

## OsmoBTS - Bug #3055

### osmo-bts-trx: slow TA loop feedback

03/10/2018 10:59 AM - fixeria

<b>Status:</b> Closed	<b>Start date:</b> 03/10/2018
<b>Priority:</b> Low	<b>Due date:</b>
<b>Assignee:</b> tnt	<b>% Done:</b> 0%
<b>Category:</b> osmo-bts-trx	
<b>Target version:</b>	
<b>Spec Reference:</b>	
<b>Description</b>	
There is probably an issue of TA loop in osmo-bts-trx: if a delay between MS and BTS is changed with a big step, e.g. from 0 to 2048 symbol periods, then a new Timing Advance value would be changed too slow, i.e. step by step after each measurement period => changing ToA from 0 to 2048 will take $2048 / 256 = 8$ measurement periods to compensate the loop...	
How to reproduce?	
Use both <a href="#">FakeTRX</a> and <a href="#">trxcon</a> from <a href="#">OsmocomBB</a> . There is an auxiliary command 'FAKE_TOA', which is intended to simulate a delay between MS and BTS:	
<a href="https://osmocom.org/projects/baseband/wiki/FakeTRX">https://osmocom.org/projects/baseband/wiki/FakeTRX</a> <a href="https://git.osmocom.org/osmocom-bb/commit/?h=fixeria/trx&amp;id=5ab622d2b72ac6df9d71d42736a6a65ac76cfab5">https://git.osmocom.org/osmocom-bb/commit/?h=fixeria/trx&amp;id=5ab622d2b72ac6df9d71d42736a6a65ac76cfab5</a>	
<b>Related issues:</b>	
Related to OsmoBTS - Feature #1851: generalize power control and TA loop code	<b>New</b> <b>11/18/2016</b>
Related to OsmocomBB - Bug #2988: unexpected bit errors when using trxcon an...	<b>Resolved</b> <b>02/23/2018</b>

### History

#### #1 - 03/14/2018 01:09 PM - laforge

- Related to Feature #1851: generalize power control and TA loop code added

#### #2 - 03/17/2018 11:37 PM - laforge

- Assignee set to sysmocom

#### #3 - 03/22/2018 05:28 PM - fixeria

- Related to Bug #2988: unexpected bit errors when using trxcon and osmo-bts-trx added

#### #4 - 06/06/2018 09:52 PM - laforge

We now have a wiki page about timing advance: [Timing\\_Advance](#)

#### #5 - 10/20/2018 07:35 PM - laforge

- Assignee changed from sysmocom to tnt

#### #6 - 01/21/2019 03:36 PM - tnt

Yes, it's slow. But OTOH it can cope with at least 1 TA value every 2 second, so that's a MS moving at ~ 1000 km/h. The way it's implemented now, it's basically heavily dampened which means it should also be very stable.

An alternative is to make the step proportional to the toa difference, basically reducing the time to adapt.

Instead of doing +1 we'd do + (toa \* alpha) alpha being the damped factor. The closer alpha is, the faster it's going to react but also the more it can oscillate in case of measurement noise.

Thoughts ? Is this really an issue ?

**#7 - 01/21/2019 04:58 PM - fixeria**

- Priority changed from Normal to Low

Yes, it's slow. But OTOH it can cope with at least 1 TA value every 2 second, so that's a MS moving at ~ 1000 km/h.

Hmm, you're right. GSM TS 05.10 says: "The control loop for the timing advance shall be implemented in such a way that it will cope with MSs moving at a speed up to 500 km/h", so the current implementation is compliant.

An alternative is to make the step proportional to the toa difference, basically reducing the time to adapt.  
Thoughts ? Is this really an issue ?

Are there any other cases when the TA value rapidly changes and we need to react quickly?  
If no, feel free to close this issue. Changing priority to low for now.

**#8 - 01/22/2019 03:50 PM - laforge**

Hi Sylvain,

On Mon, Jan 21, 2019 at 03:36:46PM +0000, tnt [REDMINE] wrote:

Yes, it's slow. But OTOH it can cope with at least 1 TA value every 2 second, so that's a MS moving at ~ 1000 km/h. The way it's implemented now, it's basically heavily dampened which means it should also be very stable.

Indeed, you're making a very valid point. I don't think we need to react too quickly to large changes, because something fishy is going on in such cases.

Thoughts ? Is this really an issue ?

I wouldn't think so.

**#9 - 01/23/2019 04:12 AM - fixeria**

*- Status changed from New to Closed*

Ok, closing. Thank you Sylvain!