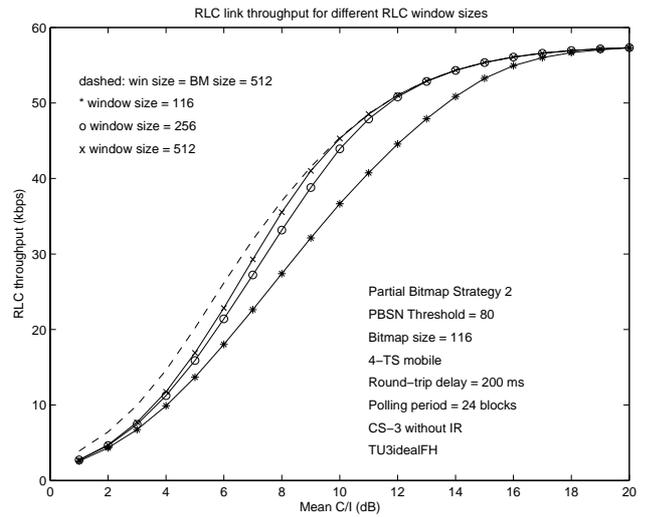


Figure 14. RLC link throughput for different RLC window sizes. Partial bitmap approach 2 applied with bitmap size of 116. Round-trip delay = 200 ms. 4-TS mobile.

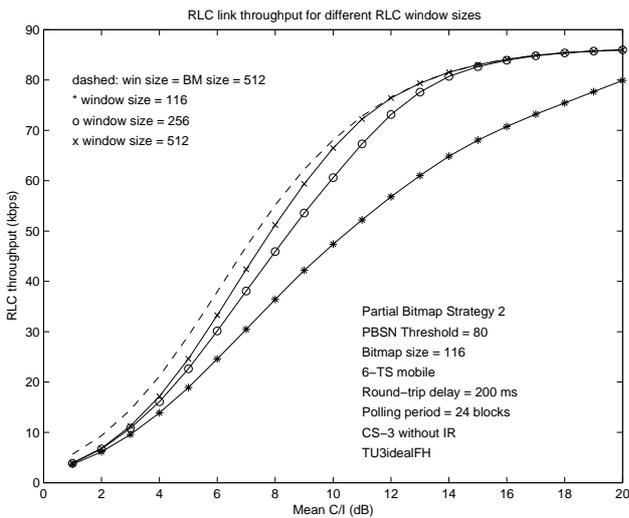


Figure 15. RLC link throughput for different RLC window sizes. Partial bitmap approach 2 applied with bitmap size of 116. Round-trip delay = 200 ms. 6-TS mobile.

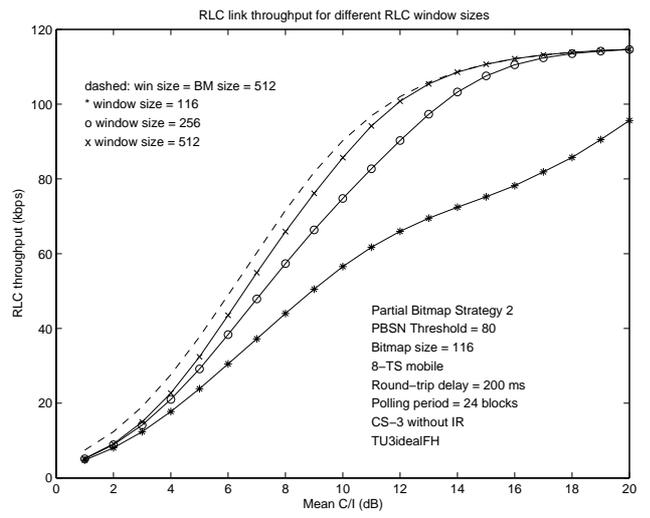


Figure 16. RLC link throughput for different RLC window sizes. Partial bitmap approach 2 applied with bitmap size of 116. Round-trip delay = 200 ms. 8-TS mobile.

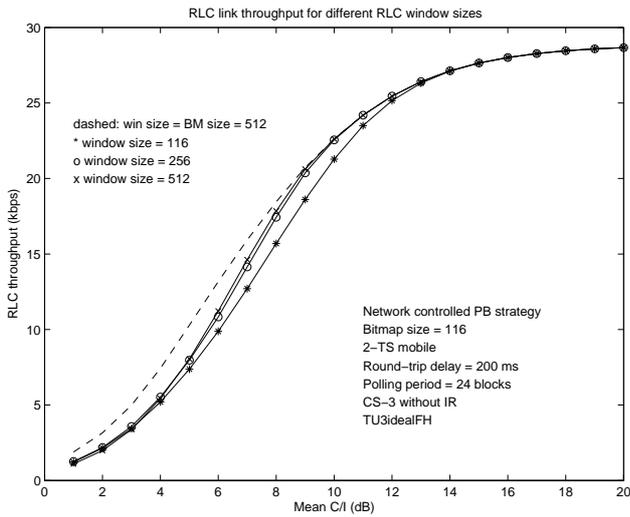


Figure 17. RLC link throughput for different RLC window sizes. Network Controlled Partial Bitmap Approach applied with bitmap size of 116. Round-trip delay = 200 ms. 2-TS mobile.

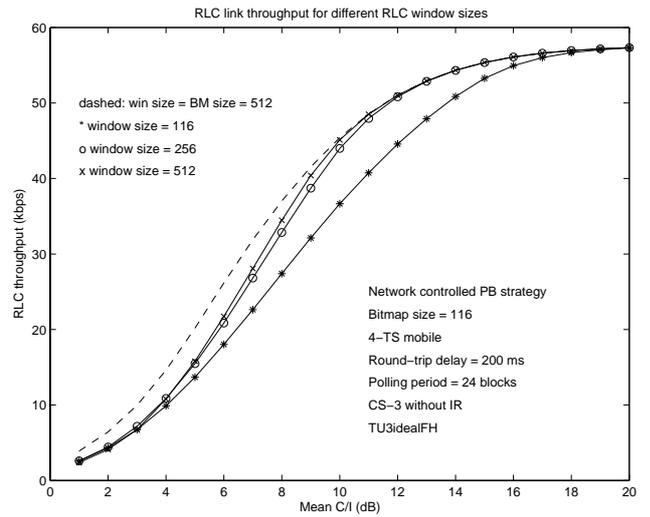


Figure 18. RLC link throughput for different RLC window sizes. Network Controlled Partial Bitmap Approach applied with bitmap size of 116. Round-trip delay = 200 ms. 4-TS mobile.

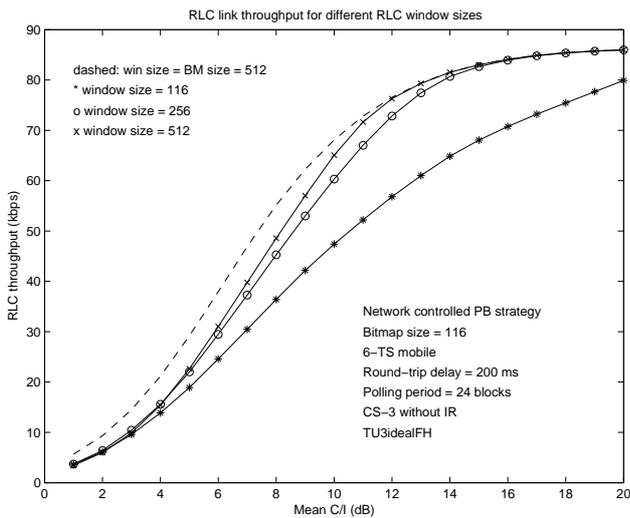


Figure 19. RLC link throughput for different RLC window sizes. Network Controlled Partial Bitmap Approach applied with bitmap size of 116. Round-trip delay = 200 ms. 6-TS mobile.

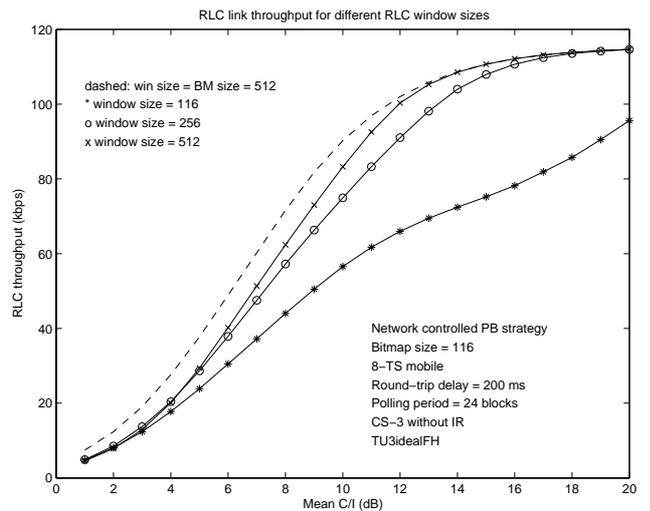


Figure 20. RLC link throughput for different RLC window sizes. Network Controlled Partial Bitmap Approach applied with bitmap size of 116. Round-trip delay = 200 ms. 8-TS mobile.

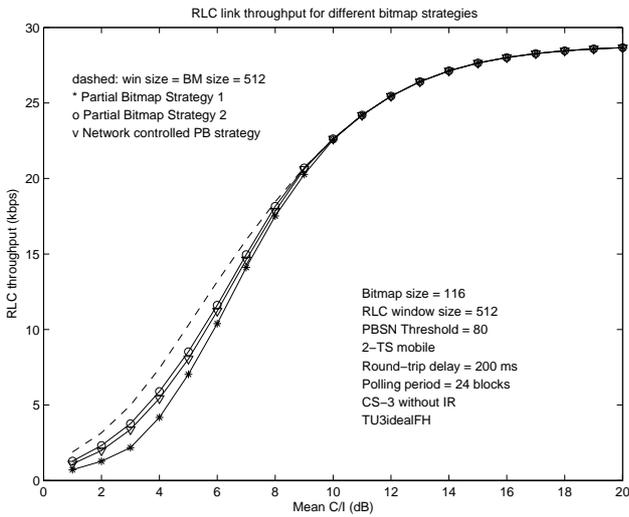


Figure 21. RLC link throughput for different partial bitmap approaches. RLC window size = 512. Round-trip delay = 200 ms. 2-TS mobile.

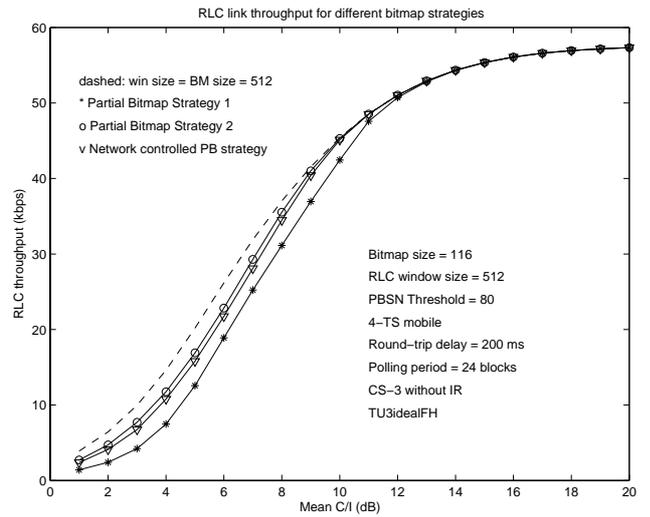


Figure 22. RLC link throughput for different partial bitmap approaches. RLC window size = 512. Round-trip delay = 200 ms. 4-TS mobile.

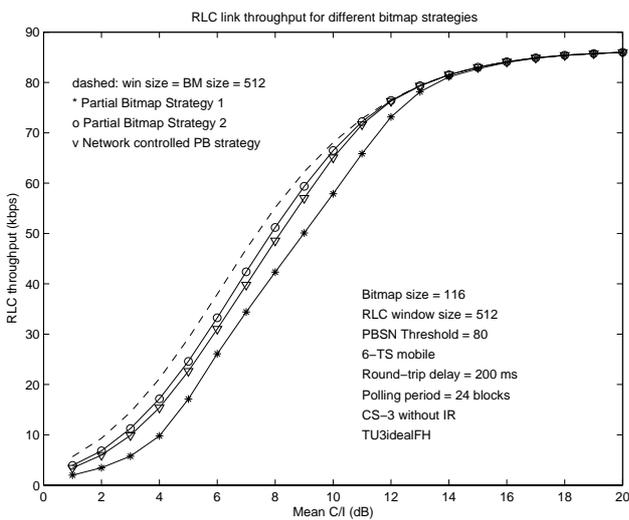


Figure 23. RLC link throughput for different partial bitmap approaches. RLC window size = 512. Round-trip delay = 200 ms. 6-TS mobile.

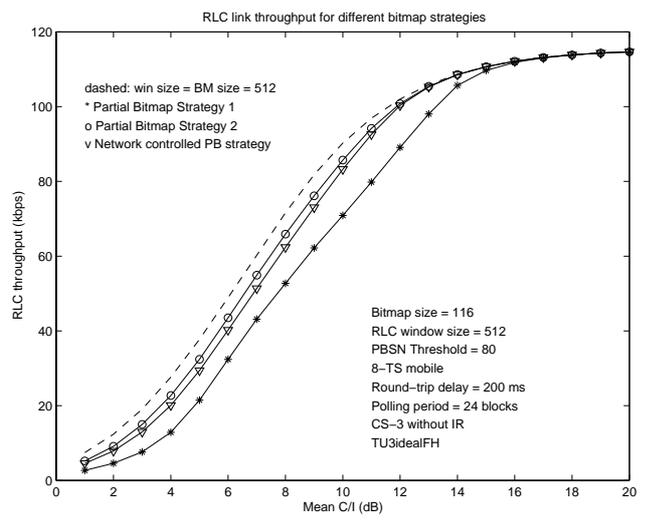


Figure 24. RLC link throughput for different partial bitmap approaches. RLC window size = 512. Round-trip delay = 200 ms. 8-TS mobile.