

OsmoBSC - Feature #1608

various handover improvements, meta-issue

02/23/2016 04:01 PM - laforge

Status:	Rejected	Start date:	02/23/2016
Priority:	High	Due date:	
Assignee:	neels	% Done:	0%
Category:	Handover		
Target version:			
Spec Reference:			

Description

This issue ended up soaking in too much complexity and now just serves as a "related" hook for more concise issues.

Related issues:

Related to OsmoMSC - Feature #1778: avoid mismatching TCH/F vs TCH/H pchan types	Resolved	07/21/2016
Related to OsmoBSC - Bug #3002: HO2: handover decision for dynamic timeslots	New	02/26/2018
Related to OsmoBSC - Feature #3480: handover from TCH/F to TCH/H to reduce load	New	08/20/2018
Related to OsmoBSC - Feature #3481: handover from TCH/H to TCH/F to reduce load	New	08/20/2018
Related to OsmoBSC - Feature #3482: handover decision: if a new call causes c...	New	08/20/2018
Related to OsmoBSC - Feature #3483: handover decision: if rxlev is ok but rxq...	New	08/20/2018
Related to OsmoBSC - Feature #3485: handover to make room for a less capable ...	New	08/20/2018
Related to OsmoBSC - Feature #3486: handover decision: congestion: figure out...	New	08/20/2018
Related to OsmoBSC - Support #3487: comprehensive documentation for Handover ...	In Progress	08/20/2018
Related to OsmoBSC - Feature #1606: hand-over for load distribution among BTSs	In Progress	02/23/2016
Related to OsmoBSC - Feature #3638: handover decision 2: load balancing acros...	New	10/09/2018
Related to OsmoBSC - Bug #3645: internal handover: when handover changes the ...	New	10/11/2018
Related to OsmoBSC - Feature #3656: inter-BSC handover outgoing: compose Cell...	New	10/15/2018

History

#1 - 10/06/2017 02:28 PM - laforge

#2 - 11/07/2017 08:26 PM - laforge

- Assignee set to neels

#3 - 11/07/2017 08:28 PM - laforge

- Priority changed from Normal to High

#4 - 11/07/2017 09:33 PM - laforge

- Priority changed from High to Urgent

#5 - 11/20/2017 12:34 PM - neels

I'm not starting on this yet, please let me know in case I should start on this before completing [#2618](#) / [#2638](#)

#6 - 11/27/2017 02:59 PM - neels

- Related to Feature #1778: avoid mismatching TCH/F vs TCH/H pchan types added

#7 - 12/10/2017 08:05 PM - laforge

- Project changed from OpenBSC to OsmoBSC

- Category deleted (libbsc)

#8 - 12/10/2017 08:31 PM - laforge

- *Category set to Handover*

#9 - 02/21/2018 12:26 AM - neels

- *% Done changed from 0 to 50*

handover_decision_2.c which was recently merged does in fact include re-assignment and handover while changing the TCH type. It also employs an AFS bias, i.e. to handover half rate to full rate to improve the quality. Will keep this issue open as a reminder to somehow test how well this works in practice.

to enable, configure osmo-bsc with

```
network
handover 1
handover algorithm 2
```

and see various handover2 * config options

#10 - 05/17/2018 11:36 AM - laforge

neels wrote:

Will keep this issue open as a reminder to somehow test how well this works in practice.

please create a separate ticket about missing automatic testing. We need proper TTCN-3 tests for all kinds of handover scenarios in the BSC, but that's not what this ticket was about originally. Also, this ticket is certainly not "New".

#11 - 05/25/2018 10:13 AM - laforge

ping?

#12 - 05/25/2018 01:21 PM - neels

- *Status changed from New to Stalled*

From my viewpoint, even though I haven't even started looking at and testing this specific aspect, the code we got "for free" from jolly's load-based HO code is already merged... Hence I ended up with "New" and "50%". From outside this must surely look insane, yes.

ok then ...

The preliminary implementation is in handover algorithm 2.

We need testing in TTCN3, possibly also osmo-gsm-tester. Generally, testing by ttcn3 seems to make most sense, since for manual tests, we need to jump through hoops to make the equipment pick the desired lchans and then make those appear switch-worthy. In ttcn3 we can simply make up exactly the physical scenario we want to test in the messages sent.

Even though we have some code, we need to first decide what **should** be happening.

The precise scenarios to change between TCH/F and TCH/H are non-trivial and currently under-specified.

Here are some short notes/questions about scenarios that need refining:

- If we set min-free-tch-f, here is a scenario to test:
A cell has min-free-tch-f = 1, i.e. we want at least one TCH/F to remain open.
A new call could come in and pick the last remaining TCH/F slot.
Once the TCH/F lchan is used, the handover decision 2 algorithm notes that we now have 0 free TCH/F but want 1 free TCH/F.
Hence it would trigger re-assignment or handover to make one TCH/F free again.
This might go to a neighbor cell (if any), or re-assign to TCH/H within the cell.
- In previous point, we'd be a bit clumsy in first assigning TCH/F, using that for a few seconds and then re-assigning, when we might have decided for TCH/H to begin with. (out of scope here?)
- Bad quality scenario: rxlev is ok, but rxqual drops below the min_rxqual threshold.
To test for this with real equipment, we could manipulate the min_rxqual threshold during a call.
In ttcn3, we could use measurement reports.
Handover decision 2 contains a switch from TCH/H to TCH/F to improve rxqual. See on_measurement_report() in handover_decision_2.c, by the comment "Bad Quality".
However, that condition seems buggy: "if (... av_rxqual > ho_get_hodec2_min_rxqual(bts->ho))" should probably be the flipped?
- To help against interference (good rxlev, low rxqual) it might even make sense to just go to a different timeslot with the same TCH kind.
Currently we explicitly avoid re-assigning to the same TCH kind though. (out of scope here?)
- Without 'min-free-slots' configured:
If a cell has TCH/H slots and TCH/F slots, and all TCH/F slots are busy, a new call being established would just go ahead and use a TCH/H slot, not needing re-assignment at all?
- If a new call is incapable of TCH/H, should that trigger another ongoing TCH/F to be re-assigned to TCH/H, to make space for the new call? (out of scope here? so far we have no handover triggered by new calls being established, rather only re-arranging already ongoing TCH lchans.)
- With dynamic timeslots [#3002](#), it could make sense to switch over from TCH/F to TCH/H and make more room.
When the "min-free-slots" for TCH/F is surpassed, that could be a trigger to switch TCH/F to TCH/H.
But at the same time that already is a trigger to ask handover to a different cell.
At which rxlevs do we favor changing to TCH/H vs. intra-BSC handover vs. inter-BSC handover?
For intra-BSC handover we could also take the cell load into account: only change to TCH/H in case the neighbor cells are already heavily loaded as well?
Trade-off by neighbor rxlev?
- If all lchans are in use, and one of them is a dynamic timeslot used as TCH/F, would we be able to take down the call, change over to TCH/H and re-assign? I guess not.

#13 - 05/25/2018 01:22 PM - neels

- Checklist item [] define scenarios added
- Checklist item [] test in ttcn3 added

#14 - 05/25/2018 01:39 PM - neels

- Related to Bug #3002: HO2: handover decision for dynamic timeslots added

#15 - 08/20/2018 11:44 AM - neels

- Related to Feature #3480: handover from TCH/F to TCH/H to reduce load added

#16 - 08/20/2018 11:46 AM - neels

- Related to Feature #3481: handover from TCH/H to TCH/F to reduce load added

#17 - 08/20/2018 11:55 AM - neels

- Related to Feature #3482: handover decision: if a new call causes congestion that would trigger immediate re-assignment, assigning the other TCH kind from the start added

#18 - 08/20/2018 11:58 AM - neels

- Related to Feature #3483: handover decision: if rxlev is ok but rxqual is bad, move from TCH/H to TCH/F added

#19 - 08/20/2018 12:18 PM - neels

- Related to Feature #3485: handover to make room for a less capable subscriber added

#20 - 08/20/2018 12:23 PM - neels

- Related to Feature #3486: handover decision: congestion: figure out the tradeoff: re-assign within same cell vs. handover to another cell added

#21 - 08/20/2018 12:23 PM - neels

- Checklist item deleted (define scenarios)
- Checklist item deleted (test in ttcn3)

Above list of scenarios is a rather large scope.
I'm splitting this up now to get a handle on it, see the Related Issues.

#22 - 08/20/2018 12:26 PM - neels

- Subject changed from hand-over between TCH/H and TCH/F during call based on load / quality to various handover improvements, meta-issue
- Description updated
- Status changed from Stalled to In Progress
- % Done changed from 50 to 0

#23 - 08/20/2018 03:00 PM - neels

- Related to Support #3487: comprehensive documentation for Handover configuration added

#24 - 08/20/2018 03:01 PM - neels

- Related to Feature #1606: hand-over for load distribution among BTSs added

#25 - 09/12/2018 01:34 PM - neels

- Priority changed from Urgent to High

This has been on "urgent" from before it was changed to a meta-issue.
Let me put it back to "high" until further notice.

#26 - 10/09/2018 02:57 AM - neels

- Related to Feature #3638: handover decision 2: load balancing across BSS added

#27 - 10/11/2018 02:09 PM - neels

- Related to Bug #3645: internal handover: when handover changes the speech codec, it should notify the MSC with BSSMAP Handover Performed added

#28 - 10/15/2018 10:10 PM - neels

- Related to Feature #3656: inter-BSC handover outgoing: compose Cell Identifier List from several ARFCN+BSIC added

#29 - 10/15/2018 10:11 PM - neels

Maybe instead of adding these issue relations I should use a "Handover" tag...?

#30 - 10/15/2018 11:30 PM - laforge

On Mon, Oct 15, 2018 at 10:11:21PM +0000, neels [REDMINE] wrote:

Maybe instead of adding these issue relations I should use a "Handover" tag...?

I would argue to use a Handover category, as categories describe work-areas / features / code-sections within a given project.

#31 - 10/16/2018 01:18 PM - neels

- Status changed from In Progress to Rejected

all the issues this was a meta-issue for now have the "Handover" category set. Closing this.