OsmoBTS - Feature #4795
Uplink Repeated SACCH Support

10/09/2020 04:58 PM - laforge

<table>
<thead>
<tr>
<th>Status:</th>
<th>Resolved</th>
</tr>
</thead>
<tbody>
<tr>
<td>Priority:</td>
<td>Low</td>
</tr>
<tr>
<td>Assignee:</td>
<td>dexter</td>
</tr>
<tr>
<td>Category:</td>
<td></td>
</tr>
<tr>
<td>Target version:</td>
<td>3GPP TS 44.006 Section 11.3</td>
</tr>
<tr>
<td>Spec Reference:</td>
<td>3GPP TS 44.006 Section 11.3</td>
</tr>
</tbody>
</table>

Description

In 3GPP Rel-7 (?) of GERAN, the concept of "repeated SACCH" was introduced.

The rationale for SACCH improvement can be found in 3GPP TDoc GP-042668 Section 1/2 (even though if later sections have not been implemented as suggested there): Particularly with AMR as a voice codec, the voice quality performance is better than that of control channels (and estimated 5dB).

So in the end, even though the voice channel would still be acceptable, calls fail due to signaling failure, both on SACCH and FACCH.

Uplink Repeated SACCH support basically replaces measurement reports on uplink SACCH (or even pending SAPI3 frames) with re-transmissions of SAPI0 frames. Due to some related logic (and sigaling in the TS 44.004 header), the BTS can then even combine the bursts from multiple transmission to decode the SACCH block.

Related issues:

Related to OsmoBTS - Feature #4794: Downlink Repeated SACCH support

Associated revisions

Revision 7c87612b - 11/28/2020 09:35 PM - dexter
l1sap: add repeated uplink SACCH

3GPP TS 44.006, section 11 describes a method how the uplink SACCH transmission can be repeated to increase transmission reliability.

Change-Id: I7e4cc33cc010866e41e9b594351a7f7bf93e08ac
Related: OS#4795, SYS#5114

Revision 7ce4fa43 - 12/12/2021 04:03 PM - fixeria
osmo-bts-trx: fix a memleak in trx_sched_set_lchan() 

An additional burst buffer is allocated in rx_data_in() for Uplink SACCH repetition, but unlike the main buffer it never gets free().

Change-Id: I93310e7aed91a49c0511e2ed30e00044061795a
Fixes: I7e4cc33cc010866e41e9b594351a7f7bf93e08ac
Related: OS#4795, SYS#5114

History

#1 - 10/09/2020 05:00 PM - laforge
- Related to Feature #4794: Downlink Repeated SACCH support added

#2 - 10/12/2020 05:04 PM - laforge

#3 - 10/16/2020 09:00 AM - laforge

See also: Chapter 7.2 of "GSM/EDGE Evolution and Performance".

#4 - 11/11/2020 05:37 PM - dexter
I have started the implementation of repeated uplink SACCH. In my experiments I can see that I am able to turn the repetition at the MS side on and off by setting the SRO bit in the L1 SACCH header of the SACCH block I send. So this part works so far.

When it comes to try the decoding of the current SACCH block by also using the bits from the previous SACCH block I am not entirely sure what to do. The best method to combine the previous with the current transmission seems to be to take the soft bits and just add them. One could also try to combine the bursts separately, given that the SACCH bursts are very far apart from each other this even would even make sense.

However, the most interesting question at the moment is how do we decide if we start/stop the SACCH (and maybe also FACCH) repetition. I think the SACCH repetition is mostly needed when the MS is approaching the outer perimeter of the cell. The SACCH contains the measurement reports. And even if we only get half of them when the repetition is on this is still better than nothing. I think this is the case we should design this for. I would use the the bit errors as a criterion. SACCH is coded as 1/2 viterbi. Maybe turing on the repetition when 1/3 of the received bits is wrong makes sense.

I have already submitted the patch to gerrit, but it is still work in progress: https://gerrit.osmocom.org/c/osmo-bts/+/21185

WIP: l1sap: add repeated uplink SACCH

Hoernchen, any ideas on how to achieve this best?

I did some experiments and I can see bad SACCH blocks that get decoded by combining them with the previous SACCH block by just adding the two blocks. I also think that looking at the BER is a good criterion to turn on SACCH repetition. I have implemented it as a hysteresis with hardcoded values. I have oriented myself towards the ranges given GSM 05.08, section 8.2.4. So at about an RXQUAL level of 4 SACCH repetition is turned on and if RXQUAL reaches a better than 3 it is turned off again. The levels are hardcoded at the moment, but they could be configurable. We still have 4 bits in the RSL_IE_OSMO_REP_ACCH_CAP we could use for this.

See also: https://gerrit.osmocom.org/c/osmo-bts/+/21185

The patches are up for review, no open review issues at the moment.

At the moment there are no open review issues. Since there were concerns about the saturated approach when adding the softbis i moved over to an averaged support (first divide by 2, then add). The results were nearly the same.

All related patches are merged, so we can resolve this ticket now.