# 1 开发背景及功能要求

PDCCH聚合等级（PDCCH aggregation level）在3gpp 38.211 7.3.2.1中定义如下：



原有系统功能在调度已经成功建立RRC连接的UE的过程中，为其选择PDCCH聚合等级（PDCCH搜索空间为USS）时，使用算法如下：

步骤1：

如果时隙配比是大上行（一个子帧中DL:S:UL = 7:1:2），选择聚合等级2；

- 否则，选择聚合等级4。

步骤2：

如果时隙配比是大上行，选择聚合等级2；

- 否则，如果上次HARQ反馈为DTX，则将步骤1中选择的PDCCH聚合等级提高一个级别，即：

聚合等级1 ->聚合等级2

聚合等级2 ->聚合等级4

 聚合等级4 ->聚合等级8

聚合等级8 ->聚合等级16

组网干扰规避协调项目要求PDCCH具有一定的组网干扰协调能力。修改上述原有系统功能，使PDCCH聚合等级能够自适应调整，可在一定程度上满足组网干扰规避协调项目的需求。

# 2 解决方案

## 2.1 接口和参数改动

#### 增加以下接口：

yang模型 - module du-cell

DU-CELL :

 ......

Container SCHEDULER\_CONFIG：

......

【增加】

Container PDCCH\_AGGR\_LVL\_ADAPTIVE\_ADJUSTMENT\_CFG：

leaf PDCCH\_AGGR\_LVL\_DFLT type uint8

leaf PDCCH\_CQI\_AVG type uint8

leaf PDCCH\_CQI\_STEP\_UP type uint8

leaf PDCCH\_CQI\_STEP\_DOWN type uint8

leaf PDCCH\_AGGR\_LVL\_THRESHOLD1 type uint8

leaf PDCCH\_AGGR\_LVL\_THRESHOLD2 type uint8

leaf PDCCH\_AGGR\_LVL\_THRESHOLD4 type uint8

leaf PDCCH\_AGGR\_LVL\_THRESHOLD8 type uint8

leaf PDCCH\_AGGR\_LVL\_THRESHOLD16 type uint8

【/增加】

#### 增加以下参数：

|  |  |  |  |
| --- | --- | --- | --- |
|  name | type | value range |  description |
| rgSchCsiCb.pdcchCqi | float | 0.000000……15.000000 | Cqi for pdcch aggregation level selection - adjusted during scheduling per UE |
| rgSchCsiCb.isCqiRprtRcvd | Bool | TRUE or FALSE | TRUE: CQI report has been decoded from UE CSI reportFALSE: no CQI report decoded yet. |
| gnu\_du\_pdcch\_aggregation\_level\_threshold[0] | float | 0.000000……15.000000 | threshold for aggregation level 1 - global setting |
| gnu\_du\_pdcch\_aggregation\_level\_threshold[1] | float | 0.000000……15.000000 | threshold for aggregation level 2 - global setting |
| gnu\_du\_pdcch\_aggregation\_level\_threshold[2] | float | 0.000000……15.000000 | threshold for aggregation level 4 - global setting |
| gnu\_du\_pdcch\_aggregation\_level\_threshold[3] | float | 0.000000……15.000000 | threshold for aggregation level 8 - global setting |
| gnu\_du\_pdcch\_aggregation\_level\_threshold[4] | float | 0.000000……15.000000 | threshold for aggregation level 16 - global setting |
| gnu\_du\_pdcch\_aggregation\_level\_dflt | CmNrAggrLvl | CM\_NR\_AGGR\_LVL1…….CM\_NR\_AGGR\_LVL16 | default pdcch aggregation level - global setting |
| gnu\_du\_pdcch\_cqi\_avg | float | 0.000000……1.000000 | pdcchCqi average factor - global setting |
| gnu\_du\_pdcch\_cqi\_step\_up | float |   | value for pdcchCqi step up - global setting |
| gnu\_du\_pdcch\_cqi\_step\_down | float |   | value for pdcchCqi step down - global setting |

## 2.2 受影响的系统组件

DU 5gnrsch

DU rrm

DU oam agent

## 2.3 受影响的系统用例

### 2.3.1 DU RRM处理新增小区配置（EVT\_RRM\_CELL\_ADD\_REQUEST）请求

在配置dedicated PDCCH resource pool时：

【修改前】

如果时隙（slot）配比为大上行（DL:S:UL = 7:1:2）:

PDCCH聚合等级1的候选集数 = 0

PDCCH聚合等级2的候选集数 = 4

PDCCH聚合等级4的候选集数 = 2

PDCCH聚合等级8的候选集数 = 1

PDCCH聚合等级16的候选集数 = 0

否则

PDCCH聚合等级1的候选集数 = 0

PDCCH聚合等级2的候选集数 = 0

PDCCH聚合等级4的候选集数 = 4

PDCCH聚合等级8的候选集数 = 2

PDCCH聚合等级16的候选集数 = 1

【/修改前】

【修改后】

*(对于所有时隙（slot）配比：)*

*PDCCH聚合等级1的候选集数 = 4*

*PDCCH聚合等级2的候选集数 = 2*

*PDCCH聚合等级4的候选集数 = 2*

*PDCCH聚合等级8的候选集数 = 1*

*PDCCH聚合等级16的候选集数 = 0*

【/修改后】

### 2.3.2 UE发送的msg3（包含CCCH SDU）经PHY解码后到达DU

1. DU 5gnrsch收到CCCH SDU并处理后，向上转交给DU APP UE

2. DU APP UE 响应并处理后，发送EVT\_RRM\_UE\_ADMIT\_REQUEST给DU RRM UE

3. DU RRM UE处理EVT\_RRM\_UE\_ADMIT\_REQUEST，【新增】*将以下参数配置到UE context:*

*gnu\_du\_pdcch\_aggregation\_level\_threshold*

*gnu\_du\_pdcch\_aggregation\_level\_dflt;*

*gnu\_du\_pdcch\_cqi\_avg;*

*gnu\_du\_pdcch\_cqi\_step\_up;*

*gnu\_du\_pdcch\_cqi\_step\_down;* 【/新增】

1. DU RRM UE 处理完成EVT\_RRM\_UE\_ADMIT\_REQUEST，发送EVT\_RRM\_UE\_ADMIT\_CONFIRM给DU APP UE
2. DU APP UE 发送EVT\_F1AP\_UE\_ID\_RESPONSE给DU F1AP, DU F1AP响应并处理。
3. DU APP UE 发送EVT\_RGR\_LVL1\_CONFIG\_REQUEST给DU 5gnrsch，

DU 5gnrsch【新增】*将以下参数配置到scheduler level1 UE control block（RgSchLvl1UeCb）中：*

*pdcchAggregationLevelThreshold*

*pdcchAggregationLevelDflt*

*pdcchCqiAvg*

*pdcchCqiStepUp*

*pdcchCqiStepDown*

*同时将isCqiRprtRcvd设置为初值FALSE*【/新增】

### 2.3.3 UE发送的UCI (包含CSI report) 经PHY解码后到达DU

5gnrsch 收到 UCI indication 并解码CSI report时：

【新增】

*如果是第一次解出CQI report，即isCqiRprtRcvd = FALSE;*

*- 设置isCqiRprtRcvd 为TRUE*

*- pdcchCqi = 解出的CQI value*

*否则，按如下方法计算结果，并将结果存储为pdcchCQI*

*- pdcchCqi \* pdcchCqiAvg + 解出的CQI value \* （1-pdcchCqiAvg）*

【/新增】

5gnrsch 收到 UCI indication 并解码CSI report时：

### 2.3.4 DU调度器（5gnrsch）在一个时隙（slot）中进行调度

在本次时隙调度开始阶段，初始化下行时隙数据时：

【新增】

*- 如果当前收到的下行HARQ反馈为ACK，且已经成功解出过CQI report:*

*pdcchCQI = pdcchCQI + pdcchCQIStepUp*

*- 如果如上计算出的pdcchCQI > 15，则pdcchCQI = 15*

*- 如果当前收到的下行HARQ反馈为DTX，且已经成功解出过CQI report：*

*pdcchCQI = pdcchCQI - pdcchCQIStepDown*

*- 如果如上计算出的pdcchCQI < 0，则pdcchCQI = 0*

【/新增】

在本次时隙调度开始阶段，初始化上行时隙数据时：

【新增】

*- 如果上行HARQ仍在存在于本时隙数据中时（上行DTX）*

*pdcchCQI = pdcchCQI - pdcchCQIStepDown*

*- 如果如上计算出的pdcchCQI < 0，则pdcchCQI = 0*

【/新增】

在进行下行或者上行调度中，选择候选UE，并进行PDCCH资源分配，选择PDCCH聚合等级时：

如果UE control block 已经存在且search space为USS（UE specific search space）:

【修改前】

(参考章节1 开发背景及功能要求)

【/修改前】

【修改后】

## IMG_256

 - 当pdcchCqi >=gnu\_du\_pdcch\_aggregation\_level\_threshold[i]，且相应pdcch aggregation level 的候选集不为0时，选择该aggregation level做为选择结果。

 - 否则，i++，继续检查下一个gnu\_du\_pdcch\_aggregation\_level\_threshold[i]

如果上述选择未成功，选择gnu\_du\_pdcch\_aggregation\_level\_dflt做为选择结果。

【/修改后】

注意：CSS(common search space)的PDCCH聚合等级选择不受改动影响。

# 3 代码实现

 (可用tortoiseGit 或者文本编辑器打开 )

****

**Debug log 验证：**

**Yzmm Web配置数据：**

"PDCCH\_AGGR\_LVL\_ADAPTIVE\_ADJUSTMENT\_CFG":{

"PDCCH\_AGGR\_LVL\_DFLT":2

"PDCCH\_CQI\_AVG":7

"PDCCH\_CQI\_STEP\_UP":15

"PDCCH\_CQI\_STEP\_DOWN":12

"PDCCH\_AGGR\_LVL\_THRESHOLD1":138

"PDCCH\_AGGR\_LVL\_THRESHOLD2":118

"PDCCH\_AGGR\_LVL\_THRESHOLD4":90

"PDCCH\_AGGR\_LVL\_THRESHOLD8":0

"PDCCH\_AGGR\_LVL\_THRESHOLD16":0

}

**尚未收到CQI之前，选择default聚合等级值：**

[04-23 14:52:42.771][DEBUG]PDCCH AL selection: Choose default aggregation level 2 for SS Id %(2),cellId[1]

**收到第一个CQI=0，以此CQI为初始pdcchCqi:**

[04-23 14:52:42.795][DEBUG]rgSCHLvl1TomDecCsiCriRiLiPmiCqi first CQI decoded = 6

**以初始pdcchCqi（6）选择出pdcch聚合等级8并调度成功:**

**UL DCI:**

[04-23 14:52:42.796][DEBUG]rgSCHGenSchdUlSchedK2 success: cellId[1] ueId[17038] cellTime[345 6] pdcchTime[345 12] puschTime[345 13]

[04-23 14:52:42.796][DEBUG]rgSCHGenSchdUlSchedK2 success: genSchdId[0] puschRsrcIdx[0] numK2[8] k2Idx[0] k2[1] numUlPdcchAvail[16] UeUlUsedPdcch[0] remUlUePerTti[3] remUlUe[4]

[04-23 14:52:42.796][DEBUG]PDCCH AL PDCCH AL selection: selected: LVL8, pdcchCqi:6.000000, threSholdForLVL8:0.000000

[04-23 14:52:42.796][DEBUG]rgSCHGenSchdUlNewTx:cellId(1) SUCCESSFUL SCHED UE 17038 ,rbsReq 1,crntTime 345:6,pdcch 345:12,pusch 345:13

**DL DCI:**

[04-23 14:52:42.813][DEBUG]rgSCHRsrcDlSchedK0K1:K0 Alloc success ,cellId:1 ueId:17038 cellTime(346:19) pdschTime(347:5) pucchTime(347:7) k0Idx:0 k0:0 dlIdx:65 k1Idx:0 k1:2 remDlUePerTti:4 numDlPdcchAvail[8] csiRsCnt[0]

[04-23 14:52:42.813][DEBUG]PDCCH AL PDCCH AL selection: selected: LVL8, pdcchCqi:6.000000, threSholdForLVL8:0.000000

[04-23 14:52:42.813][DEBUG]rgSCHCmnFillHqPPucch: ue[17038] crntTime[346 19] harqId[3] pucchResIdx[0] pucchResId[14] pucchFrmt[0] harqBitSize[1] cDaiAbs[0] pdcchTime[347 5] pdschTime[347 5] hqP-pucchTime[347 7] pucchTime[347 7]

**收到HARQ NACK，pdcchCqi=6不做调整：**

[04-23 14:52:42.818][DEBUG]rgSCHDhmHqTbTrnsFail: harqId [3] tb[0] rv[0] Ack/Nack[0] pdschTime[347 5] pucch[347 7]

**触发重传，HARQ 反馈将发生在 PUCCH 347:18：**

[04-23 14:52:42.818][DEBUG]rgSCHRsrcDlSchedK0K1:K0 Alloc success ,cellId:1 ueId:17038 cellTime(347:10) pdschTime(347:16) pucchTime(347:18) k0Idx:0 k0:0 dlIdx:76 k1Idx:0 k1:2 remDlUePerTti:4 numDlPdcchAvail[8] csiRsCnt[0]

[04-23 14:52:42.818][DEBUG]PDCCH AL PDCCH AL selection: selected: LVL8, pdcchCqi:6.000000, threSholdForLVL8:0.000000

[04-23 14:52:42.818][DEBUG]rgSCHCmnAllocPucchRsrcForHARQ: pucchTime[347 18] hqBitSize[1] tmpHqPucchRes[0x7ffe50ca8150] hqPucchRes[(nil)] resSetIndx[0]

[04-23 14:52:42.818][DEBUG][GENSCH][DL]:Retx Scheduling UE 17038 cell 1 proc 3 for totalBo 0

**在PUCCH 347:18收到HARQ ack，重传成功，pdcchCqi 按设置步长15/10 = 1.5抬升，6+1.5=7.5：**

[04-23 14:52:42.823][DEBUG]rgSCHDhmPrcFdbkForTb: ueId [17038] crntTime[348 0] pdschTime[347 16] pucch[347 18], pdcchCqi+stepUp(1.500000)=7.500000

**以pdcchCqi = 7.5选择聚合等级8：**

**UL:**

[04-23 14:52:42.836][DEBUG]rgSCHGenSchdUlSchedK2 success: cellId[1] ueId[17038] cellTime[349 6] pdcchTime[349 12] puschTime[349 13]

[04-23 14:52:42.836][DEBUG]rgSCHGenSchdUlSchedK2 success: genSchdId[0] puschRsrcIdx[0] numK2[8] k2Idx[0] k2[1] numUlPdcchAvail[16] UeUlUsedPdcch[0] remUlUePerTti[3] remUlUe[4]

[04-23 14:52:42.836][DEBUG]PDCCH AL PDCCH AL selection: selected: LVL8, pdcchCqi:7.500000, threSholdForLVL8:0.000000

**DL:**

[04-23 14:52:42.853][DEBUG]rgSCHRsrcDlSchedK0K1:K0 Alloc success ,cellId:1 ueId:17038 cellTime(350:19) pdschTime(351:5) pucchTime(351:7) k0Idx:0 k0:0 dlIdx:65 k1Idx:0 k1:2 remDlUePerTti:4 numDlPdcchAvail[8] csiRsCnt[0]

[04-23 14:52:42.853][DEBUG]PDCCH AL PDCCH AL selection: selected: LVL8, pdcchCqi:7.500000, threSholdForLVL8:0.000000

**在收到HARQ ack之后，pdcchCqi 按设置步长15/10 = 1.5抬升为9：**

[04-23 14:52:42.858][DEBUG]rgSCHDhmPrcFdbkForTb: ueId [17038] crntTime[351 9] pdschTime[351 5] pucch[351 7], pdcchCqi+stepUp(1.500000)=9.000000

**收到第二个CQI=6，更新pdcchCqi=8.099999**

[04-23 14:52:42.875][DEBUG]rgSCHLvl1TomDecCsiCriRiLiPmiCqi preCqi(9.000000)\*avg(0.700000) + cqi(6)\* (1-avg (0.300000)=8.099999

[04-23 14:52:42.875][DEBUG]rgSCHLvl1TomDecCsiCriRiLiPmiCqi:cellId[1] ueId[17038] cri[0] filter cqi[6] pdcchCqi[8.099999] pdcchCqiAvg[0.700000] pmi[1] ri[1]

**在收到HARQ ack之后，pdcchCqi 按设置步长15/10 = 1.5抬升为9.599999：**

[04-23 14:52:42.893][DEBUG]rgSCHDhmPrcFdbkForTb: ueId [17038] crntTime[354 19] pdschTime[354 15] pucch[354 17], pdcchCqi+stepUp(1.500000)=9.599999

**以pdcchCqi=9.599999选择聚合等级4：**

[04-23 14:52:42.896][DEBUG]rgSCHGenSchdUlSchedK2 success: cellId[1] ueId[17038] cellTime[355 6] pdcchTime[355 12] puschTime[355 13]

[04-23 14:52:42.896][DEBUG]rgSCHGenSchdUlSchedK2 success: genSchdId[0] puschRsrcIdx[0] numK2[8] k2Idx[0] k2[1] numUlPdcchAvail[16] UeUlUsedPdcch[0] remUlUePerTti[3] remUlUe[4]

[04-23 14:52:42.896][DEBUG]PDCCH AL PDCCH AL selection: selected: LVL4, pdcchCqi:9.599999, threSholdForLVL4:9.000000

**其后某次UL调度中，选择聚合等级1：**

[04-23 14:52:43.396][DEBUG]rgSCHGenSchdUlSchedK2 success: cellId[1] ueId[17038] cellTime[405 6] pdcchTime[405 12] puschTime[405 13]

[04-23 14:52:43.396][DEBUG]rgSCHGenSchdUlSchedK2 success: genSchdId[0] puschRsrcIdx[0] numK2[8] k2Idx[0] k2[1] numUlPdcchAvail[16] UeUlUsedPdcch[0] remUlUePerTti[3] remUlUe[4]

[04-23 14:52:43.396][DEBUG]PDCCH AL PDCCH AL selection: selected: LVL1, pdcchCqi:15.000000, threSholdForLVL1:13.800000

**但是由于在PUSCH上未解出可用信号（实际上UE deletion已经触发），即DTX，pdcchCqi按下降步长减为 13.8：**

[04-23 14:52:43.401][DEBUG]RgLiTfuDatInd: other; UEid[17038] SFN-Slot[405 13] dataSize[21] ceInfo.bitMask[128] sduCnt[1]

[04-23 14:52:43.401][ERROR]DBG\_CELLID[1] Unable to get the UE CB for CRNTI[17038]

[04-23 14:52:43.402][DEBUG]rgSCHLvl1CmnUpdPdcchCqiStepDown: DTX on puxch[405 13], isPucch[0], pdcchCqi-stepDn(1.200000)=13.800000

# **4 单UE测试**

系统预设参数设置如下：

pdcchCQI设置：

gnu\_du\_pdcch\_cqi\_avg = 0.5

gnu\_du\_pdcch\_cqi\_step\_up = 1.0

gnu\_du\_pdcch\_cqi\_step\_down = 0.8

gnu\_du\_pdcch\_aggregation\_level\_dflt = CM\_NR\_AGGR\_LVL4

PDCCH候选集参数设置：

numCandidts4AL1 = RGR\_NUM\_CANDIDTS\_N4;

numCandidts4AL2 = RGR\_NUM\_CANDIDTS\_N2;

numCandidts4AL4 = RGR\_NUM\_CANDIDTS\_N2;

numCandidts4AL8 = RGR\_NUM\_CANDIDTS\_N1;

numCandidts4AL16 = RGR\_NUM\_CANDIDTS\_N0;

PDCCH聚合等级Threshold 设置：

gnu\_du\_pdcch\_aggregation\_level\_threshold[RGR\_PDCCH\_MAX\_AL] =

 {

 14.0, //AL1 threshold, max 15.0

 12.0, //AL2 threshold, max 15.0

 10.0, //AL4 threshold, max 15.0

 6.0, //AL8 threshold, max 15.0

 0.0 //AL16 threshold, max 15.0

};

## 4.1 无衰减

### 4.1.1时隙配比: 7D1S2U 下行数据带宽30M

测试结果的KPI log如下：

Cell Level Non Persistent KPI:

CELL DLBLER #DTX DLPRB DL\_Occ Success DlNewTx DlReTx UL\_Succ UlNewTx UlReTx UlDatInd ULBLER ULPRB

1 0 0 45 48001 24917 24905 12 1541 1531 10 531 0 4

PDCCH Resource Allocation Stats:

UE-ID CELL-ID BWP\_ID PDCCH\_TYP CSID SSID AGG\_LVL FAIL SUCC

17017 1 0 0 1 2 1 0 24889

17017 1 0 0 1 2 2 0 7

17017 1 0 0 1 2 4 0 20

17017 1 0 0 1 2 8 0 2

17017 1 0 1 1 2 1 0 1535

17017 1 0 1 1 2 2 0 1

17017 1 0 1 1 2 4 0 5

17017 1 0 1 1 2 8 2 0

关于PDCCH Resource Allocation Stats：

PDCCH\_TYP列表示DCI format, 0 - DL DCI, 1- UL DCI

AGG\_LVL列表示scheduler所调度的PDCCH聚合等级

FAIL列表示scheduler所调度的PDCCH聚合等级，在实际PDCCH资源分配中失败的次数

SUCC列表示scheduler所调度的PDCCH聚合等级，在实际PDCCH资源分配中成功的次数

### 4.1.2时隙配比: 2D1S7U 下行数据带宽30M 上行数据带宽100M

测试结果的KPI log如下：

Cell Level Non Persistent KPI:

CELL DLBLER #DTX DLPRB DL\_Occ Success DlNewTx DlReTx UL\_Succ UlNewTx UlReTx UlDatInd ULBLER ULPRB

1 0 0 139 18000 8444 8444 0 31763 31759 4 30062 0 257

PDCCH Resource Allocation Stats:

UE-ID CELL-ID BWP\_ID PDCCH\_TYP CSID SSID AGG\_LVL FAIL SUCC

17019 1 0 0 1 2 1 0 8444

17019 1 0 1 1 2 1 0 31763

### 4.1.3时隙配比: 2D1S7U 下行数据带宽30M 上行数据带宽200M

测试结果的KPI log如下：

Cell DL Tput(Mb) UL Tput(Mb) UL BLER

 0 33.892 126.763 / 127.424 0.52%

Cell Level Non Persistent KPI:

CELL DLBLER #DTX DLPRB DL\_Occ Success DlNewTx DlReTx UL\_Succ UlNewTx UlReTx UlDatInd ULBLER ULPRB

1 0 0 149 18000 7843 7843 0 35626 35624 2 34050 0 260

PDCCH Resource Allocation Stats:

UE-ID CELL-ID BWP\_ID PDCCH\_TYP CSID SSID AGG\_LVL FAIL SUCC

17017 1 0 0 1 2 1 0 7710

17017 1 0 0 1 2 2 811 65

17017 1 0 0 1 2 4 199 66

17017 1 0 0 1 2 8 4 2

17017 1 0 1 1 2 1 0 32255

17017 1 0 1 1 2 2 825 2494

17017 1 0 1 1 2 4 204 867

17017 1 0 1 1 2 8 14 10

### 4.1.4时隙配比: 2D1S7U 上行数据带宽200M

PDCCH候选集参数设置修改为：

聚合等级1候选集数 = N4;

聚合等级2候选集数 = N4;

聚合等级4候选集数 = N3;

聚合等级8候选集数 = N1;

聚合等级8候选集数 = N0。

测试相关log：

 Cell DL Tput(Mb) UL Tput(Mb) UL BLER

 0 0.075 102.283 / 112.592 9.16%

CELL DLBLER #DTX DLPRB DL\_Occ Success DlNewTx DlReTx UL\_Succ UlNewTx UlReTx UlDatInd ULBLER ULPRB

1 0 597 4 18000 2202 1605 597 40541 36876 3665 36464 0 239

PDCCH Resource Allocation Stats:

UE-ID CELL-ID BWP\_ID PDCCH\_TYP CSID SSID AGG\_LVL FAIL SUCC

17018 1 0 0 1 2 1 0 1986

17018 1 0 0 1 2 2 0 215

17018 1 0 0 1 2 4 0 1

17018 1 0 1 1 2 1 0 38778

17018 1 0 1 1 2 2 0 1759

17018 1 0 1 1 2 4 0 10

此次测试中，与4.2.2相比，提高了聚合等级4的候选集数（2变为3）后，出现较多DTX，同时吞吐量下降。推测候选集设置超出了UE的PDCCH盲检能力。

## 4.2 有衰减

### 4.2.1 20dB衰减 时隙配比: 7D1S2U 下行数据带宽30M 上行数据带宽10M

测试结果的KPI log如下：

Cell Level Non Persistent KPI:

CELL DLBLER #DTX DLPRB DL\_Occ Success DlNewTx DlReTx UL\_Succ UlNewTx UlReTx UlDatInd ULBLER ULPRB

1 9 0 94 48001 26278 23718 2560 1530 1530 0 619 0 4

PDCCH Resource Allocation Stats:

UE-ID CELL-ID BWP\_ID PDCCH\_TYP CSID SSID AGG\_LVL FAIL SUCC

17023 1 0 0 1 2 1 0 27334

17023 1 0 0 1 2 2 0 19

17023 1 0 0 1 2 4 0 23

17023 1 0 0 1 2 8 0 18

17023 1 0 1 1 2 1 0 3721

17023 1 0 1 1 2 2 0 1

17023 1 0 1 1 2 4 0 4

17023 1 0 1 1 2 8 4 0

### 4.2.2 24dB衰减 时隙配比: 2D1S7U 下行数据带宽30M 上行数据带宽100M

测试结果的KPI log如下：

Cell Level Non Persistent KPI:

CELL DLBLER #DTX DLPRB DL\_Occ Success DlNewTx DlReTx UL\_Succ UlNewTx UlReTx UlDatInd ULBLER ULPRB

1 9 0 235 18000 1646 1495 151 22526 20464 2062 20281 0 239

PDCCH Resource Allocation Stats:

UE-ID CELL-ID BWP\_ID PDCCH\_TYP CSID SSID AGG\_LVL FAIL SUCC

17083 1 0 0 1 2 1 0 1310

17083 1 0 0 1 2 2 606 67

17083 1 0 0 1 2 4 1477 146

17083 1 0 0 1 2 8 10511 123

17083 1 0 1 1 2 1 0 4261

17083 1 0 1 1 2 2 320 1104

17083 1 0 1 1 2 4 1238 3773

17083 1 0 1 1 2 8 22745 13388

### 4.2.2 24dB衰减 时隙配比: 2D1S7U 下行数据带宽30M 上行数据带宽200M

PDCCH候选集参数设置修改为：

聚合等级1候选集数 = N4;

聚合等级2候选集数 = N4;

聚合等级4候选集数 = N2;

聚合等级8候选集数 = N1。

聚合等级8候选集数 = N0。

 Cell DL Tput(Mb) UL Tput(Mb) UL BLER

 0 11.496 127.141 / 127.141 0.00%

CELL DLBLER #DTX DLPRB DL\_Occ Success DlNewTx DlReTx UL\_Succ UlNewTx UlReTx UlDatInd ULBLER ULPRB

1 0 0 69 18000 11707 11671 36 35352 35333 19 34075 0 260

PDCCH Resource Allocation Stats:

UE-ID CELL-ID BWP\_ID PDCCH\_TYP CSID SSID AGG\_LVL FAIL SUCC

17017 1 0 0 1 2 1 0 9612

17017 1 0 0 1 2 2 0 1086

17017 1 0 0 1 2 4 842 922

17017 1 0 0 1 2 8 388 87

17017 1 0 1 1 2 1 0 28698

17017 1 0 1 1 2 2 0 3197

17017 1 0 1 1 2 4 1466 2772

17017 1 0 1 1 2 8 490 685

此次测试中，将聚合等级2的候选集数由2提升为4，PDCCH分配失败数量减少，吞吐量基本不变，同时没有DTX发生，说明这个配置是在UE的PDCCH盲检范围以内的。

## 4.3 测试结论及存在的问题

### 测试结论：

DU scheduler在分配PDCCH资源时，已经能够根据UE上报的CQI对PDCCH聚合等级进行动态分配。当信号质量较好时，由于UE上报的CQI较高，DU scheduler在大部分时间会选择较低的PDCCH聚合等级；当信号质量较差时，由于UE上报的CQI较低，DU scheduler会比较多的选择较高的PDCCH聚合等级。

### 测试中发现的问题：

在单UE大上行（2D1S7U）+ 有信号衰减的测试中，DU scheduler根据UE所报CQI，较多的选择了PDCCH聚合等级2，4，8，但是由于这些PDCCH聚合等级所配置的PDCCH候选集数目的限制，DU scheduler在选择这些PDCCH聚合等级以后，在PDCCH资源分配中造成较多的失败次数。

例如：测试中PDCCH聚合等级2对应的候选集数量配置为2，在某两次 slot调度中，DU scheduler在已经为分别这两个slot的PDCCH DCI format0找到了可用的K0和K2，它们的K0指向同一DL slot，并在这一DL slot上总共为同一UE进行了2次PDCCH分配，那么在随后的第三个slot调度中，仍再一次尝试在同一DL slot上分配PDCCH，根据3gpp 38.213 10.1及代码实现，将无法找到可用的PDCCH资源。

对于同一UE，是否可以将PDCCH聚合等级对应的候选集数量与某DL slot上该UE的分配次数进行比对检查，以减少这样的失败的PDCCH资源分配次数，提高系统效率？

对于多个UE，是否可以找到快速检查办法，当PDCCH资源已经耗尽时，不再调度无法获得PDCCH资源的UE，以提高调度效率？

另外，当前代码实现中，对PDCCH CQI进行抬升时，暂未考虑PUSCH CRC OK的情况 ——是否有必要在PUSCH CRC OK时也进行PDCCH CQI值的抬升？