B5G_UE - 错误 #3835

MSG5状态后pdcch漏检

2025-07-30 09:54 - 张 倩

状态:	已解决	**开始日期:**	2025-07-30
优先级:	普通	**计划完成日期:**	
指派给:	张 倩	**% 完成:**	0%
类别:		**预期时间:**	0.00 小时
目标版本:		**耗时:**	0.00 小时

描述

MSG5状态后pdcch漏检

历史记录

#1 - 2025-07-30 11:04 - 张 倩

- 文件 0210bea4-609f-4d10-8ba4-5724da654f2d.jpeg 已添加

- 文件 4dc021c3-fd07-4871-a93c-1d9e6d3478eb.jpeg 已添加

- 文件 ce91b795-9d33-4659-a196-a13409644e1d.jpeg 已添加

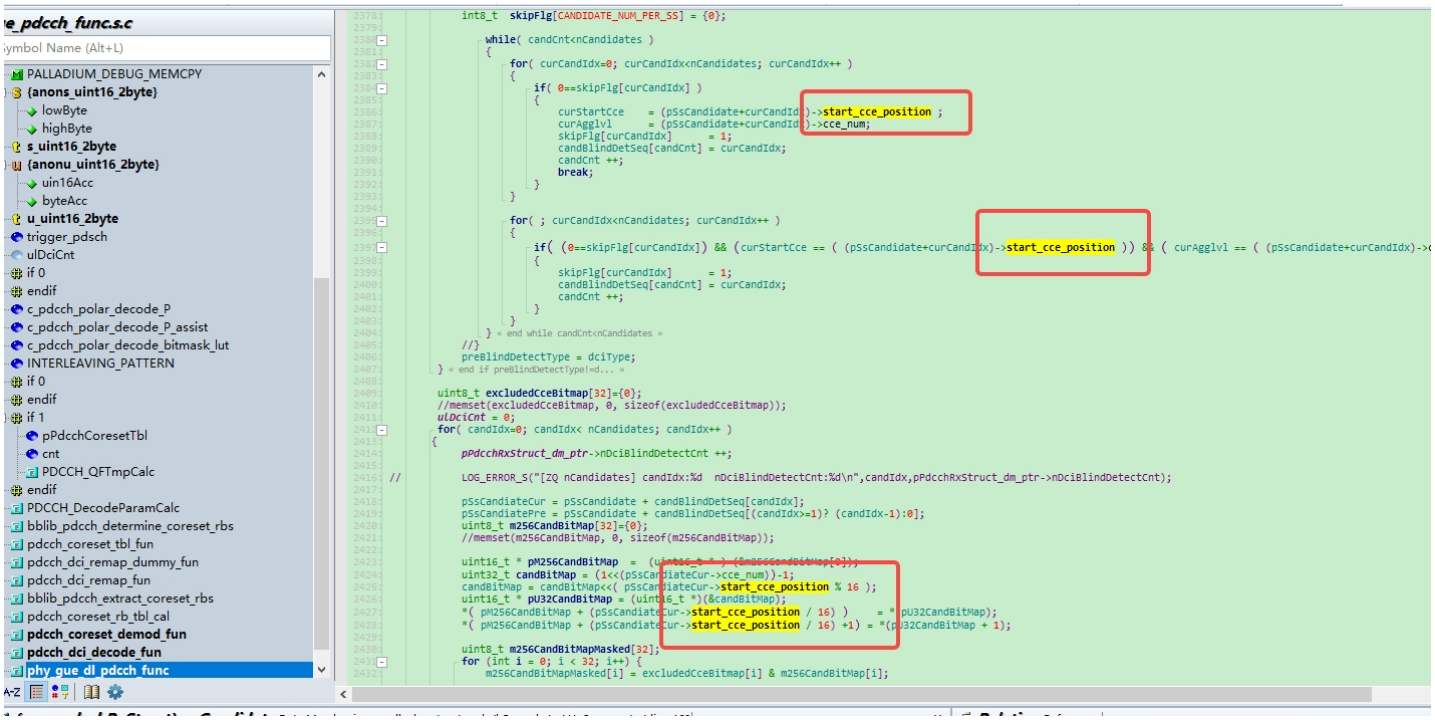
- 文件 df93a32b-acfd-41ae-ba40-cb66e16d0893.jpeg 已添加

msg5状态之前 startcce使用的是红框参数start_cce_position
mgs5状态之后 startcce使用的是绿框参数start_cce_position1[]

```
typedef struct searchSpaceCandidate
{
    bool  valid_flag ; // 0- unvalid , 1- valid
    uint8_t    start_symbol_idx ; // 起始符号
    uint8_t    start_cce_position;
    uint8_t    start_cce_position1[20];//250414 coreset 1
    uint8_t    cce_num ;
    uint8_t    dci_format;
    uint16_t   dci_size;
    uint32_t   rnti ;
} SearchSpaceCandidate;
```

原代码中只在pdcch_dci_remap_fun函数中在msg5之后将start_cce_position替换成了start_cce_position1[]，这个函数外面其他startcce用的仍然是start_cce_position，但是msg5之后start_cce_position没赋值，默认是0，导致pdcch漏检了一些候选集；

```
static int16_t pdcch_dci_remap_fun(uint16_t coresetIdx, uint16_t ssIdx, uint16_t candIdx,
        float *sigma2Ret, float * sigpowRet, float * snrDbRet )
{
    //StePCONFIG_PARAMS pConfigParams = phy_gue_main_get_ctx();
    //uint32_t        nCtxNum      = get_dl_sf_ctx(nSlotIdx);
#if 0
    PHYStateDLRx *psPHYStateRx = phy_gue_dl_get_ctx();
#endif
    struct timespec Starttime;
    struct timespec Endtime;
   // clock_gettime(CLOCK_REALTIME, &Starttime);

    //pdcchRxStruct * pPdcchRxStruct_dm_ptr = &d_psPHYStateRx->sPdcchRxStruct;

    SteCoreSet * pCoreset = *( (SteCoreSet * *)pPdcchRxStruct_dm_ptr->pCsTbl + coresetIdx );

    SearchSpaceCandidate * pSsCandidate = &pPdcchRxStruct_dm_ptr->ssCandidate[coresetIdx][ssIdx][candIdx];
#if 0
    int32_t  nRxAntNum     = LOAD_EX_W(&psPHYStateRx->nRxAntNum);
#endif
#if 1
    int32_t  nRxAntNum   = 1;
#endif
    uint16_t * pCsRbIdxTbl = (uint16_t *)pPdcchRxStruct_dm_ptr->pCsRbIdx[coresetIdx];
    uint16_t nCsRbNum      = pPdcchRxStruct_dm_ptr->nCsRbNum[coresetIdx];

    //
    struct bblib_pdcch_determine_phy_rbs_request  pdcchRbsRequest;
    struct bblib_pdcch_determine_phy_rbs_response pdcchRbsResponse;
    uint8_t slot = get_rx_nr_slot();

    pdcchRbsRequest.cce_reg_mapping_type_cs = pCoreset->cce2RegMapTyp;
    pdcchRbsRequest.interleave_size_R_cs    = pCoreset->intrlvrRows;
    pdcchRbsRequest.n_duration_cs           = pCoreset->duration;
    pdcchRbsRequest.n_reg_dundle_size_cs    = pCoreset->regBndlSz;
    pdcchRbsRequest.n_shift_cs              = pCoreset->shiftIndex;
    pdcchRbsRequest.start_cce_idx_pdcch     = pSsCandidate->start_cce_position;
    pdcchRbsRequest.n_agg_level_pdcch       = pSsCandidate->cce_num;
    pdcchRbsRequest.rb_idx_bwp_cs           = pCsRbIdxTbl;
    pdcchRbsRequest.n_rb_cs                 = nCsRbNum;

    pdcchRbsResponse.rb_idx_bwp_pdcch =  (uint16_t *)(pCsRbIdxTbl + RUP64B(MAX_NUM_OF_PRB_IN_FULL_BAND) );
    pdcchRbsResponse.rb_bitmap_pdcch  =  (uint8_t *)(pCsRbIdxTbl + RUP64B(MAX_NUM_OF_PRB_IN_FULL_BAND)*2 );
    uint16_t ueState;
    ueState=LOAD_EX_W(&ueCfg->ueState);
    if(ueState >= UE_MSG5_SS_CFG_SUCC )
    {
      pdcchRbsRequest.start_cce_idx_pdcch   = pSsCandidate->start_cce_position1[g_pdcch_freq_data_hdr_dm_ptr->slot];

    }
    uint8_t ueType;LOAD_EX_B(&ueCfg->ueType);
```



修改点：msg5状态之后，在pdcch任务入口处就把start_cce_position1[]赋值给start_cce_position

```
case UE_MSG5_SS_CFG_SUCC:
    /*
      fix bug: 特殊场景考虑如下描述
      当UE处于UE_MSG5_SS_CFG_SUCC这个状态的时候,表示msg4 DCI/pdsch都解对了且收到DU的msg4配置了
      UE反馈了MSG4 ACK但是基站可能没有解对,然后会发起msg4的重传,所以UE这个状态需要盲检msg4重传的DCI1_0
    */

    #if 1

    pPdcchRxStruct_dm_ptr->blindDetect[BLIND_DETECT_DCI_MSG5].candidateNum=LOAD_EX_B(&ueCfg->blindDetect[BLIND_DETECT_DCI_MSG5].candidateNum);
    pPdcchRxStruct_dm_ptr->blindDetect[BLIND_DETECT_DCI_MSG5].coresetIdx=LOAD_EX_B(&ueCfg->blindDetect[BLIND_DETECT_DCI_MSG5].coresetIdx);
    pPdcchRxStruct_dm_ptr->blindDetect[BLIND_DETECT_DCI_MSG5].searchSpaceIdx=LOAD_EX_B(&ueCfg->blindDetect[BLIND_DETECT_DCI_MSG5].searchSpaceIdx);
    pPdcchRxStruct_dm_ptr->blindDetect[BLIND_DETECT_DCI_MSG5].startFlag=LOAD_EX_B(&ueCfg->blindDetect[BLIND_DETECT_DCI_MSG5].startFlag);
    #endif
    uint32_t cRnti;
    cRnti= LOAD_EX_W(&ueCfg->cRnti);
    //LOG_ERROR_S("UE_MSG5_SS_CFG_SUCC cRnti: %d \n",cRnti);
    for( int8_t csIdx=0; csIdx<MAX_CORESET_NUM; csIdx++)
        for( int8_t ssIdx=0; ssIdx<MAX_SEARCH_SPACE_NUM; ssIdx++)
            for( int8_t candIdx=0; candIdx<CANDIDATE_NUM_PER_SS; candIdx++ )
            {
                pPdcchRxStruct_dm_ptr->ssCandidate[csIdx][ssIdx][candIdx].rnti = cRnti;
                pPdcchRxStruct_dm_ptr->ssCandidate[csIdx][ssIdx][candIdx].start_cce_position = pPdcchRxStruct_dm_ptr->ssCandidate[csIdx][ssIdx][candIdx].start_cce_position1[g_pdcch_freq_data_hdr_dm_ptr->slot];
            }

    break;

case UE_MSG5_DCI_PARSE_SUCC: // 当处于这个状态的时候,有可能需要去盲检msg5重传的DCI0_1
    #if 0
    ape_csu_dma_1D_G2L_ch2ch3_transfer((uint64_t)&ueCfg->blindDetect[BLIND_DETECT_DCI_MSG5],
                        (uint64_t)DM_TO_CSU_ADDR(&pPdcchRxStruct_dm_ptr->blindDetect[BLIND_DETECT_DCI_MSG5]),
                        sizeof(BlindDetectInfo),
                        DMA_TAG_G2L,
                        1);
    #endif
    #if 1
    pPdcchRxStruct_dm_ptr->blindDetect[BLIND_DETECT_DCI_MSG5].candidateNum=LOAD_EX_B(&ueCfg->blindDetect[BLIND_DETECT_DCI_MSG5].candidateNum);
    pPdcchRxStruct_dm_ptr->blindDetect[BLIND_DETECT_DCI_MSG5].coresetIdx=LOAD_EX_B(&ueCfg->blindDetect[BLIND_DETECT_DCI_MSG5].coresetIdx);
    pPdcchRxStruct_dm_ptr->blindDetect[BLIND_DETECT_DCI_MSG5].searchSpaceIdx=LOAD_EX_B(&ueCfg->blindDetect[BLIND_DETECT_DCI_MSG5].searchSpaceIdx);
    pPdcchRxStruct_dm_ptr->blindDetect[BLIND_DETECT_DCI_MSG5].startFlag=LOAD_EX_B(&ueCfg->blindDetect[BLIND_DETECT_DCI_MSG5].startFlag);
    for( int8_t csIdx=0; csIdx<MAX_CORESET_NUM; csIdx++)
        for( int8_t ssIdx=0; ssIdx<MAX_SEARCH_SPACE_NUM; ssIdx++)
            for( int8_t candIdx=0; candIdx<CANDIDATE_NUM_PER_SS; candIdx++ )
            {
                pPdcchRxStruct_dm_ptr->ssCandidate[csIdx][ssIdx][candIdx].start_cce_position = pPdcchRxStruct_dm_ptr->ssCandidate[csIdx][ssIdx][candIdx].start_cce_position1[g_pdcch_freq_data_hdr_dm_ptr->slot];
            }

    #endif
case UE_ACTIVE:
```

#2 - 2025-07-30 11:04 - 张 倩

- 状态 从 新建 变更为 已解决

文件

0210bea4-609f-4d10-8ba4-5724da654f2d.jpeg	49.4 KB	2025-07-30	张 倩
4dc021c3-fd07-4871-a93c-1d9e6d3478eb.jpeg	272 KB	2025-07-30	张 倩
ce91b795-9d33-4659-a196-a13409644e1d.jpeg	388 KB	2025-07-30	张 倩
df93a32b-acfd-41ae-ba40-cb66e16d0893.jpeg	337 KB	2025-07-30	张 倩