



# 宇阳的 板形模型 私房菜

热轧模型组

# Table of Contents

Introduction	1.1
GSM总览	1.2
SSU板形设定模型	1.3
板形模型初始化	1.3.1
包络线计算	1.3.2
凸度分配计算	1.3.3
板形评估	1.3.4
板形物理模型	1.3.5
偏导数参数或增益	1.3.6
凸度分配	1.4
辊系凸度计算	1.4.1
参数和日志	2.1
SSU参数日志验算	2.1.1
SSU参数日志FAQ	2.1.2
CFG模型参数梳理	2.1.3
CTool参数梳理	2.1.4
板形模型实战	2.2
目标凸度变化影响板形模型设定	2.2.1

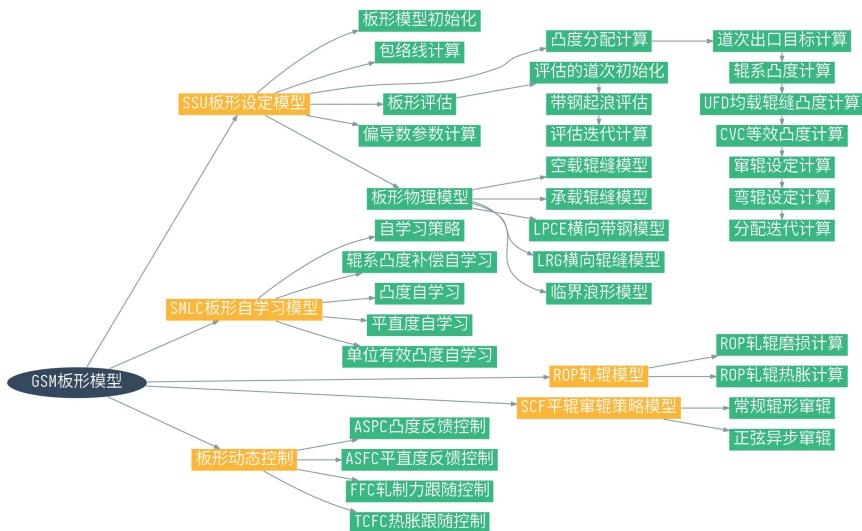
# 宇阳的板形模型私房菜（板形模型消化）

# 板形模型总览

GSM板形模型可以分为以下几个大方面：

- SSU板形设定模型
- SMLC板形自学习模型
- ROP轧辊模型
- SCF平辊窜辊策略模型
- 板形动态控制

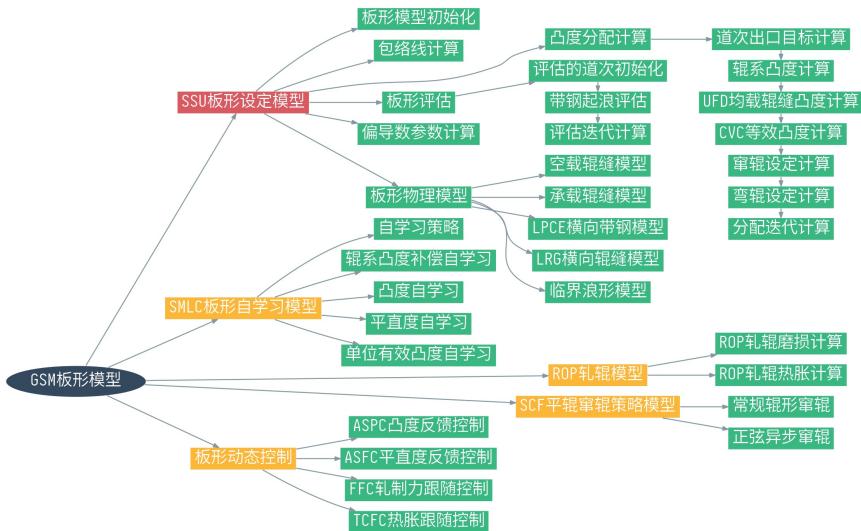
将GSM板形模型完全解剖，如下图所示。



本书从板形设定部分开始，对整个板形模型做详细的阐述。

# 板形设定模型总览

本章节主要介绍板形设定模型，本章节相对于整个GSM的位置如下图中红色节点所示。



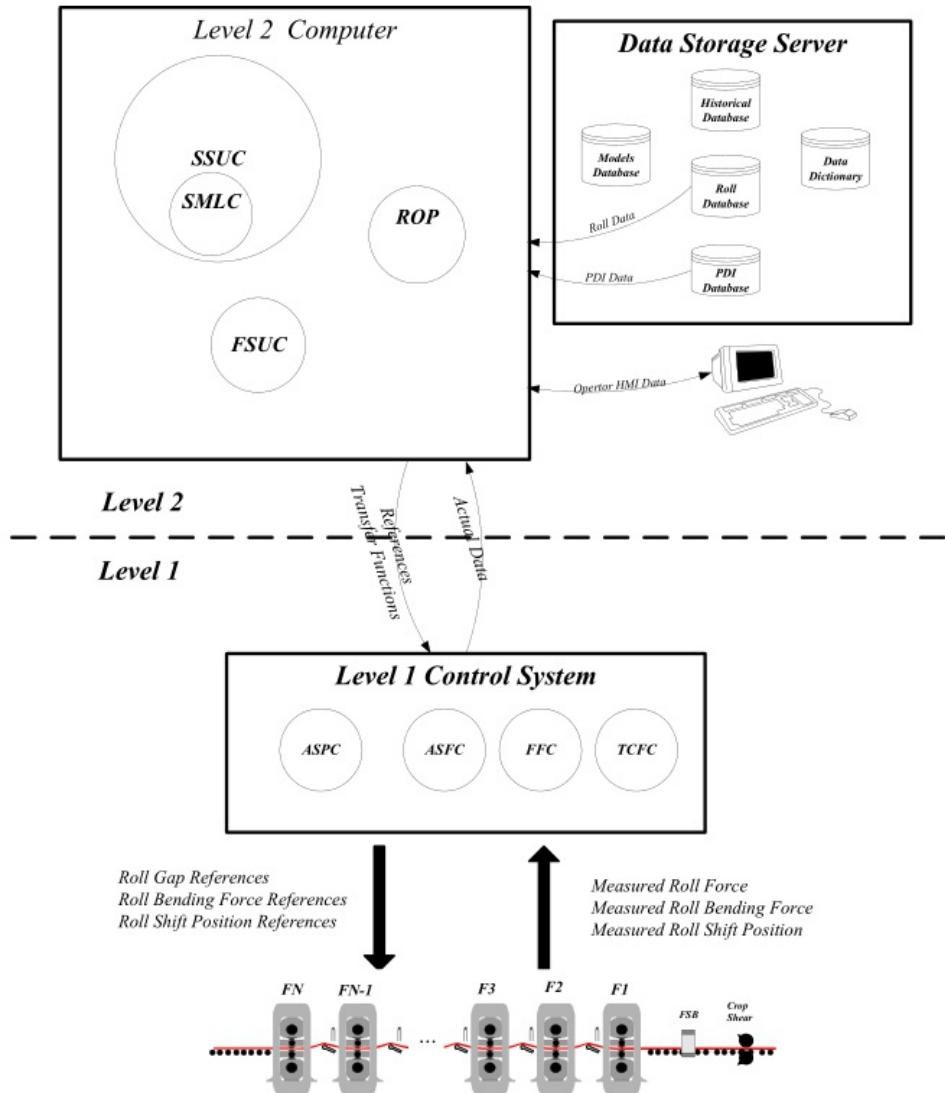
## 板形设定模型的功能

板形设定模型SSU的主要作用是计算并设定执行机构（主要是弯辊、窜辊）的参数，以获得符合质量要求的带钢凸度、带钢平直度。

SSU的输入参数来自FSU，包括各个机架的带钢出入口厚度、宽度、轧制力以及带钢温度等。SSU和FSU之间最重要的联系是各个机架的轧制力，一方面，FSU对轧制力的计算负责，另一方面各个机架的负荷分配直接影响到板形设定模型中的凸度分配。

## 板形模型的信息沟通

板形模型的参数来源于二级FSU的输出，同样也有来自L1一级基础自动化的参数和数据库的参数。板形模型的信息沟通如下图所示。



## 板形设定模型的步骤

板形设定模型的步骤可以分为如下几步：

- 根据执行机构的软硬极限以及浪形临界条件，计算板形控制的包络线。

- 根据轧机板形能力以及末道次机架出口的目标凸度，分配各个机架的入口有效单位凸度。
- 计算窜辊系统和弯辊系统的设定值。
- 评估板形设定值以保证带钢浪形。

## 包络线计算

包络线计算实际反映了轧机的板形控制能力或板形控制极限。首先要确定弯窜辊的软硬极限，再进行凸度分配操作。

执行机构的软硬极限确定了UFD均载辊缝凸度的极限。如下面的公式所示，最小的弯辊力和最大的窜辊位置对应最大的UFD均载辊缝凸度；最大的弯辊力和最小的窜辊位置对应了最小的UFD均载辊缝凸度。

$$Cp_{ufd}|_{max} = func(F_b(min), Sp(max), \theta_{pc}(max))$$

$$Cp_{ufd}|_{min} = func(F_b(max), Sp(min), \theta_{pc}(min))$$

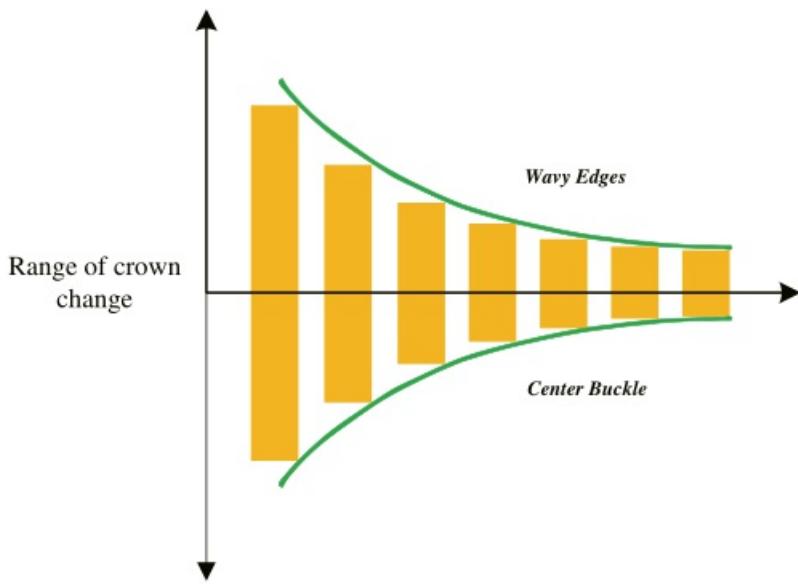
公式中， $F_b$ 为弯辊力， $Sp$ 为窜辊位置， $\theta_{pc}$ 为PC角角度。

根据最大最小UFD均载辊缝凸度可以确定最大最小的入口有效单位凸度。

$$Cp_{entry\_effct}|_{max} = Cp_{ufd}|_{max} - \epsilon_{cb} \cdot \frac{K_{chg\_attn}}{K_{pce\_infl}}$$

$$Cp_{entry\_effct}|_{min} = Cp_{ufd}|_{min} - \epsilon_{we} \cdot \frac{K_{chg\_attn}}{K_{pce\_infl}}$$

带钢的屈曲极限或临界浪形决定了每一个机架的凸度改变量。如下图所示，注意上限代表双边浪的极限，下限代表中浪的极限，两极限中间的区域表征良好板形，超出这个区间的范围表征浪形缺陷。



每个机架的最大的双边浪改变量 $\Delta Cp_{we}$ 和最大的中浪改变量 $\Delta Cp_{cb}$ 构成了各个机架的板形调整能力。

$$\Delta Cp_{we} = \epsilon_{we} \cdot \frac{1 - K_{pce\_infl} \cdot (1 - (1 - f_{prf\_rlf}) \cdot f_{strn\_rlf})}{K_{pce\_infl}}$$

$$\Delta Cp_{cb} = \epsilon_{cb} \cdot \frac{1 - K_{pce\_infl} \cdot (1 - (1 - f_{prf\_rlf}) \cdot f_{strn\_rlf})}{K_{pce\_infl}}$$

已知执行机构约束和带钢浪形约束的条件下，我们协调这二者以获得各个机架的带钢凸度包络线。

协调包络线过程中最重要的一个原则就是，上一个机架的出口单位凸度必须在当前机架的执行机构极限范围内，同时也要在下一个机架的凸度改变范围内。

# 板形模型初始化

板形模型初始化主要包含在cShapeSetupD::Main(..)函数中。

## 状态量的建立

cShapeSetupD对象有一个status状态量，显示当前板形设定的状态（红灯或绿灯），以及判断是否合法的指示器ok。初始默认情况下status设为红灯，ok设为false。

```
this->status = cMdtparam::cs_red;
this->ok     = false;
```

roll\_change\_count是判断是否换辊的计数器。

```
int    roll_change_count (0);
```

redrft\_perm用于判断是否可以重新分配各机架厚度或者重新分配压下。初始情况下redrft\_perm设为false。

后根据如下条件更新redrft\_perm的值。ssu\_load\_enab一般为false。ssu\_granted一般为true。s\_CalId指的是FCD对象中的Calculation ID，最小为1，最大为2。

```
redrft_perm = ( true == pcSched->pcSetupD->pcSetup->ssu_load
_enab ) &&
( true == pcSched->pcFSSched->pcSSys->state.ssu_granted ) &&
( 1 == pcSched->pcFCD->state.s_CalId );
```

redrft\_perm的值与iter相同。

```
if ( redrft_perm )
{
    iter = 1;
```

```
    }
else
{
    iter = 0;
}
```

在日志中的`lst.iter`与源码中此处`iter`并不完全一样。

## 短期自学习设定

短期自学习的设定受到换辊和钢种、规格跳档的影响。

### 跳档

针对钢种和规格的跳档，模型考虑了以下五种情况。

```
this->family_chg      = false;
this->narrow_to_wide_chg = false;
this->wide_to_narrow_chg = false;
this->prd_chg          = false;
this->lot_chg           = false;
```

钢种族跳档、宽度由窄变宽、宽度由宽变窄，这些都好理解。

而`prd_chg`指的是：钢种族跳档、宽度由窄变宽、宽度由宽变窄这三种情况至少有一种出现。

```
if ( (true == this->family_chg) || 
    (true == this->narrow_to_wide_chg) || 
    (true == this->wide_to_narrow_chg) )
{
    this->prd_chg = true;
}
```

`lot_chg`指的是和前一块带钢相比，钢种族、厚度索引、宽度索引其中至少一者发生改变，则称为`lot_chg`。用于弯辊力干预自适应的计算。

```

if ( // family change
    ((pcSched->pcFSSched->pcSAMP->state.pr_family > 0) &&
    (abs(pcSched->pcPDI->state.family - pcSched->pcFSSched->pcS
AMP->state.pr_family) > 0)) ||
    // gauge range table index change
    ((pcSched->pcFSSched->pcSAMP->state.pr_grt_idx > 0) &&
    (abs(pcSched->pcPDI->state.grt_idx - pcSched->pcFSSched->pc
SAMP->state.pr_grt_idx) > 0)) ||
    // width range table index change
    ((pcSched->pcFSSched->pcSAMP->state.pr_wrt_idx > 0) &&
    (abs(pcSched->pcPDI->state.wrt_idx - pcSched->pcFSSched->pc
SAMP->state.pr_wrt_idx) > 0)) )
{
    this->lot_chg = true;
}

```

## 短期自学习预设定CSAMP::Reset\_Verniers(..)

根据是否跳档以及换辊，短期自学习在开始进行设定计算前，更新凸度和平直度的自学习。所谓的预设定其实是在不同的条件下清零自学习。

- 当有机架换辊，则累积增加相应机架的换辊次数num\_rolls\_chgd。
- 若超过两个机架换辊，则清零凸度自学习与平直度自学习（vrn和err），并初始化设定辊形自学习，设定last\_pos\_shft为0。
- 当出现lot\_chg或开轧，清零弯辊力补偿以及bnd\_ofs\_counter。
- 当出现钢种族跳档，则凸度自学习（vrn和err）衰减后继承，衰减系数目前0.99。
- 如果出现窄到宽的跳档，同时周期内轧制块数超过30块，并且存在中浪趋势，则清零平直度自学习（vrn和err），若末机架工作辊辊形自学习也小于零，则清零末机架工作辊辊形自学习wr\_crn\_vrn。
- 若出现宽到窄的跳档，不做任何设定修改。

更新完短期自学习表之后，并且当前计算阶段处于course-2之后，则put短期自学习表到模型数据库（models database）。

## cShapeSetupD::Init(..)

cShapeSetupD::Init(..)初始化了动态的SHAPESETUP对象以及其它相关的动态对象，比如：LPCE、LRG、UFD和TARGT。除此之外，这个函数还计算了执行机构的软极限，同时复制外部的数据给合适的动态对象。

cShapeSetupD::Init(..)初始化之后，最初的哪两个状态布尔值更新为true。

```
this->ok      = true;
this->status  = cMd1param::cs_green;
```

cShapeSetupD::Init(..)的实现在shapsetup\_req.cxx文件中。

## 长短期自学习初始化

首先初始化凸度和平直度的目标tgt\_profile和tgt\_flatness。这两个目标一开始是PDI目标加上操作工的补偿。

模型用prf\_vrn\_sel\_flag和flt\_vrn\_sel\_flag这两个参数来标识长短期自学习的选择，默认以长期自学习为主。

自学习值包括凸度自学习prf\_vrn\_rm\_tmp和prf\_vrn\_rs\_tmp，以及平直度自学习flt\_vrn\_tmp，初始的凸度或平直度自学习为长期自学习。当这一块带钢和上一块带钢相比，出现钢种或规格跳档，则将长期自学习加上上一块增益后的短期自学习，作为新的自学习值（注意这个功能凸度自学习已经取消增益，仅长期自学习值加上短期自学习）。

## cTargtD::Init(..)

在cTargtD::Init(..)中主要确定初始的凸度以及目标有效凸度的极限。prf\_vrn是prf\_vrn\_rm\_tmp和prf\_vrn\_rs\_tmp的差，flt\_vrn就是flt\_vrn\_tmp。

目标flt为pdi平直度目标加上平直度的操作工补偿。

初始目标凸度与平直度稍有差别。

```
prf_int = (pdi_prf + prf_op_off) * matl_exp_cof + prf_vrn;
```

prf\_int为pdi凸度加上操作工补偿后的热态凸度，再加上凸度自学习量。也就是说，凸度自学习量是补偿热态下的凸度。

之后用凸度的容许偏差计算单位凸度的上下极限。

## 初始化的大循环

pcTargtD->Init(..)执行完之后，从首道次机架从前往后进行一系列的初始化工作，将近700行代码。

首先将sprp的相关调整系数初始化到相应的对象中（pcFSStdD），供后续板形计算使用，如ufd\_mult和force\_bnd\_nom。

对非空道次计算出入口厚度对轧制力的偏导数或增益DForce\_DEnthick、DForce\_DExthick。之后初始化板形相关的动态对象，按先后顺序分别为UFD对象、CRLC对象和LRG对象。

注意在CRLC对象的初始化中，SPRP中的工作辊凸度补偿f\_wr\_crn\_off\_adj需要加到长期自学习工作辊凸度补偿psSLFG->wr\_crn\_off上。

f\_wr\_crn\_off\_adj的设定根据出口凸度分为三档。

```
float f_wr_crn_off_adj = 0.0F;
if ( pcTargtD->prf_del < 0.045F )
{
    f_wr_crn_off_adj = psSPRP->wr_crn_off_adj [ passIdx ] ;
}
else if ( pcTargtD->prf_del < 0.065F )
{
    f_wr_crn_off_adj = psSPRP->wr_crn_off_adj2 [ passIdx ];
}
else
{
    f_wr_crn_off_adj = psSPRP->wr_crn_off_adj3 [ passIdx ] ;
}
```

初始化分配和评估的横向带钢对象，利用pcFSPassD->pcAlcLPceD[ iter ]->Init(..)和pcFSPassD->pcEvLLPceD[ iter ]->Init(..)进行初始化。

之后初始化和STD对象相关的参数。

- 窜辊和弯辊能否使用的标识shft\_enab和bnd\_enab。
- op弯辊力补偿op\_bnd\_off。
- 弯辊力pcFSStdDloc->force\_bnd。
- 初始化窜辊位置。
- 计算窜辊的软极限。
- 计算弯辊的软极限。
- 插值计算中间坯凸度prf\_pass0和中间坯单位凸度pu\_prf\_pass0。
- 利用长期平直度自学习值修正末道次F7的弯辊力极限force\_bnd\_lim。

同样在STD初始化过程中有这几点需要注意。

第一，注意GSM\_RB\_OFS\_BLEEDOFF这个宏在ssu\_features.hxx中定义为(1)。

```
#define GSM_RB_OFS_BLEEDOFF          (1)           // [-] GSM Operator RB offsets bleed off enabled
```

如果GSM\_RB\_OFS\_BLEEDOFF为真或为1，需要用短期自学习的弯辊补偿bending\_ofs修正op弯辊力补偿op\_bnd\_off。

第二，如果bnd\_enab为true，并且轧辊辊形是平辊，则不用目标平直度自学习量flt\_vrn修正（从中减去）初始的弯辊力force\_bnd\_nom；如果轧辊不是平辊，则用目标平直度自学习量flt\_vrn修正（从中减去）初始的弯辊力force\_bnd\_nom。

```
if ( pcFSStdDloc == pcLstFSPassD->pcFSStdD[ iter ]  
&& pcFSStdDloc->pcStdRollPrD->getProf(op_work) != rp_parab )  
{  
//-----  
// If last stand and the roll is not parabolic it SHOULD be CVC  
//-----  
    pcFSStdDloc->force_bnd = pcFSStdDloc->pcFSStd->force_bnd_no  
    m  
    - pcTargetD->flt_vrn;  
}  
else  
{
```

```
    pcFSStdDloc->force_bnd = pcFSStdDloc->pcFSStd->force_bnd_nom  
;  
}
```

force\_bnd的值和从属的对象必须从日志中理清楚。如psSSys->force\_bnd和pcFSStdDloc->force\_bnd。

第三，设定了一个level\_std标识，当换辊开轧后用于锁定窜辊，以便操作工进行调平。

```
if ( pcShapeSetup->num_coils_to_lvl >=  
    pcFSStdDloc->pcStdRollPrD->getNBarRolled(rpos_top, op_work)  
)  
{  
    level_std = true;  
}
```

对于平辊，在轧制用于调平的带钢时，相应机架的窜辊位置为零位。

```
// these are parabolic rolls  
// use SCF function references  
if ( level_std )  
    pcFSStdDloc->wr_shft = 0.0F;  
else  
    pcFSStdDloc->wr_shft = psSSys->targ_pos_shft[ passIdx ];
```

在确定窜辊软极限过程中，如果窜辊被操作工锁定了，那么窜弯辊的软极限就是当前窜辊值。

在确定窜辊软极限过程中，需要先根据窜辊的速度计算最大的窜辊位置变化量。如果是平辊的话直接选取SCF中的设定。

在中间坯凸度和单位凸度的计算中，如果有中间坯测量的凸度则用中间坯测量的凸度，如果没有则插值计算。中间坯或0道次的单位凸度和单位有效凸度包络线最大最小值均为中间坯单位凸度的计算值。同时pcTargD-

>en\_pu\_prf等于pu\_prf\_pass0。代码中pcFSPassD为中间坯道次。

```
for ( int i = minl; i <= maxl; i++ )
{
    pcFSPassD->pcVecPEnvD[ iter ]->pu_prf_env[ i ]     =
        pu_prf_pass0;
    pcFSPassD->pcVecPEnvD[ iter ]->ef_pu_prf_env[ i ]   =
        pu_prf_pass0;
}

//-----
-----
// Initialize mill entry per unit profile
//-----
-----
pcTargtD->en_pu_prf = pu_prf_pass0;
```

注意pcTargtD->flt\_vrn和psSLFG->flt\_vrn、psSAMP->flt\_vrn之间的区别。

## cShapeSetupD::References(..)

cShapeSetupD::References(..)计算了凸度与平直度控制目标下的相关设定值，必要情况下重新分配轧制力或压下。

首先从F1到F7更新动态LPCE对象和LRG对象。

进行包络线计算cPEnvD::Calculate(..)和分配计算cAlcD::Calculate(..)，之后从F1到F7对板形设定进行评估。

注意在每一道次评估前需要用op弯辊力补偿修正弯辊力。

```
force_bnd =
    pcFSPassD->pcFSStdD[ iter ]->force_bnd +
    pcFSPassD->pcFSStdD[ iter ]->op_bnd_off;
```

如果可以重新分配压下，或者平直度不满足条件，则需要重新进行板形分配计算和再一次的评估。

最后计算传递给一级的增益，以及用pcTargtD->Eval\_Delvry\_Pass(..)评估末道次。

# 板形包络线计算

在cPEnvD::Calculate(..)这个函数中计算包络线。

## F1到F7包络线初始化

包络线计算初始化的过程中，从F1到F7前机架到末机架，计算各个道次的ufd\_pu\_prf\_env以及std\_ex\_strn\_lim，并利用这二者计算出入口（上一道次）有效单位凸度的极限pcFSPassD->pcPrvAct->pcPEnvD->ef\_pu\_prf\_lim。

首先根据弯窜辊的极限值确定弯窜辊包络线的最大最小值。注意，弯辊力极限的最大值对应弯辊力包络线的最小值，弯辊力极限的最小值对应弯辊力包络线的最大值。窜辊的极限最值和包络线的最值同样是相反对应的。

```
// 弯辊力包络线的赋值  
pcFSPassD->pcPEnvD->force_bnd_env[ minl ] =  
    pcFSPassD->pcFSStdD[ iter ]->force_bnd_lim[ maxl ];  
  
pcFSPassD->pcPEnvD->force_bnd_env[ maxl ] =  
    pcFSPassD->pcFSStdD[ iter ]->force_bnd_lim[ minl ];  
  
// 窜辊位置包络线的赋值  
pcFSPassD->pcPEnvD->pos_shft_env[ minl ] =  
    pcFSPassD->pcFSStdD[ iter ]->wr_shft_lim[ maxl ];  
  
pcFSPassD->pcPEnvD->pos_shft_env[ maxl ] =  
    pcFSPassD->pcFSStdD[ iter ]->wr_shft_lim[ minl ];
```

之后用窜辊位置的极限值，代入pcCRLCD->Crns(..)计算辊系凸度的极限值pce\_wr\_crn\_lim[maxl/minl]。最大值对应最大值，最小值对应最小值。

单位轧制力的大小极限值，直接用单位轧制力赋值。可以见板形模型单位轧制力的验算内容。

```
for ( i = minl; i <= maxl; i++ )
```

```

{
    pcFSPassD->pcPEnvD->force_pu_wid_lim[ i ] =
        pcFSPassD->pcPEnvD->force_pu_wid;
}

```

用默认窜辊位置wr\_shft\_nom，代入pcCRLCD->Crns(..)计算辊系凸度pce\_wr\_crn和wr\_br\_crn。

计算pcFSPassD->pcPEnvD->dprf\_dfrcw，这里的偏导数dprf\_dfrcw是用来判断force\_pu\_wid\_lim给force\_pu\_wid\_env赋值的方向。

```

//-----
// Calculate the UFD roll gap profile derivative with respect
// rolling force per unit piece width.
//-----

pcFSPassD->pcPEnvD->dprf_dfrcw =
    pcFSPassD->pcFSStdD[ iter ]->pcUFDD->Dprf_Dfrcw (
        pcFSPassD->pcPEnvD->force_pu_wid,
        pcFSPassD->pcFSStdD[ iter ]->pcFSStd->force_bnd
    _nom,
        pce_wr_crn,
        wr_br_crn
    );

//-----
// Initialize the rolling force per unit width piece envelope.
// Note: The UFD roll gap profile derivative is used for direction.
//-----

if ( 0.0 <= pcFSPassD->pcPEnvD->dprf_dfrcw )
{
    pcFSPassD->pcPEnvD->force_pu_wid_env[ min1 ] =
        pcFSPassD->pcPEnvD->force_pu_wid_lim[ min1 ];
    pcFSPassD->pcPEnvD->force_pu_wid_env[ max1 ] =
        pcFSPassD->pcPEnvD->force_pu_wid_lim[ max1 ];
}
else

```

```

{
    pcFSPassD->pcPEnvD->force_pu_wid_env[ minl ] =
        pcFSPassD->pcPEnvD->force_pu_wid_lim[ maxl ];
    pcFSPassD->pcPEnvD->force_pu_wid_env[ maxl ] =
        pcFSPassD->pcPEnvD->force_pu_wid_lim[ minl ];
}

```

辊系凸度的极限值赋值给辊系凸度的包络线最大最小值。注意这里，极限值的最小值对应包络线的最大值，极限值的最大值对应包络线的最小值。

```

//-----
----- // Initialize the piece to work roll stack crown
n and work roll
// backup roll stack crown envelopes.
//-----

----- pcFSPassD->pcPEnvD->pce_wr_crn_env[ minl ] =
pcFSPassD->pcPEnvD->pce_wr_crn_lim[ maxl ];
pcFSPassD->pcPEnvD->wr_br_crn_env[ minl ] =
pcFSPassD->pcPEnvD->wr_br_crn_lim[ maxl ];
pcFSPassD->pcPEnvD->pce_wr_crn_env[ maxl ] =
pcFSPassD->pcPEnvD->pce_wr_crn_lim[ minl ];
pcFSPassD->pcPEnvD->wr_br_crn_env[ maxl ] =
pcFSPassD->pcPEnvD->wr_br_crn_lim[ minl ];

```

此时，我们已经有了单位轧制力包络线force\_pu\_wid\_env、弯辊力的包络线force\_bnd\_env、辊系凸度的包络线pce\_wr\_crn\_env和wr\_br\_crn\_env，利用pcUFDD->Prf(..)计算出UFD单位凸度的包络线ufd\_pu\_prf\_env。

```

for ( i = minl; i <= maxl; i++ )
{
//-----
----- // Establish the minimum / maximum UFD roll
gap per unit profile
// envelope.
//-----
-----
```

```

        line_num = __LINE__;
        pcFSPassD->pcPEnvD->ufd_pu_prf_env[ i ] =
            pcFSPassD->pcFSStdD[ iter ]->pcUFDD->Prf
(
                pcFSPassD->pcPEnvD->force_pu_wid_e
nv [ i ],
                pcFSPassD->pcPEnvD->force_bnd_env
[ i ],
                pcFSPassD->pcPEnvD->pce_wr_crn_env
[ i ],
                pcFSPassD->pcPEnvD->wr_br_crn_env
[ i ] ) /
                pcFSPassD->pcFSStdD[ iter ]->pcExPceD->
thick;
}

```

确定出口应变差极限std\_ex\_strn\_lim。

```

for ( i = we; i <= cb; i++ )
{
    //-----
    //----- Retrieve the piece critical buckling limits
for the given pass.
    //-----
    //-----

    line_num = __LINE__;
    pcFSPassD->pcPEnvD->std_ex_strn_lim[ i ] =
        pcFSPassD->pcLPceD->Crit_Bckl_Lim( i );
}

```

利用各个道次的ufd\_pu\_prf\_env以及std\_ex\_strn\_lim二者计算出入口（上一道次）有效单位凸度的极限pcFSPassD->pcPrvAct->pcPEnvD->ef\_pu\_prf\_lim。如果当前道次的带钢影响系数接近0，则松弛入口（上一道次）有效单位凸度的极限为正负1。其中ufd单位凸度包络线最小值和边浪极限一起参与计算，ufd单位凸度巴洛熙最大值和中浪极限一起参与计算。

```
//-----
```

```

-----
// Calculate the effective per unit profile limits.
//-----

-----
line_num = __LINE__;
pcFSPassD->pcPrvAct->pcPEnvD->ef_pu_prf_lim[ mi
nl ] =
    pcFSPassD->pcFSStdD[ iter ]->pcLRGD->Ef_En_
PU_Prf1(
    pcFSPassD->pcPEnvD->ufd_pu_prf_e
nv [ minl ],
    pcFSPassD->pcPEnvD->std_ex_strn_
lim [ we ] );

line_num = __LINE__;
pcFSPassD->pcPrvAct->pcPEnvD->ef_pu_prf_lim[ ma
x1 ] =
    pcFSPassD->pcFSStdD[ iter ]->pcLRGD->Ef_En_
PU_Prf1(
    pcFSPassD->pcPEnvD->ufd_pu_prf_e
nv [ maxl ],
    pcFSPassD->pcPEnvD->std_ex_strn_
lim [ cb ] );

```

考虑到最后一道次即末道次的目标可能发生变化，因此最后一道次的有效单位凸度的极限也松弛为正负1。

第一道次入口的pcLpceD->ef\_pu\_prf赋值给第一道次入口的pcPEnvD->ef\_pu\_prf\_env包络线。同时初始化包络线限制道次数。

```

for ( i = minl; i <= maxl; i++ )
{
    //-----

-----
// Initialize first pass effective entry per unit p
rofile.
    //-----


-----
( ( cFSPassD* )(pcFstFSPassD->previous_obj) )->pcPE
nvD->ef_pu_prf_env[ i ] =

```

```

        ( ( cFSPassD* )(pcFstFSPassD->previous_obj) )->
pcLPceD->ef_pu_prf;

        //-----
        // Initialize the limiting pass envelope.
        //-----
        pas_env_lim[ i ] =
        ( ( cFSPassD* )(pcFstFSPassD->previous_obj) )->
pcPass->num;
    }

```

## 协调包络线

### 确定包络线最小值组份

move\_prv用来指示道次是否前移。是否前移，说明上一机架的有效单位凸度包络线是否存在调整变化的空间。

利用pcLRGD->Ef\_Ex\_PU\_Prf3(..)计算，将上一道次的有效单位凸度包络线下限和本道次的ufd有效凸度包络线代入，获得本道次出口的有效单位凸度包络线下限pcFSPassD->pcPEnvD->ef\_pu\_prf\_env[minl]。

```

pcFSPassD->pcPEnvD->ef_pu_prf_env[ minl ] =
pcFSPassD->pcFSStdD[ iter ]->pcLRGD->Ef_Ex_PU_P
rf3 (
        pcFSPassD->pcLPceD->strn_rlf_cof,
        pcFSPassD->pcPrvAct->pcPEnvD->ef
        _pu_prf_env[ minl ],
        pcFSPassD->pcPEnvD->ufd_pu_prf_e
        nv[ minl ] );

```



如果本道次出口的有效单位凸度包络线下限低于有效单位凸度极限的下限，则需要进行一系列重新计算。

重新计算其中包括ufd\_pu\_prf、istd\_ex\_pu\_prf、ef\_en\_pu\_prf，并利用上一道次的ef\_pu\_prf\_env来clamp获得入口有效单位凸度（包络线下限）临时值ef\_en\_pu\_prf\_buf。

```
ef_en_pu_prf_buf =
    cMathUty::Clamp( ef_en_pu_prf,
        pcFSPassD->pcPrvAct->pcPEnvD->ef_pu_prf_env[ minl ],
        pcFSPassD->pcPrvAct->pcPEnvD->ef_pu_prf_env[ maxl ] );
```

之后更新move\_prv的指示器。

```
move_prv[ minl ] =
    ( ef_en_pu_prf_buf != pcFSPassD->pcPrvAct->pcPEnvD->ef_pu_
    _prf_env[ minl ] ) &&
    ( pcFSPassD->pcPrvAct->pcPEnvD->ef_pu_p
    rf_env[ minl ] != pcFSPassD->pcPrvAct->pcPEnvD->ef_pu_
    _prf_env[ maxl ] );
```

更新上一道次或入口有效单位凸度极限的最小值，注意是极限。

```
pcFSPassD->pcPrvAct->pcPEnvD->ef_pu_prf_lim[ minl ] = ef_en_pu_
prf_buf;
```

这时判断move\_prv前移指示器的状态，如果不能前移，说明上一道次的有效单位凸度或入口有效单位凸度不存在可调整和变化的空间，则将入口有效包络线的下限赋值给临时量ef\_en\_pu\_prf\_buf。

```
if ( !move_prv[ minl ] )
{
    ef_en_pu_prf_buf =
        pcFSPassD->pcPrvAct->pcPEnvD->ef_pu_
    _prf_env[ minl ];
}
```

之后用新的ef\_en\_pu\_prf\_buf值和ef\_ex\_pu\_prf值，更新ufd有效单位凸度ufd\_pu\_prf，涉及的函数是pcLRGD->UFD\_PU\_Prf3(..)。并利用新的ufd单位凸度ufd\_pu\_prf、弯辊力和窜辊包络线的下限，计算变化后的辊系凸度pce\_wr\_crn和wr\_br\_crn。

再用pcCRLCD->Shft\_Pos(..)更新窜辊位置包络线的下限。之后再次重计算(re-calculate)辊系凸度。接着考虑弯辊力包络线下限force\_bnd\_env\_min，重计算(re-calculate)辊系凸度。

```
//-----
-----  
// Re-calculate the following composite roll stack crown quantities:  
//      Piece to work roll stack crown  
//      Work roll to backup roll stack crown  
//-----  
-----  
line_num = __LINE__;  
pcFSPassD->pcFSStdD[ iter ]->pcCRLCD->Crns(  
    pcFSPassD->pcPEnvD->pos_shft_env [ m  
inl ],  
    pcFSPassD->pcPEnvD->angl_pc_env [ m  
inl ],  
    pcFSPassD->pcPEnvD->pce_wr_crn_env[ m  
inl ],  
    pcFSPassD->pcPEnvD->wr_br_crn_env [ m  
inl ] );
```

在弯窜辊都修正辊系凸度后，确定合适的弯辊力force\_bnd\_des，设定force\_bnd\_clmp指示器。

```
force_bnd_clmp =  
    force_bnd_des != pcFSPassD->pcPEnvD->force_  
bnd_env[ minl ];
```

最终更新ufd有效单位凸度包络线的下限ufd\_pu\_prf\_env\_min。

如果force\_bnd\_des不等于弯辊力包络线的下限值，那么还需要调整。重新更新ef\_en\_pu\_prf\_buf以及更新上一道次或入口有效单位凸度极限的最小值。

如果未前移，则进行如下计算，从1190到1599行。如果出口应变差std\_ex\_strn超出出口应变差的极限范围，则进行一系列修正，目前这段修正在模型中被禁止执行。这样设置的原因是避免单位轧制力包络线和辊系凸度做大规模的修改和变化影响生产稳定性，出点浪形问题也是可以接受的。

```
// was profile reduced too much
//if ( std_ex_strn < pcFSPassD->pcPEnvD->std_ex
 _strn_lim[ cb ] )
    if ( 1 < 0 )
    {...}
    // strain too high (due to low entry profile)
    //if ( std_ex_strn > pcFSPassD->pcPEnvD->std_ex
 _strn_lim[ we ] &&
        if ( 0 > 1 &&
            pcFSPassD->pcFSStdD[ iter ]->pcLRGD->pce_i
 nfl_cof >
            pcFSPassD->pcPEnvD->pcPEnv->pce_infl_c
 of_mn )
    {...}
    // 1 < 0 和 0 > 1说明这两段调整永远不会执行
```

最后再更新一次本道次的出口有效单位凸度包络线下限。

```
pcFSPassD->pcPEnvD->ef_pu_prf_env[ minl ] =
    pcFSPassD->pcFSStdD[ iter ]->pcLRGD->Ef_Ex_PU_P
rf3(
    pcFSPassD->pcLPceD->strn_rlf_cof,
    pcFSPassD->pcPrvAct->pcPEnvD->ef_pu_
    prf_env[ minl ],
    pcFSPassD->pcPEnvD->ufd_pu_prf_env
    [ minl ] );
```



# 凸度分配计算

凸度分配计算主要在cAlcD::Calculate(..)函数中执行。

## Delvry\_Pass(..)之前

cAlcD::Calculate(..)开始时，首先赋值中间坯的“分配厚度”。中间坯的“分配厚度”实际为F1的入口厚度，即中间坯的实际厚度。接着计算F1到F7的单位轧制力、分配厚度，以及引用轧辊咬入相关的对象。

如果可以重新分配压下，那么还会计算轧制力的最大改变量。

之后计算总的单位凸度改变量pu\_prf\_change\_sum。

```
pu_prf_change_sum +=  
    pcFSPassDttmp->pcEv1LPceD[ iter ]->strn_rlf_cof  
    / (pcFSPassDttmp->pcFSStdD[ iter ]->pclRGD->pce_infl_cof  
    * pcFSPassDttmp->pcEv1LPceD[ iter ]->elas_modu);
```

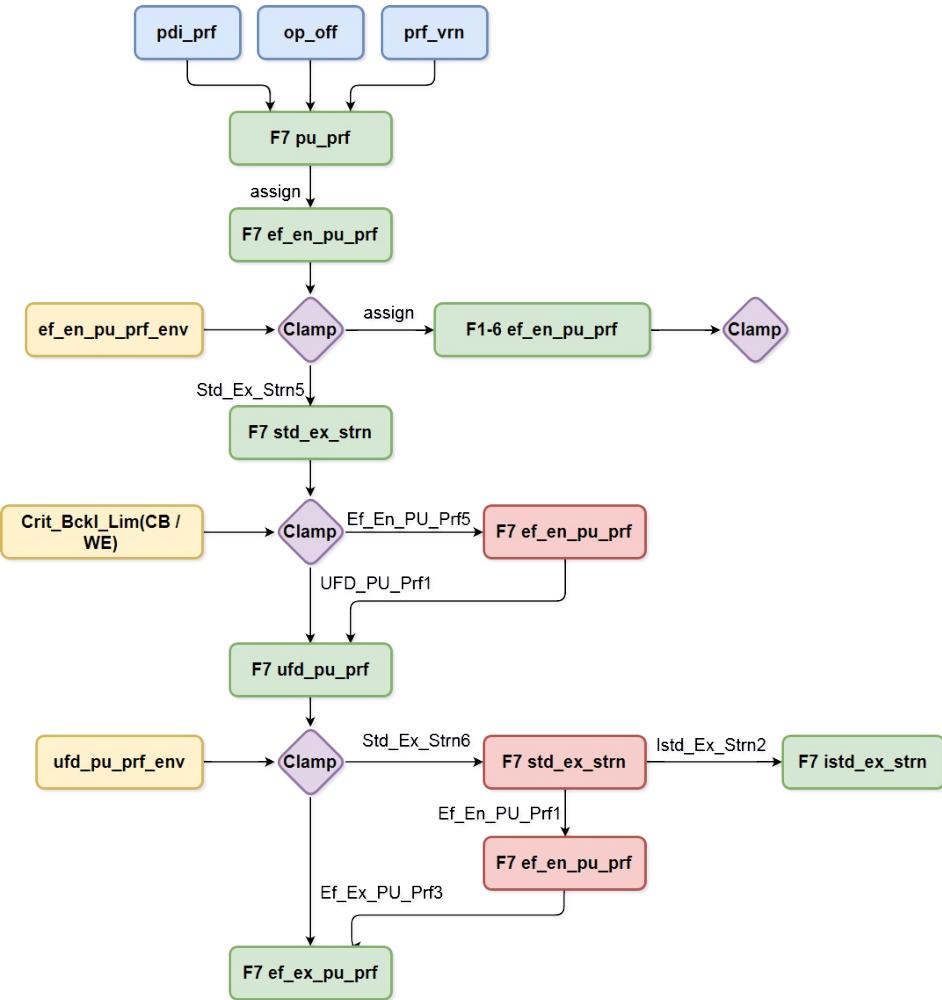
或者表示为：

$$pu\_prf\_change\_sum = \sum_{i=1}^7 \left( \frac{strn\_rlf\_cof}{pce\_infl\_cof} \cdot Q_{elas\_modu} \right)$$

## Delvry\_Pass(..)

Delvry\_Pass(..)计算F7或最后一非空过道次的入口和出口有效单位凸度，以及出口istd应变差。

计算流程如下图所示。



基础量是加了操作工补偿和凸度自学习的单位凸度pu\_prf。由pu\_prf\_env限幅。

首先将pu\_prf直接赋值给最后一道次（F7）入口有效单位凸度ef\_en\_pu\_prf。

之后将F7的ef\_en\_pu\_prf直接赋值给F1到F6的入口有效单位凸度ef\_en\_pu\_prf，并由各个道次的ef\_pu\_prf\_env限幅。这是先假设所有机架单位凸度相同，理想状态下的情况。

回到F7，用Std\_Ex\_Strn5计算末道次的机架出口应变差std\_ex\_strn，并用中浪和边浪的判别极限Crit\_Bckl\_Lim限幅。在判别极限中，中浪对应负值，边浪对应正值。限幅操作的意义是看std\_ex\_strn是否超出判别极限，若超出则一定

会出现浪形，则当前std\_ex\_strn的值肯定不合适，需要重新计算，但是std\_ex\_strn的值依赖ef\_en\_pu\_prf。因此，通过函数Ef\_En\_PU\_Prf5，利用限幅后的std\_ex\_strn和目标pu\_prf重计算F7的ef\_en\_pu\_prf。

接着利用新的std\_ex\_strn计算F7的ufd\_pu\_prf\_buf。同样在限幅时，若发现当前的ufd\_pu\_prf\_buf不合适，需要重新计算。但是ufd\_pu\_prf的计算依赖于之前的std\_ex\_strn、ef\_en\_pu\_prf计算结果，因此必须对这两个值重新计算。

最后，我们有重新计算的std\_ex\_strn、ef\_en\_pu\_prf值。即可通过新的std\_ex\_strn值，计算出F7的出口istd应变差。利用新的ef\_en\_pu\_prf和ufd\_pu\_prf\_buf计算出F7的出口有效单位凸度ef\_ex\_pu\_prf。

注意在Delvry\_Pass(..)中的均载辊缝凸度只是作为中间计算结果存在，与后面分配阶段的ufd\_pu\_prf有所区别。

## 凸度分配计算的大循环

前期准备工作做完后进入凸度分配计算的大循环。

### 局部指针的引用

在每个循环体开始执行时，先用局部指针指向本次循环要用到的所有相关动态对象。

```
// create pointers to class objects that are part of this pass
pcStdD    = pcFSPassD->pcFSStdD[ iter ];
pcCRLCD   = pcStdD->pcCRLCD;
pcAlcD    = pcFSPassD->pcAlcD;
pcUFDD    = pcStdD->pcUFDD;
pcLRGD    = pcStdD->pcLRGD;
pcLPceD   = pcFSPassD->pcLPceD;
pcPEnvD   = pcFSPassD->pcPEnvD;
pcEnPceD  = pcStdD->pcEnPceD;
pcExPceD  = pcStdD->pcExPceD;
// create a pointer to the previous active pass
pcPrvAct = pcFSPassD->pcPrvAct;
```

目的是为提高性能。

## 更新综合辊缝凸度

凸度方面，模型首先更新综合辊缝凸度，保证带钢-工作辊凸度pce\_wr\_crn和工作辊-支承辊凸度wr\_br\_crn是当前状态下的最新值。

```
//-----
-- 
// Calculate the following composite roll stack crown quantitie
s:
//      Piece to work roll stack crown
//      Work roll to backup roll stack crown
//-----
-- 
line_num = __LINE__;
pcCRLCD->Crns ( pcStdD->wr_shft,
                    pcStdD->angl_pc,
                    pce_wr_crn,
                    wr_br_crn );
```

pcCRLCD->Crns(..)的计算详见CRLC模块说明。

## 空过的分配处理

分配从F7或末道次机架，从后往前倒者来。

若非末道次机架中，若本道次空过，则传递本道次的出口厚度给上游机架，也就是空过的机架前后带钢厚度不变。并且设定本道次均载辊缝凸度ufd\_pu\_prf为0。

```
if( pcStdD->dummied )
{
    //-----
-- 
-- 
    // 保存带钢的出口厚度给入口厚度.
    //-----
```

```
-----  
        ( ( cFSPassD* )pcFSPassD->previous_obj )->pcAlcD  
->thick = pcAlcD->thick;  
        pcAlcD->ufd_pu_prf = 0.0;  
    }  
-----
```

非空过部分的计算持续到start\_over之前。

## 咬入计算与单位轧制力约束

之后进行带钢的咬入计算，咬入计算的输入量有入口宽度、出入口厚度、出入口张力、轧制速度，计算输出量有单位轧制力、前滑值和接触弧长度。

```
pcAlcD->pcRollbite->Calculate(      //@S014  
    &rbStatus,                      // OUT  
    status from calculations  
    &force_pu_wid_buf,             // OUT  
    rolling force/width  
    &fwd_slip,                    // OUT  
    exit slip ratio [-]  
    &arcon,                        // OUT  
    length of arc [minor_length]  
    ( ( cFSPassD* )pcFSPassD->previous_obj )->p  
    cAlcD->thick, // IN entry_thk  
    pcAlcD->thick,                // IN  
    exit_thk  
    pcEnPceD->width,              // IN  
    exit/entry width  
    pcStdD->speed,                // IN  
    roll peripheral speed  
    pcEnPceD->tension,            // IN E  
    ntry tension  
    pcExPceD->tension )           // IN E  
    xit tension
```

为什么把咬入计算放在这里，是因为后面有重分配压下的打算，即redrft\_perm为true时，需约束单位轧制力。

```

if( redrft_perm )
{
    pcAlcD->force_pu_wid = (float) force_pu_wid_buf;

    //-----
    // Restrict the rolling force per unit piece width to within
    // the rolling force per unit piece width envelope.
    //

    line_num = __LINE__;
    cAlcD::Eval_Frc_PU_Wid( force_pu_wid_clp,
                            pcAlcD->force_pu_wid,
                            pcStdD->force_strip / p
    cStdD->pcEnPceD->width,
                            pcPEnvD->force_pu_wid_e
    nv,
                            pcAlcD->pcRollbite->Pre
    cision() );
}

pcAlcD->flt_ok = true;

```

约束完单位轧制力后，设定一个标识浪形是否合格的指示器：pcAlcD->flt\_ok，其默认值为true。

## 均载辊缝单位凸度的计算

最重要的计算到来了，均载辊缝单位凸度的计算。

注意这里有两个均载辊缝单位凸度，一个是pcLPceD->ufd\_pu\_prf，另一个是pcAlcD->ufd\_pu\_prf，这两个ufd\_pu\_prf是相对的，因为要比较它们之间的偏差。

如果下游机架的带钢影响系数为0，则只计算实际的pcAlcD->ufd\_pu\_prf，不更新pcLPceD->ufd\_pu\_prf。如果下游机架的带钢影响系数不为0，则计算目标均载辊缝单位凸度pcAlcD->ufd\_pu\_prf，接着计算弯窜辊，最后用18项线性方程更新pcLPceD->ufd\_pu\_prf。

计算弯窜辊过程中可以选择执行机构的计算先后顺序，目前是先计算窜辊，再计算弯辊。执行机构的计算顺序保存在actr\_prior中。cAlc::actrtyp\_shift的条件则进行窜辊位置的计算，cAlc::actrtyp\_bend的条件则进行弯辊力的计算。注意在优先级别actr\_prior中cAlc::actrtyp\_none，指的是无执行机构执行计算，表示预设位。

在窜辊计算中，首先根据目标均载辊缝单位凸度pcAlcD->ufd\_pu\_prf利用18项线性方程反算综合辊缝凸度：带钢-工作辊凸度pce\_wr\_crn和工作辊-支承辊凸度wr\_br\_crn。之后pce\_wr\_crn代入pcCRLCD->Shft\_Pos(..)计算窜辊位置，最后利用pcCRLCD->Crns(..)和新计算的窜弯辊值更新综合辊缝凸度。

在窜辊计算中，用pcUFDD->Bnd\_Frc(..)反算弯辊力，注意输出量force\_bnd为force\_bnd\_des限幅后的结果。

设定一个表示目标均载辊缝单位凸度和实际均载辊缝单位凸度偏差的指示器。若偏差大于ufd\_pu\_prf\_tol（目前为0.0001）则设定为true，表示均载辊缝单位凸度偏差超出了容许的范围，引出了后面有关alc\_lim缩小偏差的一系列计算。

```
alc_lim = fabs( pcAlcD->ufd_pu_prf - pcLPceD->ufd_pu_prf ) > pc  
AlcD->pcAlc->ufd_pu_prf_tol;
```

## 大循环中redrft\_perm相关计算

重算单位轧制力以及更新相关动态参数。由于redrft\_perm为false，这里的计算内容忽略。

## alc\_lim相关计算

首先计算个局部变量的ufd\_pu\_prf，用pcUFDD->Prf(..)计算，注意若轧制力不允许重新分配，那么这个局部的ufd\_pu\_prf和pcLPceD->ufd\_pu\_prf（目标）是一样的。也就是说，若轧制力重新分配，需要使用新的弯辊力和综合凸度去更新ufd\_pu\_prf。

接着利用pcLRGD->Ef\_En\_PU\_Prf3(..)计算新的入口有效单位凸度ef\_en\_pu\_prf\_buf。此时旧的入口有效单位凸度为ef\_en\_pu\_prf (old)。

上一道次的有效单位凸度包络线当然可以约束ef\_en\_pu\_prf\_buf。但是这种约束并不准确，因为厚度可能会变，因此约束标准应当有所放宽。所以在程序中用出口有效单位凸度ef\_ex\_pu\_prf来约束。约束后新的入口有效单位凸度设为ef\_en\_pu\_prf。

用新的ef\_en\_pu\_prf求出std\_ex\_strn和ef\_ex\_pu\_prf。至此，在这个阶段我们获得了可能合适的出入口有效单位凸度。但是别急，还需要判断出口的浪形，才能决定我们目前的分配凸度是否合适。

分配模型中，如果进入了alc\_lim的计算，在重计算出入口有效单位凸度后必须进行浪形判别。F1和F6本道次的浪形判别，由本道次的应变差和下一道次应变差是否超死区极限决定。F7道次的浪形判别，仅由本道次F7的应变差是否超限决定。

当浪形判别不通过，或者说flt\_ok为假时，可以稍微放宽一点标准。如果非末道次机架的下道次应变差不超死区极限，那么也算本道次浪形判别通过。

```
if ( pcFSPassD != pcLstActFSPassD )
{
    pcAlcD->flt_ok =
        ( std_ex_strn_dn <= bck1_lim_dn[ we ] ) &&
        ( std_ex_strn_dn >= bck1_lim_dn[ cb ] );
}
```

到现在这一步，如果浪形判别还不能通过，那么需要重新设定目标单位凸度。先重新设定目标有效单位凸度，用pcTargD->Pass\_Mill\_Targ(..)计算获得，变量为ef\_pu\_prf\_alt。再利用F7的istd\_ex\_strn反推std\_ex\_strn，结合

`ef_pu_prf_alt`计算出目标单位凸度`pu_prf`。注意在`alc`模块中，大循环的这个位置是`alc`局部`pu_prf`变量第一次介入的地方，局部`pu_prf`变量预设为0。之后对`pu_prf`进行限幅，这样新的目标单位凸度就诞生了。

接着是一个难点问题。

```
pcCritFSPassD = pcFSPassD;
```

将更新了目标单位凸度的道次地址赋值给`pcCritFSPassD`指针。这个`pcCritFSPassD`最开始是指向F7道次的。`pcCritFSPassD`指针设定的意义在于：在迭代计算的过程中，浪形判别不合格的相应道次必须比之前更新过目标单位凸度的道次低。

在`alc_lim`计算过程的最后，若目标单位凸度发生改变，则设定`start_over`指示器为`true`，以进行后续`start_over`的流程。

## 非空过道次的更新

在`alc_lim`计算之后，更新给定条件下的`ufd_pu_prf`、`ef_pu_prf`、`strn`、`prf`，注意这些值都属于`lpce`对象。

## start\_over流程

如果目标单位凸度发生改变，则更新目标单位凸度的迭代次数（累积加一）。之后从F7重新开始大循环的计算，从`Delivry_Pass(..)`起步重算出入口有效单位凸度，并设定F7的有效单位凸度为出口有效凸度。

如果目标单位凸度没有发生改变，说明本道次的浪形是符合判别条件的，不需要更改目标；或者目标均载辊缝凸度达到了实际的均载辊缝凸度。进一步说，有两种情况会进入`start_over`为假的流程，一种是未进入`alc_lim`计算的状态，另一种是进入了`alc_lim`的计算，但是浪形判别合格的状态。则当前道次对象`pcFSPassD`可以前移一个道次。出入口有效单位凸度现在敲定是合适的。

之后是：对不均匀变形道次`ef_en_pu_prf`修正的过程。在此阶段，如下一段代码需要注意，在理解上可能会出错。

```

//-----
// Increment pointer to previous dynamic PASS object.
//-----
pcFSPassD = ( cFSPassD* )pcFSPassD->previous_obj;

//-----
// Save the effective entry per unit profile of the previous
// pass into the effective exit per unit profile for this pass.
//-----
ef_ex_pu_prf = ef_en_pu_prf;

//-----
// Determine the upstream effective per unit profile to aim
// towards using the extreme downstream pass where the piece
// influence coefficient is zero.
//-----
ef_en_pu_prf = cMathUty::Clamp ( ef_ex_pu_prf,
    pcPceIZFSPassD->pcPEnvD->ef_pu_prf_env[ min1 ],
    pcPceIZFSPassD->pcPEnvD->ef_pu_prf_env[ max1 ] );

```

这段代码执行之后，ef\_ex\_pu\_prf这个变量的意义已经发生改变，不再代表本道次的出口有效单位凸度。因为前面我们的pcFSPassD道次对象已经前移一个道次，因此程序的设计者为了简练，直接使用局部变量ef\_en\_pu\_prf代表其它含义。

如代码所示，ef\_ex\_pu\_prf保存的是原ef\_en\_pu\_prf的值，而新的ef\_en\_pu\_prf是受到pcPceIZFSPassD道次ef\_pu\_prf\_env包络线限幅之后的值。

没经过浪形判别或经过浪形判别但没有改变目标单位凸度的过程参数ef\_en\_pu\_prf，必须从出现不均匀变形的机架开始，用每个机架的单位凸度最大改变量约束和修正本道次的ef\_en\_pu\_prf。ef\_en\_pu\_prf修正过程是个循环，从pcPceIZFSPassD道次的下一道次开始，且当前道次的上一道次在不均匀延伸的机架中，直至末道次。

ef\_pu\_prf\_chg[cb/we]并不能直接作为真正的有效单位凸度最大改变量，或真正有效凸度改变约束条件。在不均匀变形的机架中，它需要本道次的ef\_pu\_prf\_env和上一道次的ef\_pu\_prf\_env介入，来获得一个更窄的变化区间

`ef_pu_prf_dlt[minl/maxl]`, 用所有存在不均匀变形机架的这个区间来修正本道次的`ef_en_pu_prf`。

```
//-----
// Calculate the delta effective per unit profile change
// from stand entry to interstand exit.
//-----

ef_pu_prf_dlt[ minl ] =
    cMathUty::Max( pcBufFSPassD->pcFSStd[ iter ]->pcLRGD->ef_pu
    _prf_chg[ cb ],
    cMathUty::Max( ef_ex_pu_prf,
                    pcBufFSPassD->pcPEnvD->ef_pu_prf_env[ m
    inl ] ) -
    cMathUty::Min( ef_en_pu_prf,
                    ((cFSPassD*)pcBufFSPassD->previous_obj)->pc
    PEnvD->ef_pu_prf_env[ maxl ]));

ef_pu_prf_dlt[ maxl ] =
    cMathUty::Min( pcBufFSPassD->pcFSStd[ iter ]->pcLRGD->ef_pu
    _prf_chg[ we ],
    cMathUty::Min( ef_ex_pu_prf,
                    pcBufFSPassD->pcPEnvD->ef_pu_prf_env[ m
    axl ] ) -
    cMathUty::Max( ef_en_pu_prf,
                    ((cFSPassD*)pcBufFSPassD->previous_obj)
    ->pcPEnvD->ef_pu_prf_env[ minl ]));
```

`ef_pu_prf_dlt`的计算，讲白了就是用本道次机架的有效凸度减前一道次机架的有效凸度，只不过将本道次机架和前道次机架的包络线和`ef_ex_pu_prf`、`ef_en_pu_prf`联系起来，用于收窄死区。`ef_pu_prf_chg`依据于理论计算，第二项的差值依据于本道次和上道次的包络线，若中浪则取最大的，若边浪则取最小的。

先求有效凸度改变的总量。从出现不均匀变形的机架（`pcPceIZFSPassD`道次的下一道次）开始，如果`pcPceIZFSPassD`道次包络线限幅后的`ef_en_pu_prf`小于原值，说明`pcPceIZFSPassD`道次的包络线区间整体小于原`ef_en_pu_prf`，那么

`ef_pu_prf_sum`累加`ef_pu_prf_dlt[maxl]`。反之，若`pcPceIZFSPassD`道次包络线限幅后的`ef_en_pu_prf`大于原值，说明`pcPceIZFSPassD`道次的包络线区间整体小于原`ef_en_pu_prf`，那么`ef_pu_prf_sum`累加`ef_pu_prf_dlt[minl]`。

注意`pcFSPassD`是本道次的前一道次，而在修正`ef_en_pu_prf`的循环当中，`pcBufFSPassD`指向本道次的前一道次时，循环结束。

这时`ef_en_pu_prf`变量的含义又发生了改变，变回了字面意思，即本道次的入口有效单位凸度。修正是从原始初设定的入口有效单位凸度加上（或减去）与累加有效凸度改变总量成比例的一部分`ef_pu_prf_dlt`，作为新的入口有效单位凸度存在。

```
// 以边浪情形为例
ef_en_pu_prf =
    ef_ex_pu_prf - ef_pu_prf_dlt[ maxl ] *
    ( ef_ex_pu_prf - ef_en_pu_prf ) /
    ef_pu_prf_sum;
```

修正结束后，考虑`ef_en_pu_prf`和`ef_ex_pu_prf`偏差太大的情况，则计算`ef_en_pu_prf_dft`作为最终的`ef_en_pu_prf`。

```
float ef_en_pu_prf_dft = ef_ex_pu_prf + (pcTargtD->en_pu_prf -
pcTargtD->pu_prf)
    * pcFSPassD->pcAlcD->pu_prf_change;
```

最后用`pcTargtD->Eval_Ef_En_PU_Prf(..)`评估一下`ef_en_pu_prf`和`ef_ex_pu_prf`。

## 后续其它计算

若轧制力不重新分配，则预设所有机架`force_ssu`为0。

# 板形评估

板形评估的过程主要在cShapeSetupD::Evaluate(..)函数中执行。

## 评估准备工作

板形评估的作用是用来评估带钢的浪形是否满足要求。因此会设定flt\_ok指示器。

```
pcLPceD->flt_ok = true;
```

之后更新横向带钢模型，获得最新的应变释放系数、带钢弹性模量以及最新的屈曲判别极限。

```
pcLPceD->Update( pcFSStdD->pcExPceD->temp_avg,
                    pcFSStdD->pcExPceD->width,
                    thick,
                    pcFSStdD->pcExPceD->tension );
```

如果当前机架非空过，则计算当前机架轧制力相对于出口带钢厚度的偏导数，供后面横向辊缝模型参数的更新使用。

```
if ( !pcFSStdD->pcRollbite->Calculate_DForce_DExthick( 0.01F ) )

{
    EMSG << "Evaluate: PID="
    << pcFSStdD->pcExPceD->pcPce->prod_id
    << ", invalid ROLLBITE dforce_dexgag"
    << END_OF_MESSAGE;
}
```

横向辊缝模型参数的更新。

```

pcFSStdD->pcLRGD->Update( pcFSStdD->dummied,
                                pcFSStdD->pcExPceD->width,
                                pcFSStdD->pcEnPceD->thick,
                                thick,
                                force_pu_wid,
                                pcFSStdD->pcEnPceD->pcPce->family,
                                pcFSStdD->pcRollbite->DForce_DExt
hick(),
                                pcFSStdD->fs,
                                pcFSStdD->arcon,
                                pcLPceD );

```

之后利用cCRLCD::Crns(..)计算带钢-工作辊和工作辊-支承辊的辊系凸度。

```

pcFSStdD->pcCRLCD->Crns ( pcFSStdD->wr_shft,
                                pcFSStdD->angl_pc,
                                pcFSStdD->pcCRLCD->pce_wr_cr,
                                pcFSStdD->pcCRLCD->wr_br_cr);

```

## 单位凸度和浪形的计算

准备工作结束后，利用pcUFDD->Prf(..)计算pcLPceD->ufd\_pu\_prf。

```

pcLPceD->ufd_pu_prf =
    pcFSStdD->pcUFDD->Prf ( force_pu_wid,
                                force_bnd,
                                pcFSStdD->pcCRLCD->pce_wr_cr
    ,
                                pcFSStdD->pcCRLCD->wr_br_cr
) / thick;

```

用上一个机架的有效单位凸度和当前机架的UFD均载辊缝凸度，计算出口应变差。

```
pcLPceD->strn = pcFSStdD->pcLRGD->Std_Ex_Strn1( pcPrvLP  
ceD->ef_pu_prf,  
                                         pcLPceD  
->ufd_pu_prf );
```

如果应变差超出了屈曲判别的极限，且弯辊力设定值还可以修改，则根据当前的应变差重新计算UFD均载辊缝单位凸度，并更新弯辊力的值，最后再次更新应变差。

这是如果应变差还是不满足屈曲判别的要求，那么修正带钢的凸度改变削弱因子。

```
pcFSStdD->pcLRGD->Cor_Prf_Chng_Attn_Fac_Sup (  
    bckl_lim,  
    pcLPceD->strn );
```

最后更新出口应变差和单位有效凸度，并根据前两者计算新的凸度prf。

# 板形物理模型

板形模型的精度，需要一些板形理论参数支撑。比如带钢影响系数、应变释放系数等，本文进行简要说明。

## 带钢影响系数

金属的横向流动对板形和浪形的影响都很大。模型在这里运用了“带钢影响系数”来表征这一影响因素。

带钢影响系数反映了入口比例凸度对出口比例凸度的影响，取值在0到1之间。如果带钢影响系数取值为0，说明带钢板形的变化不会造成带钢的延伸，如果带钢影响系数取值为1，说明带钢板形的变化完全影响了带钢的延伸，并且带钢在横向没有出现重新分配。

## 凸度削弱因子

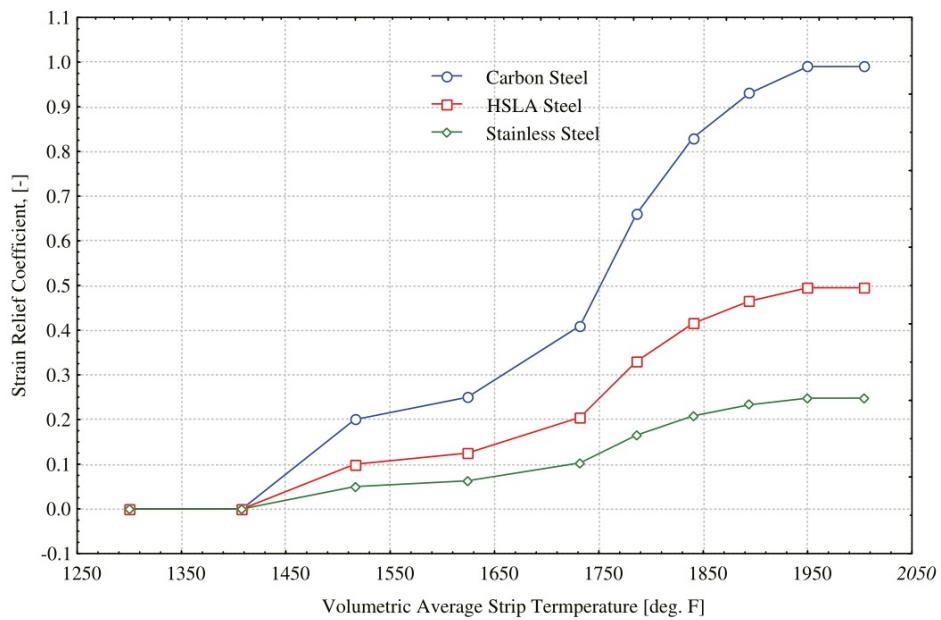
板形的变化会对自身的板形有抑制作用。如果带钢的比例凸度增加，边部的带钢的压下率要明显高于中部带钢的压下率。因此，更高的轧制力导致了轧辊出现更大的压扁和偏斜，进而导致了边部的带钢更厚，从而对比例凸度造成抑制。在这里，这种现象和带钢的硬度紧密相关联。同时，带钢的比例凸度增加，会导致带钢在宽度方向上出现更不均匀的纵向延伸，边部的带钢受更大的压应力。因此，在带钢边部分布的单位轧制力更大，更高的轧制力导致了轧辊出现更大的压扁和偏斜，进而导致了边部的带钢更厚，从而对比例凸度造成抑制。在这里，这种现象和带钢所受的张力紧密相关联。针对以上现象，板形理论用“凸度削弱因子”来表征这种特性。在板形设定模型的评估阶段，如果修改弯辊力也无法满足临界浪形的要求，则重新计算凸度削弱因子，以改变双边浪和中浪的最大调整量。

## 应变释放系数

应变恢复系数用来描述带钢的塑性恢复和机架间带钢的二次变形。应变恢复系数是个经验系数，它和带钢温度和化学成分有关，在板形设定模型当中，这个值通过带钢机架出口温度插值计算获得。

应变释放系数的取值在0和1之间。取值0说明应变不会释放，取值1说明应变完全释放。应变的释放与否和带钢的速度有很大关系。因为带钢的速度的背后，是轧制过程时间、带钢温度和化学成分对应变释放的影响。

如下图所示是不同的钢种（不同的化学成分）对带钢应变释放系数的影响。一般来说，不锈钢相对于碳钢需要更长的时间进行应变恢复。



上一机架出口的带钢板形，到下一机架的入口前发生应变恢复。上一机架出口的应变差一般不等于下一机架入口的应变差，二者之间需要应变释放系数进行修正，如下面公式所示。其中 $f_{strn_r,lf}$ 为应变释放系数。

$$\epsilon_{entry}^{i+1} = (1 - f_{strn_r,lf}) \cdot \epsilon_{exit}^i$$

## 板形恢复系数

由于应变恢复，带钢的板形会出现一些微小的变化。因此我们用“板形恢复系数”来表征应变释放对板形变化的影响大小。因此各个机架的单位凸度改变量如下公式所示。

$$\Delta Cp^i = f_{prf\_recv} \cdot \Delta \epsilon_{exit}^i = f_{prf\_recv} \cdot f_{strn\_rlf} \cdot \epsilon_{exit}^i$$

下一机架入口的单位凸度如下公式所示。

$$Cp_{entry}^{i+1} = Cp_{exit}^i - \Delta Cp^i = Cp_{exit}^i - f_{prf\_recv} \cdot f_{strn\_rlf} \cdot \epsilon_{exit}^i$$

## 偏导数参数或增益

在板形模型的中间计算结果当中会涉及到一些偏导数参数，这些参数反映了两个板形物理量之间的变化关系，或者说增益。

### 弯辊力增益

弯辊力主要的偏导数计算在cShapeSetupD::Xfer\_Functions(..)函数中完成。这个函数中计算了以下几个参数对之间的增益。

- 弯辊力对轧制力的增益
- 弯辊力对窜辊位置的增益
- 弯辊力对UFD均载辊缝凸度的增益
- 弯辊力对机架出口应变差的增益
- 弯辊力对带钢-工作辊辊系凸度的增益
- 带钢-工作辊辊系凸度对工作辊-支承辊辊系凸度的增益

### UFD凸度增益

UFD凸度增益分为以下四个方面。

- UFD凸度对弯辊力的增益
- UFD凸度对轧制力的增益
- UFD凸度对带钢-工作辊辊系凸度的增益
- UFD凸度对工作辊-支承辊辊系凸度的增益

这四个增益以系数（mult）的形式参与和UFD模块有关的计算。



# 凸度分配和板形的关系

热轧七机架连轧，从中间坯到F7出口成品，精轧模型确定了各个机架的负荷分配，板形模型在精轧模型的基础上确定了各个机架的凸度分配。凸度分配的策略是板形设定模型的主旋律。

## 浪形和凸度的关系

凸度分配的好坏直接影响了带钢的浪形，因此带钢的浪形缺陷通过机架间的凸度分配，和带钢凸度之间建立关系。上游机架凸度小同时辊缝大，而下游机架凸度大同时辊缝小，会导致边浪缺陷。上游机架凸度大同时辊缝大，而下游机架凸度小同时辊缝小，会导致中浪缺陷。为了有效描述板形，我们引入了比例凸度的概念，比例凸度指的是当前机架的凸度和当前带钢厚度的比值。

因此，浪形和凸度之间的关系依靠“比例凸度的变化”进行衔接。热连轧前后机架之间比例凸度不同，比例凸度的变化如果在允许的范围内，则不会出现中浪和边浪，但是如果比例凸度的变化超过了允许的范围，则必然产生带钢的浪形。这里所允许的范围被称作死区。如下图所示为死区。

在板形模型中，用应变差（differential）的概念来表示比例凸度变化在浪形死区上的分布情况。比如在某一机架出口，此处的应变差指的是出口的比例凸度和入口有效比例凸度的差值。什么是“入口有效比例凸度”？也就是入口的比例凸度减去入口的应变差。应变差的变化就是模型对“比例凸度变化”的具体描述。

## 浪形和凸度关系的数学证明

# 辊系凸度计算

辊系凸度是板形模型中比较重要的概念。辊系凸度其实就是将空载情况下轧辊的影响因素全部转化为凸度值，再将各个等效的凸度累加的结果。

辊系凸度分为带钢-工作辊辊系凸度和工作辊-支承辊辊系凸度。以带钢工作辊辊系凸度为例子。

带钢-工作辊辊系凸度 = 磨损 + 热胀 + CVC等效凸度 + 工作辊系自学习 + 工作辊补偿。

如下图所示，运算获得总辊系凸度时，各个影响因素之间是相加的关系。

Composite Roll Stack Crowns																			
Std	Pce	WR	WR	BR	Pce	WR	BR	WR	WR	BR	WR	WR	WR	Dlt	Pce	Dlt	WR	SLFG	SPRP
#	Gap	Gap	Wear	Wear	Thrm	Thrm	Grnd	Grnd	Eqv	Vern	Offs	WR_Gap	BR_Gap	WR	WR	Offs	Offs	Offs	Offs
1	-0.7641	-0.4791	-0.0065	-0.0000	-0.0065	-0.1924	-0.1024	-0.8600	-0.0345	0.0000	0.0000	0.0000	0.0000	0.0048	-0.0055	-0.0075	-0.0075		
2	-0.4295	-0.2192	-0.0093	-0.0000	-0.0093	-0.1923	-0.1023	-0.5224	-0.0345	0.0000	0.0000	0.0000	0.0000	0.0034	-0.0048	-0.0058	-0.0058		
3	-0.1554	-0.0003	-0.0186	-0.0000	-0.0186	-0.1393	-0.1393	-0.2761	-0.0345	0.0000	0.0000	0.0000	0.0000	0.0033	-0.0006	-0.0025	-0.0025		
4	-0.0256	0.1063	-0.0256	-0.0000	-0.0256	0.1686	0.1686	-0.1686	-0.0345	0.0000	0.0000	0.0000	0.0000	0.0033	0.0010	-0.0028	-0.0028		
5	0.1370	0.2398	-0.0354	-0.0000	-0.0354	0.2894	0.2894	-0.0369	-0.0345	0.0000	0.0000	0.0000	0.0000	0.0052	0.0002	-0.0020	-0.0020		
6	0.8108	0.1425	-0.0355	-0.0000	-0.0355	0.2144	0.2144	-0.1681	-0.0345	0.0000	0.0000	0.0000	0.0000	0.0042	0.0002	-0.0016	-0.0016		
7	-0.0125	0.1213	-0.0412	-0.0000	-0.0412	0.2858	0.2858	-0.1771	-0.0345	0.0000	0.0000	0.0000	0.0000	0.0030	0.0011	-0.0016	-0.0016		

Composite Roll Stack Crowns																			
Std	Pce	WR	WR	BR	WR	BR	Pce	WR	WR	BR	WR	WR	WR	Dlt	Pce	Dlt	WR	SLFG	SPRP
#	Gap	Gap	Wear	Wear	Thrm	Thrm	Grnd	Grnd	Eqv	Vern	Offs	WR_Gap	BR_Gap	WR	WR	Offs	Offs	Offs	Offs
1	-0.8299	-0.4583	-0.0181	-0.0000	-0.0256	0.2275	0.2219	-0.7974	-0.2298	0.0000	-0.0200	-0.4223	0.0000	0.0000	0.0000	0.0077	-0.5000	0.4000	-0.4000
2	-0.4948	-0.0712	-0.0319	-0.0000	-0.0527	0.2873	0.2819	-0.4712	-0.2298	0.0000	-0.0200	-0.2582	0.0000	0.0000	0.0000	0.0918	-0.3500	0.2500	-0.2500
3	-0.3075	0.0112	-0.0451	-0.0000	-0.0858	0.2294	0.2154	-0.3656	-0.2298	0.0000	-0.0200	-0.1062	0.0000	0.0000	0.0000	0.0938	-0.2000	0.0000	0.0000
4	-0.2045	0.0760	-0.0663	-0.0000	-0.1149	0.2525	0.2379	-0.3675	-0.2298	0.0000	-0.0200	-0.0833	0.0000	0.0000	0.0000	0.0967	-0.1000	0.1000	0.1000
5	-0.3903	-0.0558	-0.0486	-0.0000	-0.0865	0.1752	0.1669	-0.3936	-0.2298	0.0000	-0.0200	-0.1833	0.0000	0.0000	0.0000	0.0967	-0.2000	0.1000	0.1000
6	-0.2037	-0.1460	-0.0754	-0.0000	-0.0755	0.1637	0.1358	-0.3680	-0.0000	0.0000	-0.0300	-0.0979	0.0000	0.0000	0.0000	0.0979	0.0000	0.0000	0.0000
7	-0.1051	-0.0764	-0.0667	-0.0000	-0.0634	0.2816	0.1565	-0.3680	0.0000	-0.0300	0.0000	-0.1500	0.0000	0.0000	0.0000	0.1500	0.0000	0.0000	0.0000

Composite Roll Stack Crowns																			
Std	Pce	WR	WR	BR	WR	BR	Pce	WR	WR	BR	WR	WR	WR	Dlt	Pce	Dlt	WR	SLFG	SPRP
#	Gap	Gap	Wear	Wear	Thrm	Thrm	Grnd	Grnd	Eqv	Vern	Offs	WR_Gap	BR_Gap	WR	WR	Offs	Offs	Offs	Offs
1	-1.8299	-0.4583	-0.0181	-0.0000	-0.0256	0.2275	0.2219	-0.7970	-0.2298	0.0000	-0.0200	-0.4223	0.0000	0.0000	0.0000	0.0077	-0.5000	-0.4000	-0.4000
2	-0.4948	-0.0712	-0.0319	-0.0000	-0.0527	0.2873	0.2819	-0.4712	-0.2298	0.0000	-0.0200	-0.2582	0.0000	0.0000	0.0000	0.0918	-0.3500	0.2500	-0.2500
3	-0.3075	0.0112	-0.0451	-0.0000	-0.0858	0.2294	0.2154	-0.3656	-0.2298	0.0000	-0.0200	-0.1062	0.0000	0.0000	0.0000	0.0938	-0.2000	0.0000	0.0000
4	-0.2045	0.0760	-0.0663	-0.0000	-0.1149	0.2525	0.2379	-0.3675	-0.2298	0.0000	-0.0200	-0.0833	0.0000	0.0000	0.0000	0.0967	-0.1000	0.1000	0.1000
5	-0.3903	-0.0558	-0.0486	-0.0000	-0.0865	0.1752	0.1669	-0.3936	-0.2298	0.0000	-0.0200	-0.1833	0.0000	0.0000	0.0000	0.0967	-0.2000	0.1000	0.1000
6	-0.2037	-0.1460	-0.0754	-0.0000	-0.0755	0.1637	0.1358	-0.3680	0.0000	0.0000	-0.0300	-0.0979	0.0000	0.0000	0.0000	0.0979	0.0000	0.0000	0.0000
7	-0.1051	-0.0764	-0.0667	-0.0000	-0.0634	0.2816	0.1565	-0.3680	0.0000	-0.0300	0.0000	-0.1500	0.0000	0.0000	0.0000	0.1500	0.0000	0.0000	0.0000

# SSU日志验算

日志验算和说明。

## 开头

### Int H w和Fin H w

--- Profile --- 这一栏当中的“H w”指的是加了凸度自学习量的目标凸度。  
Int H w为初始的prf，Int H w为最终的prf。Int H w的计算过程如下所示。

```
prf_int = (pdi_prf + prf_op_off) * matl_exp_cof + prf_vrn;
```

热膨胀系数可以忽略不计，那么Int H w就是PDI的凸度、操作工补偿的凸度、凸度自学习的和。

Profile	
PDI [mm]: 0.0200	Int H w/ [mm]: 0.0602
Vrn [mm]: 0.0400	Fin H w/ [mm]: 0.0602
Vrn RM [mm]: 0.0150	Del C wov [mm]: 0.0200
Vrn RS [mm]: 0.0000	PU [mm/mm]: 0.0238
Tol LO [mm]: 0.0000	Ef PU [mm/mm]: 0.0000
Tol HI [mm]: 0.0000	Achv [-]: T
Op Ofs [mm]: 0.0000	Dev Lim [mm]: 0.0100
Raw Vrn RM [mm]: 0.0250	
New Vrn RM [mm]: 0.0250	

在板形模型目标初始化阶段，prf\_vrn会被赋值为prf\_vrn\_rm和prf\_vrn\_rs的差（prf\_vrn\_rs常年为零）。从实际数据来看，prf\_vrn\_rm和prf\_vrn\_rs的差，与Vrn还是存在差距的，说明这里的Vrn在实际计算中还会出现变化。（可能存在板形问题？）

## 中间坯凸度插值计算

中间坯的凸度，在模型计算中为插值计算。

## 辊系凸度

### 工作辊辊系凸度pce\_wr\_cr的验算

工作辊辊系凸度pce\_wr\_cr在日志中的标识为Pce WR Gap，如下图所示。

Std #	Pce Gap [mm]	Composite Roll Stack Crowns												WR Offs [mm]
		WR BR	Pce WR Gap [mm]	BR Wear [mm]	WR BR Wear [mm]	Pce WR Thrm [mm]	WR BR Thrm [mm]	WR Grnd [mm]	BR Eqv [mm]	WR Vern [mm]	WR Offs [mm]			
1	-0.4531	-0.0802	0.0000	-0.0000	0.0000	0.0000	0.0000	-0.1534	0.2398	0.0000	-0.0200	-0.2796		
2	-0.3415	-0.0014	0.0000	-0.0000	0.0000	0.0000	0.0000	-0.2020	0.2398	0.0000	-0.0200	-0.1195		
3	-0.2348	0.0740	0.0000	-0.0000	0.0000	0.0000	0.0000	-0.3256	0.2398	0.0000	-0.0200	0.1108		
4	-0.1446	0.1377	0.0000	-0.0000	0.0000	0.0000	0.0000	-0.3053	0.2398	0.0000	-0.0200	0.1806		
5	-0.0143	0.2297	0.0000	-0.0000	0.0000	0.0000	0.0000	-0.1752	0.2398	0.0000	-0.0200	0.1809		
6	-0.3262	-0.2304	0.0000	-0.0000	0.0000	0.0000	0.0000	-0.3600	0.0000	0.0000	-0.0300	0.0638		
7	-0.3309	-0.2337	0.0000	-0.0000	0.0000	0.0000	0.0000	-0.3600	0.0000	0.0000	-0.0300	0.0591		

pce\_wr\_cr由五部分组成，分别是

- 带钢-工作辊磨损pce\_wr\_w\_cr
- 带钢-工作辊热胀pce\_wr\_t\_cr
- 工作辊辊形等效凸度wr\_grn\_cr
- 工作辊凸度自学习wr\_cr\_vrn
- 工作辊凸度补偿wr\_cr\_off

把这五部分加起来就是pce\_wr\_cr的值。

### 工作辊凸度补偿的验算

用长期遗传的工作辊凸度补偿cSLFG\_wr\_crn\_off（对应日志中的SLFG WR Offs）和SPRP中的工作辊凸度补偿系数cSPRP\_wr\_crn\_off\_adj相加，即可获得工作辊凸度补偿wr\_cr\_off。

WR	WR	Dlt	Pce	Dlt	WR	SLFG	SPRP
Vern	Offs	WR_Gap	[mm]	BR_Gap	[mm]	WR	WR
	[mm]					Offs	Offs
-0.0200	-0.2796	0.0000		0.0000		0.1204	-0.4000
-0.0200	-0.1195	0.0000		0.0000		0.1305	-0.2500
-0.0200	0.1108	0.0000		0.0000		0.1108	0.0000
-0.0200	0.1806	0.0000		0.0000		0.0806	0.1000
-0.0200	0.1809	0.0000		0.0000		0.0809	0.1000
-0.0300	0.0638	0.0000		0.0000		0.0638	0.0000
-0.0300	0.0591	0.0000		0.0000		0.0591	0.0000

## 中间坯有效单位凸度

中间坯不存在应变差的概念，中间坯的单位凸度就是中间坯的有效单位凸度。

--- Transfer Bar ---中有每卷带钢的中间坯凸度Prof，由模型插值计算获得；以及中间坯的厚度，R2出口测量获得。可以通过凸度求厚度的商，作为中间坯的有效单位凸度，其值与--- Allocation Results (1st Iter.) ---这一栏中的零道次pass0的EF PU Prf对应。中间坯凸度值与--- Allocation Results (1st Iter.) ---这一栏中的零道次pass0的Prf值对应。

----- Allocation Results (1st Iter.) -----							
Des	Bnd	Ef PU		UFD PU	Std Ex		
#	Frc	Prf	Prf	Prf	Strn		
	[ kN ]	[mm/mm]	[mm]	[mm/mm]	[mm/mm]		
0		0.0063	0.2865				
1	815	0.0214	0.4122	0.0239	0.00017		
2	890	0.0234	0.2100	0.0268	0.00014		
3	890	0.0236	0.1417	0.0272	0.00016		
4	890	0.0238	0.1026	0.0287	0.00018		
5	840	0.0238	0.0814	0.0305	0.00021		
6	315	0.0239	0.0686	0.0297	0.00018		
7	320	0.0239	0.0602	0.0212	-0.00009		

## 单位宽度轧制力的验算

单位宽度轧制力，直接拿轧制力除以带钢精轧宽度，由于受到限幅，一般单位轧制力的值是真正被限幅后的值。注意这里的精轧宽度指的是中间坯在F1入口的宽度，不是订单宽度也不是目标宽度。

所以，单位轧制力包络线的最大值、单位轧制力包络线的最小值，以及--Allocation Requirements (1st Iter.) --这一栏中的单位轧制力分配结果一般三者在各自机架内都相等。

## 带钢弹性模量的计算误差

手动验算的时候发现elas\_mod的计算结果总是与SSU日志中的记录结果存在一定的偏差，偏差精度在100以内。

产生这样偏差的原因是因为插值。弹性模量的插值通过每个机架的温度完成。插值表如下表所示。

avg_pce_tmp_interp_vec	elas_modu_interp_vec
600	138269
650	128069
700	117905
750	107751
800	97589
850	87415
900	77232
950	67054
1000	56909
1050	46829
1100	36863
1500	27067

带钢弹性模量的插值向量从27067到138269MPa，与温度的插值相比已经不在一个数量级。温度一单位改变会造成弹性模量更大的变动，因此出现100MPa以内的波动很正常。

## Adaptation Results

在Adaptation Results这一栏中，注意开头的几个参数。

-----from prf scan----- Adaptation Results						
	Bnd	Pce Wr	WR			Prf
#	CrnErr ok CrnVrn	GapErr ok CrnVrn	ok CrnOff	ok ErrRM	ok	
	[mm]	[mm]	[mm]	[mm]	[mm]	
1	0.0000 T -0.0200	-0.3765 T -0.0200	T -0.4411 T	0.0000 F		
2	0.0000 T -0.0200	-0.2250 T -0.0200	T -0.2733 T	0.0000 F		
3	0.0000 T -0.0200	-0.1377 T -0.0200	T -0.0152 T	0.0000 F		
4	0.0000 T -0.0200	-0.0876 T -0.0200	T 0.0897 T	0.0000 F		
5	0.0099 T -0.0200	-0.0809 T -0.0200	T -0.0092 T	0.0000 F		
6	0.0000 T -0.0300	-0.0611 T -0.0300	T -0.0068 T	0.0000 F		
7	-0.0033 T -0.0300	-0.1539 T -0.0300	T 0.0144 T	0.0000 T		

- Bnd Crn Err对应代码中的stk\_bnd\_err。
- Bnd Crn Vrn对应代码中的stk\_vrn\_bnd。（本质就是wr\_cr\_vrn）
- Pce WrR Gap Err对应代码中的stk\_rep\_err。
- WR Crn Vrn对应代码中的stk\_crn\_vrn。
- WR Crn Off对应代码中的stk\_crn\_off

这些参数均属于pcSFBObsD指向的对象。

# SSU日志 FAQ

## Profile一栏中Vrn RM和Vrn RS是什么？

Profile一栏中Vrn RM和Vrn RS一般情况下指的是长期自学习值。

pcTargtD->pcTargt->prf\_vrn\_sel\_flag默认值为true，在cfg\_targt.txt文件中设定，若此值为false，则Profile一栏中Vrn RM和Vrn RS为短期自学习的psSAMP->prf\_vrn\_rm和psSAMP->prf\_vrn\_rs。

## 为什么弯辊力包络线最大值和最小值相反？

标签max和min指的是ufd有效单位凸度的最大值和最小值，最大的弯辊力会计算获得最小的有效单位凸度，最小的弯辊力会计算获得最大的有效单位凸度。为保持一致性，弯辊力包络线max与min对调。

## wr\_crn\_vrn\_z是什么？

算是凸度自学习的一个初始值。当换辊算不准时restore进行补偿。配置文件里面有，据说很好用。

# CFG模型参数整理

C++代码相关的cfg参数整理。

板形的cfg参数可以分为两大类，一类是SSU模块本身的配置参数，一类的是与FSU共享的配置参数。

与板形计算相关的SSU核心配置参数文件如下。

```
include = SSU$CONFIG:cfg_alc.txt;           ! create and configu
re static ALC object
include = MDS$CONFIG:cfg_fcrlc.txt;          ! create and configu
re CRLC object
include = MDS$CONFIG:cfg_flpce.txt;           ! create and configu
re LPCE object
include = MDS$CONFIG:cfg_flrg.txt;            ! create and configu
re LRG object
include = SSU$CONFIG:cfg_fspass.txt;           ! create and configu
re FSPass objects
include = SSU$CONFIG:cfg_fsstd.txt;            ! create and configu
re static FSSTD object
include = MDS$CONFIG:cfg_fufd.txt;             ! create and configu
re UFD object
include = SSU$CONFIG:cfg_penv.txt;              ! create and configu
re static PENV object
include = SSU$CONFIG:cfg_shapesetup.txt;        ! create and configu
re static SHAPESETUP object
include = SSU$CONFIG:cfg_shapefeedback.txt;     ! create and configu
re static SHAPEFEEDBACK object
include = SSU$CONFIG:cfg_targt.txt;              ! create and configu
re static TARGT object
include = SSU$CONFIG:cfg_sfbobs.txt;             ! create and configu
re static Obs object
```

FSU相关的配置参数如下。

```
include = MDS$CONFIG:cfg_fmill.txt;           !create the remainde
r of static mill object for FM
```

```

include = MDS$CONFIG:cfg_ftmpgrad.txt;           !create and configur
e a tmpgradcfg object
include = FSHARED$CONFIG:cfg_std.txt;           !create the static s
tand objects
include = MDS$CONFIG:cfg_fmtr.txt;              !create the static m
otor objects
include = MDS$CONFIG:cfg_frollbite.txt;          !create the static r
oll bite objects
include = GSM$CONFIG:cfg_sensor.txt;             !create the static s
ensor objects
include = GSM$CONFIG:cfg_fzone.txt;              !create and configur
e the zone objects
include = FSHARED$CONFIG:cfg_map.txt;            !create the static m
ap objects

```

## SSU/cfg\_alc

```

class = cAlc;
cAlc = alc;

//单位轧制力调节系数
frcw_adj_mod      = 0.9;                      ! [-] rolling f
orce adjustment modifier

//分配计算过程中迭代的最大次数
loop_count_lim    = 10;                         ! [-] maximum n
umber of iterations

//单位轧制力最小值
force_pu_wid_mn  = 2.0;                        ! [kN/mm] minim
um rolling force per unit piece width

//目标UFD均载辊缝单位凸度和实际UFD均载辊缝单位凸度的偏差容许范围
ufd_pu_prf_tol   = 0.0001;                     ! [mm/mm] UFD r
oll gap per unit profile tolerance

//是否合理的指示器
vld                = true;                      ! [-] validity
indicator

```

```

//计算窜弯辊过程中，窜辊万滚的计算顺序，一般先计算窜辊，再计算弯辊

    actr_prior      = 3,           ! [-] mechanica
1 actuator priority
                           actrtyp_shift,       ! roll CVC s
hifting system (SMS)          actrtyp_bend,        ! roll bendi
ng system                     actrtyp_none;        ! force fall
through
end;
end;

```



## SSU/cfg\_fcrlc

```

class = cCRLC;
cCRLC = crlc;

//窜辊计算迭代的次数
! [-] Newton-Raphson maximum number of iterations on ro
ll shift position
iter_mx = 15;

//CVC插值CVC等效凸度的向量，从cvc1标签到cvc4标签
! [mm] CVC work roll ground crown vector as a f(CVC pro
file type, CVC roll shift position)
cvc_cr_mat = 44,
            -0.860, -0.634, -0.522, -0.409, -0.296
, -0.183, -0.070, 0.043, 0.156, 0.268, 0.494,
            -0.700, -0.608, -0.494, -0.379, -0.265
, -0.150, -0.035, 0.079, 0.194, 0.308, 0.400,
            -0.700, -0.608, -0.494, -0.379, -0.265
, -0.150, -0.035, 0.079, 0.194, 0.308, 0.400,
            -0.700, -0.608, -0.494, -0.379, -0.265
, -0.150, -0.035, 0.079, 0.194, 0.308, 0.400;

//CVC插值的位置向量
! [mm] CVC roll shift position vector
cvc_shft_vec = 11,

```

```

                -150.00, -100.00, -75.00,      -50.00,   -25
.00,      0.00,    25.00,    50.00,    75.00,   100.00,   150.00;

//带钢-工作辊辊系凸度的最大最小偏差
! [mm] minimum / maximum piece to work roll stack crown
error
pce_wr_cr_er = 2,
-1.1999,
1.1999;

//窜辊迭代计算的带钢-工作辊凸度偏差最大值或容许范围
! [mm] Newton-Raphson piece to work roll stack crown to
tolerance
pce_wr_cr_tol = 0.01;

//每次窜辊迭代计算的窜辊位置变化
! [mm] Newton-Raphson delta roll shift position
pos_shft_dlt = 2.00;

//最小和最大辊系凸度补偿
! [mm] minimum / maximum work roll stack crown offset
wr_cr_of = 2,
-1.20,
1.20;

//最小和最大辊系凸度自学习量
! [mm] minimum / maximum work roll stack crown vernier
wr_cr_vr = 2,
-1.20,
1.20;

//CVC辊缝系数
cvc_gap_coef = 4, 0.000, 0.0000, 0.0000, 0.0000;

!!@2ND-2(MAC005) begin
//轧辊热胀和磨损的系数
pce_wr_wear_mult = 1.00;
wr_br_wear_mult = 1.00;
pce_wr_thrm_mult = 1.00;
wr_br_thrm_mult = 1.00;

//初始cvc宽度

```

```

        cvc_width_nominal= 1275.0; !! Nominal cvc width for a1
calculation[mm]
    //窜辊最大极限
    cvc_Sm          = 150.0; !! Shift position limitaion
    //支承辊是否是cvc的指示器
    br_roll_cvc     = true;
!!@2ND-2(MAC005) end
end;
end;

```

## SSU/cfg\_target

```

class = cTarget;
cTarget = target;
// 平直度自学习当中的平直度偏差极限范围
    flt_err_lim      = 2,                                ! [kN] flatness
errorlimits
                    -250.,                               ! minimum
                    250.;                               ! maximum

// 平直度自学习衰减系数
    flt_vrn_bleed   = 0.9;                            ! [-] target fla
tness vernierbleed-off

// 平直度自学习极限范围
    flt_vrn_lim      = 2,                                ! [kN] target fl
atnessvernier limits
                    -800.0,                               ! minimum
                    800.0;                               ! maximum

// 平直度自学习PI控制系数
    flt_vrn_i_gn    = 0.6;                            ! [-] target flat
ness controlloop integral gain
    flt_vrn_p_gn    = 0.3;                            ! [-] target fla
tness controlloop proportional gain

// 凸度波动极限
    prf_dev_lim     = 0.010;                           ! [mm] target pr
ofile deviationlimit

```

```

// 凸度自学习偏差极限
prf_err_lim      = 2,                      ! [mm] profile error limits
error limits
                    -0.100,                  ! minimum
                    0.100;                  ! maximum

// 凸度最大范围
prf_lim          = 2,                      ! [mm] absolute limits
limits
                    0.000,                  ! minimum
                    0.250;                  ! maximum

// 凸度精度要求
prf_tol           = 2,                      ! [mm] target profile tolerances
ofiletolerances
                    -0.050,                  ! minimum
                    0.050;                  ! maximum

// 凸度自学习衰减系数
prf_vrn_bled     = 0.9;                     ! [-] target profile vernier (re-predicted - setup) bleed-off

// 凸度自学习极限
prf_vrn_lim       = 2,                      ! [mm] target profile vernier limits
ofilevernier
                    -0.070,                  ! minimum
                    0.070;                  ! maximum

// 凸度自学习rm的PI控制系数
prf_vrn_rm_i_gn = 0.4;
! [-] target profile vernier (re-predicted - measured) control loop integral gain
prf_vrn_rm_p_gn = 0.2;
! [-] target profile vernier (re-predicted - measured) control loop proportional gain

// 凸度自学习rs的PI控制系数
prf_vrn_rs_i_gn = 0.2;
! [-] target profile vernier (re-predicted - setup) control loop integral gain

```

```

// 平直度自学习，学习的临界点
flt_err_thrshld = 100;
! [kN] Flatness errorthreshold minimum for flatness feed
back

// 操作工如果调整错误，但仍然进行自学习的弯辊力临界点
opr_mx_wrng_corr = 100;
! [kN] Maximum operator correction in WRONG direction an
d still dof flatness feedback

// APC修正开始的机架
apc_start_std = 1;
! [-] APC Correction startstand

// 是否允许计算出口应变差
en_ex_strn_calc = true;
! [-] Enable exit strain calculation

// 出口应变差对应的最大厚度
ex_strn_thk = 10.00; ! [mm] Exit stra
in match forthickness less than or equal to.

// 是否排除不锈钢钢种
exclude_stainless = false; ! [-] Exclude sta
inless steel

!@(CC087) start
// 凸度自学习，长短期以哪个为主的标识，默认长期自学习
prf_vrn_sel_flag = true; ! [-] Vernier se
lection flag(false=samp, true=slfg)

// 平直度自学习，长短期以哪个为主的标识，默认长期自学习
flt_vrn_sel_flag = true; ! [-] Vernier se
lection flag(false=samp, true=slfg)
!@(CC087) end

!@2ND(LC060) start
// 工作辊凸度补偿，标识是否使用sprp数据，默认使用长期自学习和sprp

wr_crn_off_sel_flag = true; ! [-] Work roll of
fset selection flat (false= slfg, true = slfg+sprp)
!@2ND(LC061) end

```

```
    end;  
end;
```



# C-Tool 板形GSM

和板形控制有关的C-Tool参数。

## GSM调节参数总览

GSM的C-TOOL表中的参数如下表所示。

参数名	所在CTOOL的表（忽略前缀）	初始参数变量名
UFD调节值	UFD_Stand(Tuning)Multiplier.xlsb	psSPRP->ufd_mult
初始弯辊力	NormalBendForce.xlsb	psSPRP->bend_nom
窜辊机构最小软极限调节值	ShiftActuatorLimit.xlsb	psSPRP->min_accu_lmt
窜辊机构最大软极限调节值	ShiftActuatorLimit.xlsb	psSPRP->max_accu_lmt
弯辊机构最小软极限调节值	BendingActuatorLimit.xlsb	psSPRP->min_bend_lmt
弯辊机构最大软极限调节值	BendingActuatorLimit.xlsb	psSPRP->max_bend_lmt
边浪调节因子	WavinessTuning.xlsb	psSPRP->wav_mult
边浪补偿值	WavinessTuning.xlsb	psSPRP->wav_ofs
中浪调节因子	CenterBuckleTuning.xlsb	psSPRP->cb_mult
中浪补偿值	CenterBuckleTuning.xlsb	psSPRP->cb_ofs
综合凸度补偿系数	WorkCrownOffsetTuning.xlsb	psSLFG->wr_crn_off

# UFD multiplier调节值

## 调节参数在表中的位置

UFD凸度multiplier调节值位于GSM的UFD\_Stand(Tuning)Multiplier.xlsb当中。按钢种组分类，针对每个宽度区间和厚度区间、每个机架，有一个UFD凸度multiplier调节值，供工艺人员进行调节。

如下图所示。

Width index	Gauge index	F1	F2	F3	F4	F5	F6	F7
		ufd-mult	ufd-mult	ufd-mult	ufd-mult	ufd-mult	ufd-mult	ufd-mult
0	800<= w < 900	0	1.00<=h<1.15	1	1	1	1	1
	1	1.15<=h<1.30	1	1	1	1	1	1
	2	1.30<=h<1.50	1	1	1	1	1	1
	3	1.50<=h<1.70	1	1	1	1	1	1
	4	1.70<=h<1.85	1	1	1	1	1	1
	5	1.85<=h<2.05	1	1	1	1	1	1
	6	2.05<=h<2.25	1	1	1	1	1	1
	7	2.25<=h<2.50	1	1	1	1	1	1
	8	2.50<=h<2.75	1	1	1	1	1	1
	9	2.75<=h<3.00	1	1	1	1	1	1
	10	3.00<=h<3.40	1	1	1	1	1	1
	11	3.40<=h<4.00	1	1	1	1	1	1
	12	4.00<=h<5.00	1	1	1	1	1	1
	13	5.00<=h<6.00	1	1	1	1	1	1
	14	6.00<=h<7.50	1	1	1	1	1	1
	15	7.50<=h<9.00	1	1	1	1	1	1
	16	9.00<=h<10.50	1	1	1	1	1	1
	17	10.50<=h<11.50	1	1	1	1	1	1
	18	11.50<=h<12.70	1	1	1	1	1	1
	19	12.70<=h<14.00	1	1	1	1	1	1
	20	(spare)	1	1	1	1	1	1
	21	(spare)	1	1	1	1	1	1
	22	(spare)	1	1	1	1	1	1
	23	(spare)	1	1	1	1	1	1
	24	(spare)	1	1	1	1	1	1
	25	(spare)	1	1	1	1	1	1
	26	(spare)	1	1	1	1	1	1
	27	(spare)	1	1	1	1	1	1

## UFD调节参数的作用

这个参数的作用是对UFD凸度的计算进行补偿和修正。同时影响到其它所有涉及到UFD凸度的工艺量计算，或者说影响到其它以UFD凸度为参数的函数，如单位宽度轧制力、弯辊力、带钢-工作辊凸度的计算等。

## UFD调节参数参与的过程

### 初始化

首先在cShapeSetupD::Init()的初始化过程，从sSPRP结构中将此参数映射的变量ufd\_mult赋值给pcFSSstdD对象中的变量ufd\_mult（两者同名）。

在cShapeSetupD::Init()初始化过程中，有一个cUFDD::Init()函数，pcFSSstdD对象的ufd\_mult作为最后一个参数变量传入，对相关的cUFDD对象进行了初始化。pcFSSstdD对象的ufd\_mult被赋值给了cUFDD对象的ufd\_modifier。ufd\_modifier是cUFDD对象的一个属性。之后的各种计算，此参数均以ufd\_modifier的变量形式进行参与。

## UFD调节参数参与的函数总览

cUFDD::Init()初始化之后，ufd\_modifier参与计算的函数如下表所示。

受参数影响的函数	函数作用
cUFDD::Prf	计算辊缝凸度
cUFDD::Bnd_Frc	计算弯辊力
cUFDD::Frc_PU_Wid	计算单位宽度轧制力
cUFDD::Pce_WR_Crn	单独计算带钢-工作辊凸度
cUFDD::Crns	计算综合轧辊凸度

### cUFDD::Prf

cUFDD::Prf此函数是用来计算UFD辊缝凸度的函数，输入量为单位宽度轧制力、弯辊力、带钢-工作辊凸度、工作辊-支撑辊凸度。在函数中，ufd\_modifier介入的地方如下所示。

```
float cUFDD::Prf(
    const float force_pu_wid, // [kn/mm_mtton/
    mm_eton/in] rolling // force per
    unit piece width
    const float force_bnd, // [kn_mtton_eto
    n] roll bending force
    const float pce_wr_crn, // [mm_mm_in] p
    iece to work roll // stack cro
```

```

wn
    const float wr_br_crn           // [mm_mm_in] wo
rk roll to backup
                                // roll crown

    //const float ufd_modifier      // [-] UFD (f
amily,grt,wrt)tuning modifier
        ) const                   // [mm_mm_in] U
FD roll gap profile

{   // Begin of PRF function

    return
        ( b_cof[ 0 ] * force_pu_wid +
        b_cof[ 1 ] * pow( force_pu_wid, float(1.5) ) +
        b_cof[ 2 ] * pce_wr_crn +
        b_cof[ 3 ] * wr_br_crn * force_pu_wid +
        b_cof[ 4 ] * wr_br_crn * pow( force_pu_wid, float(1.5)
) +
        b_cof[ 5 ] * force_bnd +
        b_cof[ 6 ] * force_bnd * force_pu_wid +
        b_cof[ 7 ] * force_bnd * pow( force_pu_wid, float(2.0)
) +
        b_cof[ 8 ] * wr_br_crn +
        b_cof[ 9 ] * force_pu_wid +
        b_cof[ 10 ] * force_bnd +
        b_cof[ 11 ] * force_pu_wid +
        b_cof[ 12 ] * pow( force_pu_wid, float(1.5) ) +
        b_cof[ 13 ] * force_bnd +
        b_cof[ 14 ] * pce_wr_crn +
        b_cof[ 15 ] * pce_wr_crn +
        b_cof[ 16 ] +
        b_cof[ 17 ] ) * ufd_modifier;

}   // End of PRF function

```

ufd\_modifier是作为一个乘数，乘到UFD辊缝凸度的返回值上。在包络线生成、分配计算、评估计算等阶段，均运用到了cUFDD::Prf函数。当ufd\_modifier或ufd调整参数增大，cUFDD::Prf的计算结果增大；当ufd\_modifier或ufd调整参数减小，cUFDD::Prf的计算结果减小。

## cUFDD::Bnd\_Frc

cUFDD::Bnd\_Frc此函数用来计算弯辊力，输入量为UFD辊缝凸度、单位宽度轧制力、带钢-工作辊凸度、工作辊-支撑辊凸度、弯辊力软极限（弯辊力软极限的调节系数在这里介入），输出量为计算的最终弯辊力force\_bnd和实际需要的弯辊力force\_bnd\_des。如下所示。

```
void cUFDD::Bnd_Frc(
    const float ufd_prf, // [mm_mm_in] U
    FD roll gap profile
    const float force_pu_wid, // [kn/mm_mton/
    mm_eton/in] rolling // force per
    unit piece width
    const float pce_wr_crn, // [mm_mm_in] p
    piece to work roll // stack cro
    wn
    const float wr_br_crn, // [mm_mm_in] w
    work roll to backup // roll stac
    k crown
    const float force_bnd_lim[2], // [mton_eton_k
    n] roll bending // soft limit
    t (min,max)
    float& force_bnd, // [mton_eton_k
    n] roll bending force
    float& force_bnd_des // [mton_eton_k
    n] desired roll // bending f
    orce
) const

{ // Begin of BND_FRC function

//-----
// Calculate the roll bending force.
//-----
force_bnd_des =
( ufd_prf / ufd_modifier -
```

```

        b_cof[ 0 ] * force_pu_wid -
        b_cof[ 1 ] * pow( force_pu_wid, float(1.5) ) -
        b_cof[ 2 ] * pce_wr_crn -
        b_cof[ 3 ] * wr_br_crn * force_pu_wid -
        b_cof[ 4 ] * wr_br_crn * pow( force_pu_wid, float(1.5)
    ) -
        b_cof[ 8 ] * wr_br_crn -
        b_cof[ 9 ] * force_pu_wid -
        b_cof[ 11 ] * force_pu_wid -
        b_cof[ 12 ] * pow( force_pu_wid, float(1.5) ) -
        b_cof[ 14 ] * pce_wr_crn -
        b_cof[ 15 ] * pce_wr_crn -
        b_cof[ 16 ] -
        b_cof[ 17 ] ) /
( b_cof[ 5 ] +
  b_cof[ 6 ] * force_pu_wid +
  b_cof[ 7 ] * pow( force_pu_wid, float(2.0) ) +
  b_cof[ 10 ] +
  b_cof[ 13 ] );

//-----
// Restrict the roll bending force to within soft limits.
//-----
force_bnd = cMathUty::
    Clamp( force_bnd_des,
            force_bnd_lim[min1],
            force_bnd_lim[max1] );

} // End of BND_FRC function

```



在函数中，首先计算实际需要的弯辊力，如下图所示。在这个过程中，ufd\_modifier作为除数参与弯辊力计算，从UFD辊缝凸度中除去。

```

force_bnd_des =
( ufd_prf / ufd_modifier -
  b_cof[ 0 ] * force_pu_wid -
  b_cof[ 1 ] * pow( force_pu_wid, float(1.5) ) -
  b_cof[ 2 ] * pce_wr_crn -
  b_cof[ 3 ] * wr_br_crn * force_pu_wid -
  b_cof[ 4 ] * wr_br_crn * pow( force_pu_wid, float(1.5)

```

```

) -
    b_cof[ 8 ] * wr_br_crn -
    b_cof[ 9 ] * force_pu_wid -
    b_cof[ 11 ] * force_pu_wid -
    b_cof[ 12 ] * pow( force_pu_wid, float(1.5) ) -
    b_cof[ 14 ] * pce_wr_crn -
    b_cof[ 15 ] * pce_wr_crn -
    b_cof[ 16 ] -
    b_cof[ 17 ] ) /
( b_cof[ 5 ] +
    b_cof[ 6 ] * force_pu_wid +
    b_cof[ 7 ] * pow( force_pu_wid, float(2.0) ) +
    b_cof[ 10 ] +
    b_cof[ 13 ] );

```



## cUFDD::Frc\_PU\_Wid

cUFDD::Frc\_PU\_Wid此函数用于计算单位宽度轧制力，输入量为UFD辊缝凸度、弯辊力、综合轧辊凸度，输出量为单位宽度轧制力。ufd\_modifier在这里参与的是计算过程中被求导函数的常数项的计算。cof\_4计算中，ufd\_modifier从辊缝凸度中除去，对辊缝凸度进行修正。

## cUFDD::Pce\_WR\_Crn

```

float cUFDD::Pce_WR_Crn(
    const float ufd_prf, // [mm_mm_in] U
    FD roll gap profile
    const float force_pu_wid, // [kn/mm_mton/
    mm_eton/in] rolling // force per
    unit piece width
    const float force_bnd, // [kn_mton_eto
    n] roll bending force
    const float wr_br_crn // [mm_mm_in] w
    ork roll to backup // roll stac
    k crown
) const // [mm_mm_in] p

```

```

iece to work roll
                //      stack cro
wn

{   // Begin of PCE_WR_CRN function

    return
        ( ufd_prf / ufd_modifier -
        b_cof[ 0 ] * force_pu_wid -
        b_cof[ 1 ] * pow( force_pu_wid, float(1.5) ) -
        b_cof[ 3 ] * wr_br_crn * force_pu_wid -
        b_cof[ 4 ] * wr_br_crn * pow( force_pu_wid, float(1.5)
    ) -
        b_cof[ 5 ] * force_bnd -
        b_cof[ 6 ] * force_bnd * force_pu_wid -
        b_cof[ 7 ] * force_bnd * pow( force_pu_wid, float(2.0)
    ) -
        b_cof[ 8 ] * wr_br_crn -
        b_cof[ 9 ] * force_pu_wid -
        b_cof[ 10 ] * force_bnd -
        b_cof[ 11 ] * force_pu_wid -
        b_cof[ 12 ] * pow( force_pu_wid, float(1.5) ) -
        b_cof[ 13 ] * force_bnd -
        b_cof[ 16 ] -
        b_cof[ 17 ] ) /
        ( b_cof[ 2 ] +
        b_cof[ 14 ] +
        b_cof[ 15 ] );

}   // End of PCE_WR_CRN function

```

cUFDD::Pce\_WR\_Crn函数在包络线生成和分配计算阶段进行调用，而在分配阶段仅调用一次。

传入的UFD辊缝凸度对应的UFD单位凸度来自cLRGD->UFD\_PU\_Prf3，是利用入口有效凸度、出口有效凸度以及应变释放系数计算的结果，ufd\_modifier是对这个UFD单位凸度对应的辊缝凸度进行修正。

## cUFDD::Crns

cUFDD::Crns是用来同时计算带钢-工作辊凸度和工作辊-支撑辊凸度的。输入量为UFD辊缝凸度、单位宽度轧制力和弯辊力。

在这里ufd\_modifier不直接参与cUFDD::Crns的补偿，但是在分配计算阶段，其输入量弯辊力是利用cUFDD::Bnd\_Frc进行计算的结果。

## 初始弯辊力

### 初始弯辊力在表中的位置

初始弯辊力调节值位于GSM的NormalBendForce.xlsb当中。按钢种组分类，针对每个宽度区间和厚度区间、针对每个道次，有一个初始弯辊力调节值，供工艺人员进行调节。如下图所示。

A	B	C	D	E	F	G	H	I	J	K	
1	Normal Bend Force										
2	Last Get : 1/14/2017 10:43:19 AM										
3	Width index	Gauge index			bend-nom [kN/roll]						
4				F1	F2	F3	F4	F5	F6	F7	
5	0	800<=w < 900	0	1.00<=h<1.15	890	890	890	890	840	840	840
6			1	1.15<=h<1.30	890	890	890	890	840	840	840
7			2	1.30<=h<1.50	890	890	890	890	840	840	840
8			3	1.50<=h<1.70	890	890	890	890	840	840	840
9			4	1.70<=h<1.85	890	890	890	890	840	840	840
10			5	1.85<=h<2.05	890	890	890	890	840	840	840
11			6	2.05<=h<2.25	890	890	890	890	840	840	840
12			7	2.25<=h<2.50	890	890	890	890	840	840	840
13			8	2.50<=h<2.75	890	890	890	890	840	840	840
14			9	2.75<=h<3.00	890	890	890	890	840	840	840
15			10	3.00<=h<3.40	890	890	890	890	840	840	840
16			11	3.40<=h<4.00	890	890	890	890	840	840	840
17			12	4.00<=h<5.00	890	890	890	890	840	840	840
18			13	5.00<=h<6.00	890	890	890	890	840	840	840
19			14	6.00<=h<7.50	890	890	890	890	840	840	840
20			15	7.50<=h<9.00	890	890	890	890	840	840	840
21			16	9.00<=h<10.50	890	890	890	890	840	840	840
22			17	10.50<=h<11.50	890	890	890	890	840	840	840
23			18	11.50<=h<12.70	890	890	890	890	840	840	840
24			19	12.70<=h<14.00	890	890	890	890	840	840	840
25			20	(spare)	890	890	890	890	840	840	840
26			21	(spare)	890	890	890	890	840	840	840
27			22	(spare)	890	890	890	890	840	840	840
28			23	(spare)	890	890	890	890	840	840	840
29			24	(spare)	890	890	890	890	840	840	840
30			25	(spare)	890	890	890	890	840	840	840
31			26	(spare)	890	890	890	890	840	840	840
32			27	(spare)	890	890	890	890	840	840	840
33	1	900 <= w < 1050	0	1.00<=h<1.15	890	890	890	890	840	840	840
34			1	1.15<=h<1.30	890	890	890	890	840	840	840
35			2	1.30<=h<1.50	890	890	890	890	840	840	840

### 初始弯辊力的初始化

首先在cShapeSetupD::Init()的初始化过程，从sSPRP结构中将此参数psSPRP->bend\_nom[passIdx]按制定的道次赋值给pcFSStd对象中的变量pcFSPassD->pcFSStdD[ iter ]->pcFSStd->force\_bnd\_nom。

在cfg\_fsstd.txt中，前6道次机架的force\_bnd\_nom为1500kN，第七机架为1000kN。

## 初始弯辊力的作用

### PENV

在包络线计算过程中，force\_bnd\_nom主要参与UFD辊缝凸度对单位轧制力偏导数的计算。作为第二个参数传入。

```
pcFSPassD->pcPEnvD->dprf_dfrcw =
    pcFSPassD->pcFSStdD[ iter ]->pcUFDD->Dprf_D
    frcw (
        pcFSPassD->pcPEnvD->force_pu_wid
        ,
        pcFSPassD->pcFSStdD[ iter ]->pc
    FSStd->force_bnd_nom,
        pce_wr_crn,
        wr_br_crn
    );
```

## 分配计算

在分配计算阶段，force\_bnd\_nom参与SSU轧制力的计算。

## 窜辊机构软极限调节值

### 调节值位置

窜辊机构软极限调节值位于GSM的ShiftActuatorLimit.xlsb当中。按钢种组分类，针对每个宽度区间和厚度区间、针对每个道次，有一组最大和最小窜辊软极限调节值，供工艺人员进行调节。

## 窜辊机构软极限调节值的作用

在SSU初始设定阶段，窜辊机构软极限调节值主要是用来作为乘数，乘到窜辊硬极限上，用于修正和约束窜辊软极限。

## 窜辊调节值参与过程

首先利用窜辊速度和卷卷带钢之间的间隙时间计算窜辊步长的最大变化量。默认的窜辊步长最大变化量为100mm，与计算值比较并取最小值。如果磨损存在异常，则从新按磨损异常的规则确立最大窜辊步长。之后计算窜辊的位置极限，原窜辊位置加上最大最小窜辊步长变化量求窜辊位置的软极限。

## 弯辊机构软极限调节值

### 调节值位置

弯辊机构软极限调节值位于BendingActuatorLimit.xlsb中，针对每个钢种族，不同宽度与不同厚度、不同机架分别有最大最小两个调节值。

### 调节值作用

psSPRP->min\_bend\_lmt参与计算弯辊力软极限。

```
pcFSStdDloc->force_bnd_lim[ i ] = pcFSStdDloc->force_bnd_lim_org  
[ i ] =  
    pcFSStdDloc->pcFSStd->force_bnd_lim[ i ] *  
    psSPRP->min_bend_lmt[ pcFSStdDloc->pcFSStd->num-1 ];
```

## CenterBuckleTuning和WavinessTuning

CTOOL中GSM模块中有CenterBuckleTuning和CenterBuckleTuning两张表，这两张表主要用来对屈曲判别标准的中浪和双边浪极限值进行调整。个人理解是用来调整在带钢约束条件下的死区极限范围。

这两个表对于每个机架F1到F7分别有两个参数。一个是比例系数multiplier，作为乘数而存在，另一个是补偿值Offset，作为加数而存在。

在模型代码中，用到参数表的地方主要在LPCE模块当中。

参数表中的中浪和双边浪调节系数作为函数的参数参与LPce对象的初始化过程cLPceD::Init()，保存在bckl\_mul和bckl\_off中。参数表中的数据

在cLPceD::Crit\_Bckl\_Lim中参与屈曲极限值的计算。首先建立屈曲极限的缓冲区并初始化为0值。中浪和双边浪的屈曲极限的原极限计算值是在cLPceD::Crit\_Bckl()当中完成的，并存储于bckl\_lim。如下图所示。需要用到带钢的宽度、厚度、带钢所受的机架张力和杨氏模量等进行计算。

之后在cLPceD::Crit\_Bckl\_Lim中用调节系数进行修正，multiplier作为乘数，offset作为加数。值得注意的是，补偿计算极限值有一个约束条件，那就是在补偿计算后，双边浪的极限值必须大于中浪的极限值，也就是说必须保证死区的存在，否则返回中浪极限值和双边浪极限值的均值。如下图中的if条件分支所示。

## 工作辊综合凸度补偿

综合凸度补偿的调节在WorkCrownOffsetTuning.xlsb当中，初始变量为psSLFG->wr\_crn\_off[ pass\_idx ]，主要参与综合凸度的计算。

Width index	Gauge index	F1	F2	F3	F4	F5	F6	F7
		wr_crn_off_adj	wr_crn_off_adj	wr_crn_off_adj	wr_crn_off_adj	wr_crn_off_adj	wr_crn_off_adj	wr_crn_off_adj
0	800<= w < 900	0 1.00<=h<1.15 1.15<=h<1.30 1.30<=h<1.50 1.50<=h<1.70 1.70<=h<1.85 1.85<=h<2.05 2.05<=h<2.25 2.25<=h<2.50 2.50<=h<2.75 2.75<=h<3.00 3.00<=h<3.40 3.40<=h<4.00 4.00<=h<5.00 5.00<=h<6.00 6.00<=h<7.50 7.50<=h<9.00 9.00<=h<10.50 10.50<=h<11.50 11.50<=h<12.70 12.70<=h<14.00 (spare) (spare) (spare) (spare) (spare) (spare) (spare)	-0.4 -0.4 -0.4 -0.4 -0.4 -0.4 -0.4 -0.4 -0.4 -0.4 -0.4 -0.4 -0.4 -0.4 -0.4 -0.4 -0.4 0 0 0 0 0 0 0	-0.25 0	0 0.1 0	0.1 0	0 0	0 0
	900 <= w < 1050	0 1.00<=h<1.15 1.15<=h<1.30	-0.4 -0.4	-0.25 -0.25	0 0	0.1 0.1	0.1 0.1	0 0

在ShapeSetup初始化过程中，综合凸度的补偿值作为输入参数传入cCRLCD::Init()进行初始化。

```

pcFSPassD->pcFSStdD[ seg ]->pcCRLCD->Init(
    inhb_t_w_calc,
    pce_wr_t_w_crn,
    wr_br_t_w_crn,
    psSAMP->wr_crn_vrn[ pass_idx ],
//@2ND-2(MAC014) begin
    // ( psSLFG->wr_crn_off[ pass_idx ] + psSPRP-
>wr_crn_off_adj[ pass_idx ] ),
    ( psSLFG->wr_crn_off[ pass_idx ] + f_wr_crn_
off_adj ),
//@2ND-2(MAC014) end
    pcFSPassD->pcFSStdD[ seg ]->pcEnPceD->width,
    pcFSPassD->pcFSPass->wr_crn_vrn_i_gn,
    pcFSPassD->pcFSPass->wr_crn_off_i_gn,
    pcFSPassD->pcFSPass->wr_crn_cor_i_gn,
    pcFSPassD->pcFSStdD[ seg ]->pcStdRollPrD
//psPDI->grt_idx
);

```



# 板形模型实战

模型的掌握和使用，万万不可脱离实战。

# 目标凸度变化影响板形模型设定

## 事件经过

2018年5月15日，15:15，丙（白）班，MRTLA42005钢种，卷号M18057558H，规格3.2\*1172mm。带钢穿带过程中F1-F4机架跑偏操作侧严重，带钢头部至F6-7机架时瞬间跑偏传动侧，操作侧起浪，操作调整F6-7调平，随即带钢反向跑偏后起套堆钢。

从视频回放来看，存在两方面原因：第一，F1-F7机架跑偏严重；第二，F6-7出现反向跑偏。带钢跑偏严重主要是由于窜辊量变化较大引起。窜辊变化的原因主要为目标凸度剧变造成整个板形模型凸度分配异常，具体影响过程详见下文。

## 轧制力分析

轧制力方面如下。

机架	FBK/KN	MEAS/KN	偏差
F1	25292	22984	-9.13%
F2	21368	21824	2.13%
F3	19981	19268	-3.57%
F4	17007	17273	1.56%
F5	15672	15443	-1.46%
F6	12749	12483	-2.09%
F7	13318	13300	-0.14%

F1轧制力预报出现偏差，但是在10%以内，属于正常范围，其余预报正常；但是F7轧制力大于F6轧制力，属于异常情况，查询发现F7压下率锁在13%。按照轧制力分配原则，F5-F7机架轧制力应按照等比例分配，保证板形稳定，但是F7轧制力分配明显异常。

## 窜辊和板形所受影响

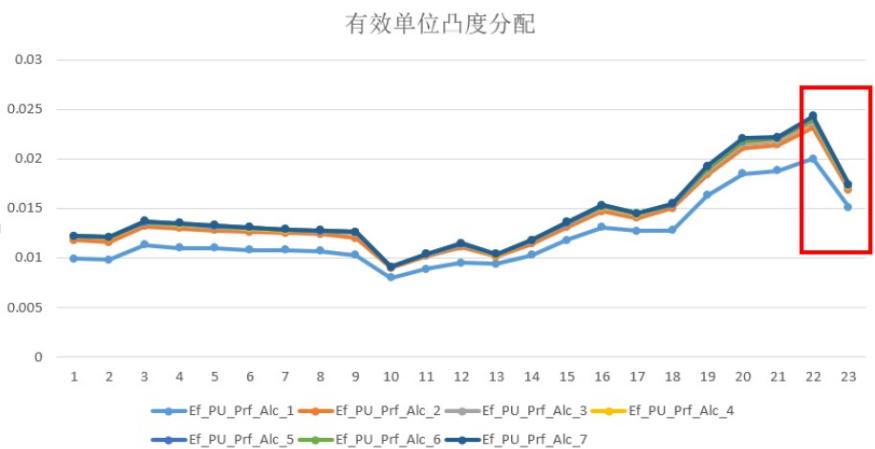
从开轧至堆钢的窜辊情况如下。

卷号	F1	F2	F3	F4	F5	F6	F7
M18057536H	36.6	10.2	-6.3	-2.3	-30.0	-0.0	0.0
M18057537C	28.7	0.3	-12.5	-7.3	-60.0	-0.0	0.0
M18057538C	13.2	-10.8	-29.3	-20.6	-90.0	0.0	24.3
M18057539C	12.4	-12.9	-30.8	-21.3	-120.0	0.0	55.5
M18057540C	14.4	-9.3	-29.0	-21.4	-133.1	0.0	83.9
M18057541C	15.2	-8.8	-29.8	-21.7	-131.1	24.3	108.1
M18057542C	14.9	-10.1	-30.3	-23.6	-133.3	55.5	126.9
M18057543C	14.3	-9.7	-31.5	-25.2	-133.4	83.9	139.3
M18057544C	31.8	-0.0	0.0	-0.0	-0.0	-0.0	-0.0
M18057545M	79.5	41.8	33.8	30.0	-7.4	126.9	140.0
M18057546M	63.6	23.4	19.8	38.6	-25.2	139.3	134.0
M18057547C	49.3	11.2	9.9	29.7	-38.1	140.0	94.7
M18057548C	81.3	32.4	22.3	9.8	-31.2	140.0	77.4
M18057549C	61.3	15.8	11.7	1.2	-38.4	134.0	56.2
M18057550C	41.1	-1.5	-2.3	-9.6	-53.8	94.7	32.3
M18057551C	25.5	-10.3	-5.7	-9.9	-48.5	77.4	6.7
M18057552C	50.4	13.0	2.4	-4.1	-37.2	56.2	-19.2
M18057553H	25.3	5.8	-9.6	-9.5	-69.2	32.3	-44.2
M18057554H	-4.5	-12.4	-22.9	-21.0	-92.7	6.7	-67.0
M18057555P	-25.7	-37.4	-35.1	1.4	-57.3	-19.2	-86.3
M18057556P	-32.8	-47.1	-41.6	-3.2	-50.5	-44.2	-101.4
M18057557H	-87.1	-83.4	-65.7	-48.8	-109.6	-66.9	-111.4
M18057558H]	13.1	-15.4	-22.7	10.0	-38.7	-86.3	-115.8

M18057557H到M18057558H窜辊变化剧烈。M18057558H窜辊剧烈变化的原因如下：M18057558H这卷带钢的CVC设定值达到窜辊设定软极限，如下图所示。

WR Shft	Min [mm]	Nom [mm]	Max [mm]
-140.60	0.00	13.06	
-133.39	0.00	-15.39	
-105.73	0.00	-22.73	
-78.79	0.00	10.09	
-139.58	0.00	-38.72	
-86.34	0.00	-86.34	
-115.80	0.00	-115.80	

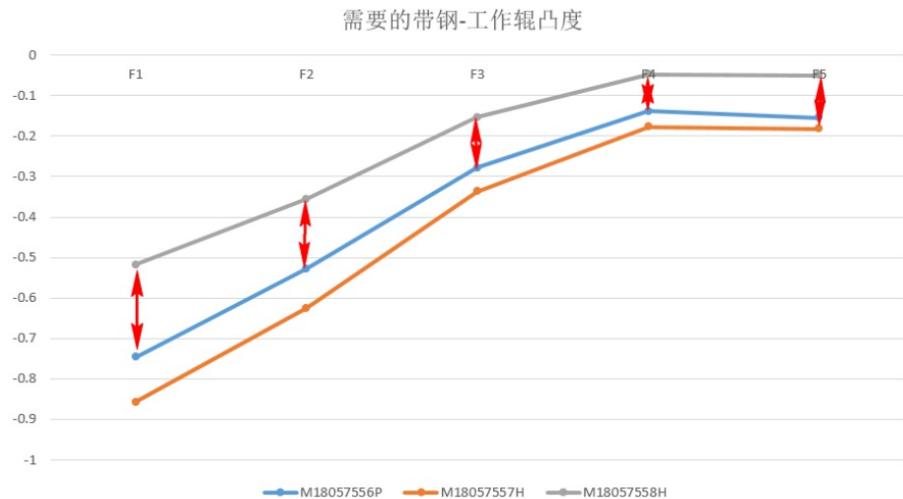
以F1为例，M18057557H带钢F1的CVC窜辊为-87.1mm，根据窜辊限幅的计算，F1机架的窜辊最大步长为100mm，因此下一卷带钢M18057558H的窜辊限幅的最大值为+13.1mm。但是，实际需要的窜辊变化量比窜辊限幅最大值还要大，因此在计算板形设定时，窜辊值被限制在软极限。造成实际窜辊变化量远超窜辊限幅最大值的原因，是因为凸度分配计算中，同规格同钢种的各个机架出入口有效单位凸度变化大。



有效单位凸度在分配计算中是从F7往前计算，F7出口依赖目标凸度作为输入，M18057557H目标凸度为50  $\mu\text{m}$ ，M18057558H目标凸度为30  $\mu\text{m}$ ，同钢种同规格，50  $\mu\text{m}$ 到30  $\mu\text{m}$ 为一个天一个地。因此造成板形模型分配计算中，从F7

到F1推算过程中，F7的初始计算存在较大偏差。M18057557H的F7出口有效单位凸度为 $0.024 \mu\text{m/mm}$ ，而M18057558H的F7出口有效单位凸度为 $0.0174 \mu\text{m/mm}$ ，相差较大。

有效单位凸度偏差较大，导致计算的所需的带钢-工作辊凸度偏差也较大，变化量能达到50%以上。



卷号	F1	F2	F3	F4	F5
M18057556P	-0.74584	-0.52818	-0.27803	-0.13864	-0.155366
M18057557H	-0.85769	-0.62475	-0.33696	-0.17793	-0.182116
M18057558H	-0.51785	-0.35594	-0.15402	-0.04754	-0.0495497

所需的带钢-工作辊凸度是窜辊迭代计算的直接输入量，M18057558H和M18057557H相比，所需的带钢-工作辊凸度减小1倍以上，所分配设定的窜辊值必然向正方向移动明显。以上解释了为什么目标凸度的变化影响了整个板形模型分配的计算。

## 解决措施

解决措施很简单，梳理各钢种PDI目标凸度，规范同钢种凸度目标值设定。

