

Opérateur DEFI_CONTACT

1 But

Affecter des conditions de contact unilatéral et de frottement en mécanique ou des conditions unilatérales sur les autres degrés de liberté.

Table des matières

1 But.....	1
2 Syntaxe.....	4
3 Principes.....	12
3.1 Opérations d'appariement (méthodes maillées hors XFEM).....	13
3.1.1 Opérande APPARIEMENT.....	13
3.1.2 Opérandes MAILLE_MAIT/GROUP_MA_MAIT/MAILLE_ESCL/GROUP_MA_ESCL.....	13
3.1.3 Opérandes SANS_NOEUD/SANS_GROUP_NO.....	14
3.1.4 Opérandes TYPE_APPA/DIRE_APPA.....	14
3.1.5 Opérandes TOLE_APPA et TOLE_PROJ_EXT.....	14
3.1.6 Choix des normales.....	15
3.1.6.1 Type de normale (NORMALE).....	15
3.1.6.2 Choix des normales maître ou esclave (VECT_MAIT/VECT_ESCL).....	15
3.1.6.3 Lissage des normales (LISSAGE).....	16
3.1.7 Modification du jeu.....	16
3.1.7.1 Opérandes DIST_MAIT/DIST_ESCL.....	17
3.1.7.2 Opérandes DIST_POUTRE/DIST_COQUE/CARA_ELEM.....	17
3.2 Opérations d'appariement spécifiques à la formulation CONTINUE.....	17
3.2.1 Opérandes RACCORD_LINE_QUAD/GROUP_NO_RACC/NOEUD_RACC	17
3.2.2 Opérandes FOND_FISSURE/GROUP_NO_FOND/NOEUD_FOND/GROUP_MA_FOND/ MAILLE_FOND	17
3.2.3 Opérandes SANS_NOEUD_FR/SANS_GROUP_NO_FR	18
3.2.4 Opérande EXCLUSION_PIV_NUL	18
3.3 Choix des méthodes de résolution.....	18
3.3.1 Frottement.....	18
3.3.2 Choix de la méthode pour le contact en formulation DISCRETE.....	19
3.3.3 Bref descriptif de chaque méthode.....	19
3.3.3.1 Méthode CONTRAINTE.....	19
3.3.3.2 Méthode GCP.....	20
3.3.3.3 Méthodes PENALISATION.....	20
3.3.3.4 Méthodes LAGRANGIEN	20
3.3.4 Paramètres locaux (zone par zone) de la formulation DISCRETE.....	20
3.3.4.1 Méthode VERIF.....	20
3.3.4.2 Méthode PENALISATION.....	20
3.3.4.3 Paramètres spécifiques pour le frottement	20
3.3.5 Paramètres globaux de contrôle de la formulation DISCRETE.....	21
3.3.5.1 Méthode VERIF.....	21
3.3.5.2 Méthode GLISSIERE.....	21
3.3.5.3 Méthodes CONTRAINTE et LAGRANGIEN	21
3.3.5.4 Méthode GCP.....	21
3.3.6 Paramètres locaux (zone par zone) de la formulation CONTINUE.....	22

3.3.6.1 Coefficients de la formulation CONTINUE.....	22
3.3.6.2 Opérande INTEGRATION.....	23
3.3.6.3 Opérande COMPLIANCE.....	23
3.3.6.4 Opérande USURE.....	24
3.3.6.5 Opérande GLISSIERE.....	24
3.3.7 Paramètres locaux (zone par zone) de la formulation XFEM.....	24
3.3.7.1 Opérande FISS_MAIT	24
3.3.7.2 Coefficients de la formulation XFEM.....	24
3.3.7.3 Opérande GLISSIERE.....	25
3.3.7.4 Opérande INTEGRATION.....	25
3.3.7.5 Opérande ALGO_LAGR.....	25
3.3.7.6 Opérande COEF_ECHELLE.....	25
3.3.7.7 Opérande RELATION.....	25
3.4 Contrôle de l'algorithme de résolution.....	26
3.4.1 Boucle de réactualisation géométrique REAC_GEOM.....	26
3.4.2 Boucle de réactualisation du frottement REAC_FROT.....	27
3.4.3 Boucle de point fixe pour les statuts de contact.....	27
3.4.4 Opérandes CONTACT_INIT/SEUIL_INIT.....	28
3.5 Structure de données VALE_CONT.....	28
3.6 Formulation LIAISON_UNIL.....	29
3.6.1 Paramètres locaux (zone par zone) de la formulation LIAISON_UNIL.....	29
3.6.1.1 Opérandes MAILLE/GROUP_MA/NOEUD/GROUP_NO.....	29
3.6.1.2 Opérandes SANS_NOEUD/SANS_GROUP_NO.....	29
3.6.1.3 Opérande NOM_CMP.....	29
3.6.1.4 Opérandes COEF_IMPO et COEF_MULT.....	29
3.6.2 Exemple.....	30

2 Syntaxe

```
char_contact = DEFI_CONTACT
(
  ◆ MODELE           = mo,                      [modele]
  ◆ INFO             = /1,                      [DEFAULT]
                      /2,

  ◆ FORMULATION      = /'DISCRETE'              [DEFAULT]
                      /'CONTINUE'
                      /'XFEM'
                      /'LIAISON_UNIL'

  ## Pour les formulations de contact-frottement (FORMULATION !
  ='LIAISON_UNIL')
  ◆ LISSAGE          = /'NON',                  [DEFAULT]
                      /'OUI',

  ◆ VERI_NORM        = /'OUI',                  [DEFAULT]
                      /'NON',

  ◆ FROTTEMENT       = /'SANS',                  [DEFAULT]
                      /'COULOMB',

  # Contrôle de la boucle de point fixe sur la géométrie

  {b_bouc_geom_disc FORMULATION == 'DISCRETE'
    ◆ REAC_GEOM       = /'AUTOMATIQUE',          [DEFAULT]
                      /'CONTROLE',
                      /'SANS'

    {b_automatique REAC_GEOM == 'AUTOMATIQUE'
      ◆ ITER_GEOM_MAXI = /5,                      [DEFAULT]
                      /iter_geom_maxi,          [I]
      ◆ RESI_GEOM      = /0.05,                  [DEFAULT]
                      /resi_geom,               [R]
    }
    {b_controle REAC_GEOM == 'CONTROLE'
      ◆ NB_ITER_GEOM   = /2,                      [DEFAULT]
                      /nb_iter_geom,            [I]
    }
  }

  {b_bouc_geom_cont FORMULATION == 'CONTINUE'
    ◆ REAC_GEOM       = /'AUTOMATIQUE',          [DEFAULT]
                      /'CONTROLE',
                      /'SANS'

    {b_automatique REAC_GEOM == 'AUTOMATIQUE'
      ◆ ITER_GEOM_MAXI = /5,                      [DEFAULT]
                      /iter_geom_maxi,          [I]
      ◆ RESI_GEOM      = /0.0001,                [DEFAULT]
                      /resi_geom,               [R]
    }
    {b_controle REAC_GEOM == 'CONTROLE'
      ◆ NB_ITER_GEOM   = /2,                      [DEFAULT]
                      /nb_iter_geom,            [I]
    }
  }

  {b_bouc_geom_xfem FORMULATION == 'XFEM'
    ◆ REAC_GEOM       = /'SANS',                  [DEFAULT]
                      /'CONTROLE',
                      /'AUTOMATIQUE'

    {b_automatique REAC_GEOM == 'AUTOMATIQUE'
```

```
        ♦ ITER_GEOM_MAXI      =  /5,                                [DEFAULT]
                                /iter_geom_maxi,                    [I]
        ♦ RESI_GEOM           =  /0.0001,                          [DEFAULT]
                                /resi_geom,                        [R]
    }
    {b_controle REAC_GEOM == 'CONTROLE'
        ♦ NB_ITER_GEOM        =  /2,                                [DEFAULT]
                                /nb_iter_geom,                      [I]
    }
}

# Contrôle de la boucle sur les statuts de contact

{b_bouc_cont_disc FORMULATION == 'DISCRETE'
    ♦ ITER_CONT_MULT         =  /4,                                [DEFAULT]
                                /iter_cont_mult,                    [I]
}

{b_bouc_cont_cont FORMULATION == 'CONTINUE'
    ♦ ITER_CONT_TYPE         =  /'MAXI',                            [DEFAULT]
                                /'MULT'
}

{b_bouc_cont_mult ITER_CONT_TYPE == 'MULT'
    ♦ ITER_CONT_MULT         =  /4,                                [DEFAULT]
                                /iter_cont_mult,                    [I]
}

{b_bouc_cont_maxi ITER_CONT_TYPE == 'MAXI'
    ♦ ITER_CONT_MAXI         =  /30,                                [DEFAULT]
                                /iter_cont_maxi,                    [I]
}

}

{b_bouc_cont_xfem FORMULATION == 'XFEM'
    ♦ ITER_CONT_TYPE         =  /'MAXI',                            [DEFAULT]
                                /'MULT'
}

{b_bouc_cont_mult ITER_CONT_TYPE == 'MULT'
    ♦ ITER_CONT_MULT         =  /4,                                [DEFAULT]
                                /iter_cont_mult,                    [I]
}

{b_bouc_cont_maxi ITER_CONT_TYPE == 'MAXI'
    ♦ ITER_CONT_MAXI         =  /30,                                [DEFAULT]
                                /iter_cont_maxi,                    [I]
}

}

# Contrôle de la boucle de point fixe sur les seuils de frottement

{b_bouc_frot FROTTEMENT == 'COULOMB' et
    ((FORMULATION == 'CONTINUE') ou (FORMULATION == 'XFEM'))
    ♦ REAC_FROT              =  /'AUTOMATIQUE',                      [DEFAULT]
                                /'CONTROLE',
    {b_automatique REAC_GEOM == 'AUTOMATIQUE'
        ♦ ITER_FROT_MAXI     =  /2,                                [DEFAULT]
                                /iter_frot_maxi,                    [I]
        ♦ RESI_FROT          =  /0.0001,                          [DEFAULT]
                                /resi_frot,                          [R]
    }
    {b_controle REAC_FROT == 'CONTROLE'
        ♦ NB_ITER_FROT       =  /2,                                [DEFAULT]
                                /nb_iter_frot,                      [I]
    }
}
```

```
}

# Paramètres généraux des formulations discrètes

{b_para_discret FORMULATION == 'DISCRETE'
  # Méthodes dualisées (CONTRAINTE, LAGRANGIEN)
  ♦ STOP_SINGULIER = /'OUI', [DEFAULT]
                    /'NON'
  ♦ NB_RESOL = /10, [DEFAULT]
                    /nb_resol [I]

  # Méthode 'VERIF'
  ♦ STOP_INTERP = /'NON', [DEFAULT]
                  /'OUI'
  ♦ TOLE_INTERP = /0., [DEFAULT]
                  /tole_interp [R]

  # Méthode 'GCP'
  ♦ RESI_ABSO = /resi_abso [R]
  ♦ REAC_ITER = /3, [DEFAULT]
                  /reac_iter
  ♦ ITER_GCP_MAXI = /0, [DEFAULT]
                  /iter_gcp_maxi
  ♦ PRE_COND = /'SANS', [DEFAULT]
                /'DIRICHLET'
  ♦ ITER_PRE_MAXI = /0, [DEFAULT]
                  /iter_pre_maxi
  ♦ COEF_RESI = /1., [DEFAULT]
                /coef_resi [R]
  ♦ RECH_LINEAIRE = /'ADMISSIBLE', [DEFAULT]
                    /'NON_ADMISSIBLE'
}

# Affectation du cas DISCRET

{b_affe_disc FORMULATION == 'DISCRETE'
  ZONE = _F(
    ♦ /MAILLE_MAIT = l_maille_mait
  [l_maille]
    /GROUP_MA_MAIT = grma_mait [gr_maille]
    ♦ /MAILLE_ESCL = l_maille_escl [l_maille]
    /GROUP_MA_ESCL = grma_escl [gr_maille]

    ♦ SANS_NOEUD = l_snoeud [l_noeud]
    ♦ SANS_GROUP_NO = l_sgrno [l_gr_noeud]

    ♦ APPARIEMENT = /'MAIT_ESCL' [DEFAULT]
                    /'NODAL'

    ♦ NORMALE = /'MAIT' [DEFAULT]
                /'MAIT_ESCL'
                /'ESCL'

    ♦ VECT_MAIT = /'AUTO' [DEFAULT]
                  /'FIXE'
                  /'VECT_Y'
    {b_vect_mait VECT_MAIT == 'FIXE'
      ♦ MAIT_FIXE = (Yx,Yy,Yz) [R]
    }
    {b_vect_mait VECT_MAIT == 'VECT_Y'
      ♦ MAIT_VECT_Y = (Yx,Yy,Yz) [R]
    }

    ♦ VECT_ESCL = /'AUTO' [DEFAULT]
                  /'FIXE'
  }
```

```

                                /'VECT_Y'
{b_vect_escl VECT_ESCL == 'FIXE'
  ◆ ESCL_FIXE      = (Yx,Yy,Yz)                                [R]
}
{b_vect_escl VECT_ESCL == 'VECT_Y'
  ◆ ESCL_VECT_Y = (Yx,Yy,Yz)                                [R]
}

  ◆ TYPE_APPA      = /'PROCHE'                                [DEFAULT]
                                /'FIXE'
{b_appa_fixe TYPE_APPA == 'FIXE'
  ◆ DIRE_APPA      = (Yx,Yy,Yz)                                [R]
}

  ◆ DIST_POUTRE     = /'NON'                                    [DEFAULT]
                                /'OUI'
  ◆ DIST_COQUE       = /'NON'                                    [DEFAULT]
                                /'OUI'
{b_cara DIST_POUTRE == 'OUI' or DIST_COQUE == 'OUI'
  ◆ CARA_ELEM = caract                                          [cara_elem]
}

  ◆ DIST_MAINT      = dist_maint                                [fonction]
  ◆ DIST_ESCL       = dist_escl                                  [fonction]

  ◆ TOLE_APPA        = /-1.0                                    [DEFAULT]
                                /tole_appa                        [R]
  ◆ TOLE_PROJ_EXT     = /0.50                                    [DEFAULT]
                                /tole_proj_ext                    [R]

  ◆ ALGO_CONT        = /'CONTRAINTES'                            [DEFAULT]
                                /'GCP'
                                /'PENALISATION'
                                /'LAGRANGIEN'
                                /'VERIF'

{b_penal_cont ALGO_CONT == 'PENALISATION'
  ◆ E_N          = e_n                                          [R]
}

{b_verif_cont ALGO_CONT == 'VERIF'
  ◆ GROUP_MA_FOND = l_grma_fond                                [l_gr_maille]
}

{b_active_cont ALGO_CONT == 'CONTRAINTES'
  ◆ GLISSIERE      = /'NON'                                    [DEFAULT]
                                /'OUI'
  {b_glissiere GLISSIERE == 'OUI'
    ◆ ALARME_JEU    = /0                                          [DEFAULT]
                                /alarme_jeu                        [R]
  }
}

{b_frottement FROTTEMENT == 'COULOMB'
  ◆ COULOMB        = coulomb                                    [R]
  ◆ COEF_MATR_FROT = coef_matr_frot                            [R]
  ◆ ALGO_FROT       = /'PENALISATION'                            [DEFAULT]
                                /'LAGRANGIEN'
  {b_penal_frot ALGO_FROT == 'PENALISATION'
    ◆ E_T          = e_t                                          [R]
  }
}

```

```
    })
# Paramètres généraux de la formulation continue

{b_para_continue FORMULATION == 'CONTINUE'
  ZONE = _F(
    ◆ /MAILLE_MAIT      = l_maille_mait      [l_maille]
      /GROUP_MA_MAIT    = grma_mait          [gr_maille]
    ◆ /MAILLE_ESCL      = l_maille_escl      [l_maille]
      /GROUP_MA_ESCL    = grma_escl          [gr_maille]

    ◆ SANS_NOEUD        = l_snoeud           [l_noeud]
    ◆ SANS_GROUP_NO     = l_sgrno            [l_gr_noeud]

    ◆ APPARIEMENT       = /'MAIT_ESCL'       [DEFAULT]
                          /'NODAL'

    ◆ NORMALE           = /'MAIT'            [DEFAULT]
                          /'MAIT_ESCL'
                          /'ESCL'

    ◆ VECT_MAIT         = /'AUTO'            [DEFAULT]
                          /'FIXE'
                          /'VECT_Y'
    {b_vect_mait VECT_MAIT == 'FIXE'
      ◆ MAIT_FIXE       = (Yx,Yy,Yz)         [R]
    }
    {b_vect_mait VECT_MAIT == 'VECT_Y'
      ◆ MAIT_VECT_Y     = (Yx,Yy,Yz)         [R]
    }

    ◆ VECT_ESCL         = /'AUTO'            [DEFAULT]
                          /'FIXE'
                          /'VECT_Y'
    {b_vect_escl VECT_ESCL == 'FIXE'
      ◆ ESCL_FIXE       = (Yx,Yy,Yz)         [R]
    }
    {b_vect_escl VECT_ESCL == 'VECT_Y'
      ◆ ESCL_VECT_Y     = (Yx,Yy,Yz)         [R]
    }

    ◆ TYPE_APPA         = /'PROCHE'          [DEFAULT]
                          /'FIXE'
    {b_appa_fixe TYPE_APPA == 'FIXE'
      ◆ DIRE_APPA       = (Yx,Yy,Yz)         [R]
    }

    ◆ DIST_POUTRE       = /'NON'             [DEFAULT]
                          /'OUI'
    ◆ DIST_COQUE        = /'NON'             [DEFAULT]
                          /'OUI'
    {b_cara DIST_POUTRE == 'OUI' or DIST_COQUE == 'OUI'
      ◆ CARA_ELEM       = carac              [cara_elem]
    }

    ◆ DIST_MAIT         = dist_mait           [fonction]
    ◆ DIST_ESCL         = dist_escl           [fonction]

    ◆ TOLE_APPA         = /-1.0              [DEFAULT]
                          /tole_appa         [R]
    ◆ TOLE_PROJ_EXT     = /0.50              [DEFAULT]
                          /tole_proj_ext     [R]

    ◆ FOND_FISSURE      = /'NON'             [DEFAULT]
```



```

                                /'OUI'
{b_fond FOND_FISSURE ==      'OUI'
  ◆ /NOEUD_FOND              =  l_fnoeud          [l_noeud]
    /GROUP_NO_FOND          =  l_fgrno           [l_gr_noeud]
    /MAILLE_FOND            =  l_fmaille         [l_maille]
    /GROUP_MA_FOND          =  l_fgrma           [l_gr_maille]
}

◆ RACCORD_LINE_QUAD=        /'NON'                [DEFAULT]
                                /'OUI'

{b_racc RACCORD_LINE_QUAD == 'OUI'
  ◆ /NOEUD_RACC              =  l_rnoeud          [l_noeud]
    /GROUP_NO_RACC          =  l_rgrno           [l_gr_noeud]
}

◆ GLISSIERE                  =  /'NON'                [DEFAULT]
                                /'OUI'
◆ EXCLUSION_PIV_NUL=        /'NON'                [DEFAULT]
                                /'OUI'
◆ CONTACT_INIT              =  /'NON'                [DEFAULT]
                                /'OUI'
◆ COMPLIANCE                 =  /'NON'                [DEFAULT]
                                /'OUI'
{b_glissiere COMPLIANCE ==  'OUI'
  ◆ ASPERITE                  =  asperite          [DEFAULT]
  ◆ E_N                       =  e_n              [DEFAULT]
  ◆ E_V                       =  /0                [DEFAULT]
                                /e_v              [R]
}

◆ INTEGRATION                 =  /'NOEUD'            [DEFAULT]
                                /'GAUSS'
                                /'SIMPSON'
                                /'SIMPSON1'
                                /'SIMPSON2'
                                /'NCOTES'
                                /'NCOTES1'
                                /'NCOTES2'

◆ ALGO_CONT                   =  /'STANDARD'          [DEFAULT]
                                /'AVANCE'
                                /'PENALISATION'

{b_standard ALGO_CONT ==      'STANDARD'
  ◆ COEF_CONT                 =  /100.              [DEFAULT]
                                /coef_cont         [R]
}
{b_avance ALGO_CONT ==      'AVANCE'
  ◆ COEF_REGU_CONT            =  /100.              [DEFAULT]
                                /coef_regu_cont     [R]
  ◆ COEF_STAB_CONT            =  /100.              [DEFAULT]
                                /coef_stab_cont     [R]
  ◆ COEF_PENA_CONT            =  /100.              [DEFAULT]
                                /coef_pena_cont     [R]
}
{b_standard ALGO_CONT ==      'PENALISATION'
  ◆ COEF_PENA_CONT            =  coef_pena_cont     [R]
}

{b_frottement FROTTEMENT ==  'COULOMB'
  ◆ COULOMB                   =  coulomb           [R]
  ◆ /SANS_NOEUD_FR            =  l_sfnoeud         [l_noeud]
}

```

```
        /SANS_GROUP_NO_FR      =  l_sgrno                                [l_gr_noeud]
    {b_excl_frot SANS_NOEUD_FR != None et SANS_GROUP_NO_FR != None
        ♦ EXCL_FROT_1          =  (Yx,Yy,Yz)                            [R]
        ♦ EXCL_FROT_2          =  (Yx,Yy,Yz)                            [R]
    }

    ♦ SEUIL_INIT               =  /0.                                    [DEFAULT]
                                /seuil_init                               [R]

    ♦ USURE                    =  /'SANS'                                [DEFAULT]
                                /'ARCHARD'
    {b_archard USURE = 'ARCHARD'
        ♦ K                    =  /usure_k                                [R]
        ♦ H                    =  /usure_h                                [R]
    }

    ♦ ALGO_FROT                =  /'STANDARD'                            [DEFAULT]
                                /'AVANCE'
                                /'PENALISATION'

    {b_standard ALGO_FROT == 'STANDARD'
        ♦ COEF_FROT            =  /100.                                    [DEFAULT]
                                /coef_frot                               [R]
    }
    {b_avance ALGO_FROT == 'AVANCE'
        ♦ COEF_REGU_FROT       =  /100.                                    [DEFAULT]
                                /coef_regu_frot                          [R]
        ♦ COEF_STAB_FROT       =  /100.                                    [DEFAULT]
                                /coef_stab_frot                          [R]
        ♦ COEF_PENA_FROT       =  /100.                                    [DEFAULT]
                                /coef_pena_frot                          [R]
    }
    {b_standard ALGO_FROT == 'PENALISATION'
        ♦ COEF_PENA_FROT       =  /coef_pena_frot                        [R]
    }
    }
)
}

# Parametres generaux de la methode XFEM
{b_para_continue FORMULATION == 'XFEM'
    ZONE = _F(
        ♦ FISS_MAIT            =  fiss_mait                                [fiss_xfem]
        ♦ INTEGRATION          =  /'FPG4'                                [DEFAULT]
                                /'FPG2'
                                /'FPG3'
                                /'FPG6'
                                /'FPG7'
                                /'NOEUD'
                                /'GAUSS'
                                /'SIMPSON'
                                /'SIMPSON1'
                                /'NCOTES'
                                /'NCOTES1'
                                /'NCOTES2'
        ♦ COEF_ECHELLE         =  /1.E6                                    [DEFAULT]
                                /coef_echelle                             [R]
        ♦ ALGO_LAGR             =  /'VERSION1'                            [DEFAULT]
                                /'VERSION2'
                                /'NON'
        ♦ TOLE_PROJ_EXT         =  /0.50                                  [DEFAULT]
                                /tole_proj_ext                             [R]
    )
}
```

```

    ♦ ALGO_CONT          =  /'STANDARD'          [DEFAULT]
                          /'AVANCE'
                          /'PENALISATION'

    {b_standard ALGO_CONT ==  'STANDARD'
      ♦ COEF_CONT        =  /100.                [DEFAULT]
                          /coef_cont            [R]
    }
    {b_avance ALGO_CONT ==  'AVANCE'
      ♦ COEF_REGU_CONT    =  /100.                [DEFAULT]
                          /coef_regu_cont        [R]
      ♦ COEF_STAB_CONT    =  /100.                [DEFAULT]
                          /coef_stab_cont        [R]
      ♦ COEF_PENA_CONT    =  /100.                [DEFAULT]
                          /coef_pena_cont        [R]
    }
    {b_standard ALGO_CONT ==  'PENALISATION'
      ♦ COEF_PENA_CONT    =  /coef_pena_cont      [R]
    }

    ♦ GLISSIERE          =  /'NON'                [DEFAULT]
                          /'OUI'

    ♦ RELATION           =  /'NON'                [DEFAULT]
                          /'CZM_XFEM'

    {b_frottement FROTTEMENT == 'COULOMB'
      ♦ COULOMB           =  coulomb              [R]
      ♦ SEUIL_INIT        =  /0.                  [DEFAULT]
                          /seuil_init            [R]
      ♦ ALGO_FROT         =  /'STANDARD'          [DEFAULT]
                          /'AVANCE'
                          /'PENALISATION'

      {b_standard ALGO_FROT ==  'STANDARD'
        ♦ COEF_FROT       =  /100.                [DEFAULT]
                          /coef_frot              [R]
      }
      {b_avance ALGO_FROT ==  'AVANCE'
        ♦ COEF_REGU_FROT   =  /100.                [DEFAULT]
                          /coef_regu_frot          [R]
        ♦ COEF_STAB_FROT   =  /100.                [DEFAULT]
                          /coef_stab_frot          [R]
        ♦ COEF_PENA_FROT   =  /100.                [DEFAULT]
                          /coef_pena_frot          [R]
      }
      {b_standard ALGO_FROT == 'PENALISATION'
        ♦ COEF_PENA_FROT   =  /coef_pena_frot      [R]
      }
    }
  )
}

# Affectation du cas LIAISON_UNIL

{b_affe_unil FORMULATION == 'LIAISON_UNIL'
  ZONE=F(
    ♦ /NOEUD              =  l_noeud
    [l_noeud]
      /GROUP_NO            =  l_grno              [l_gr_noeud]
      /MAILLE              =  l_maille            [l_maille]
      /GROUP_MA            =  l_grma              [l_gr_maille]
    ♦ NOM_CMP              =  l_cmp                [l_TXM]
  )
}

```

```

    ◆ COEF_IMPO          = 1_c_impo          [fonction]
    ◆ COEF_MULT          = 1_c_mult          [1_fonction]
    ◆ SANS_NOEUD         = 1_snoeud          [1_noeud]
    ◆ SANS_GROUP_NO      = 1_sgrno          [1_gr_noeud]
)
}
);
```

3 Principes

Cette commande permet de décrire les zones soumises à des conditions de contact unilatéral avec ou sans frottement. La description se fait à deux niveaux :

- Les paramètres globaux comme le choix de la formulation ou les paramètres de contrôle de l'algorithme de résolution non-linéaire (boucles de point fixe, paramètres spécifiques pour le solveur).
- Les paramètres locaux à chaque zone de contact.

Le concept issu de DEFI_CONTACT est ensuite renseigné comme paramètre dans le mot-clef simple CONTACT des opérateurs STAT_NON_LINE [U4.51.03] et DYNA_NON_LINE [U4.53.01]. Chaque zone comprend chacune deux surfaces pouvant entrer en contact qui sont décrites par la donnée des mailles qui les constituent.

Les ensembles de mailles potentiellement en contact sont : surfaciques et linéiques en dimension 3 (QUAD9, QUAD8, QUAD4 et TRIA7, TRIA6, TRIA3 et SEG3, SEG2), linéiques et concentrées en dimension 2 (SEG3, SEG2 et POI1). Les mailles de type POI1 doivent obligatoirement être sur la surface esclave. Elles ne sont pas utilisables avec la formulation CONTINUE.

Attention :

*Pour les formulations discrètes, en dimension 3, le traitement du contact avec des mailles surfaciques quadratiques de type QUAD8 nécessite de lier les nœuds milieux des côtés aux sommets de façon à avoir des résultats corrects. Cette opération est faite automatiquement dans le code. Néanmoins, pour les calculs 3D milieux continus avec des éléments hexaédriques quadratiques, l'utilisation d'éléments HEXA27 (à faces QUAD9) est **fortement conseillée**. On peut transformer un maillage constitué de mailles HEXA20 à l'aide de l'opérateur CREA_MAILLAGE [U4.23.02].*

Les structures étudiées peuvent subir de grands glissements l'une par rapport à l'autre. Il existe quatre grands types de formulations :

1. Les formulations discrètes (voir [R5.03.50]) qui correspondent à la résolution du problème unilatéral sur le maillage (déplacements et forces nodales). Cette formulation est accessible via FORMULATION='DISCRETE'. Elle est utilisable avec ou sans frottement de Coulomb.
2. La formulation continue (voir [R5.03.52]) qui est une méthode de lagrangien augmenté écrite à l'aide de l'écriture variationnelle mixte déplacements/pression de contact/pression de frottement. Cette formulation est accessible via FORMULATION='CONTINUE'. Elle est utilisable avec ou sans frottement de Coulomb.
3. Les formulations sur les éléments XFEM (voir [R7.02.12] et [R5.03.53] pour la version grands glissements), qui est une déclinaison de la formulation continue au cas des éléments XFEM. Cette formulation est accessible via FORMULATION='XFEM'. Elle est utilisable avec ou sans frottement de Coulomb.
4. La formulation de type liaison unilatérale (FORMULATION='LIAISON_UNIL'). Proche des formulations discrètes, elle s'appuie sur un algorithme utilisé en contact pour d'imposer des conditions aux limites de type inégalité sur n'importe quel degré de liberté. Cette formulation est traitée séparément dans la dernière partie de ce document.

Avant de faire un calcul avec contact, il est **indispensable** d'avoir lu les documentations de référence ainsi que la documentation [U2.04.04] de conseils aux utilisateurs qui explicitent le rôle de la plupart des mots-clés décrits ci-dessous et donnent les précautions d'utilisation. Toutes les méthodes de résolution du contact/frottement dites formulations « maillées » (formulations discrètes [R5.03.50] ou formulation continue [R5.03.52]) reposent sur une stratégie en deux temps :

- une opération d'appariement qui consiste à trouver quelles mailles et quels nœuds sont potentiellement en situation de contact;
- une opération de résolution proprement dite qui consiste à résoudre le problème de contact unilatéral avec ou sans frottement.

Les paragraphes 13 et 17 se concentrent sur la phase d'appariement tandis que le paragraphe 19 traite des méthodes de résolution du problème.

Le paragraphe 26 discute de la non-linéarité géométrique du contact, ainsi que de la non-linéarité de seuil. Enfin le paragraphe 28 décrit la structure de données produite par un calcul de contact.

La formulation 'LIAISON UNIL' est abordée dans le dernier paragraphe 29.

3.1 Opérations d'appariement (méthodes maillées hors XFEM)

```

◆ LISSAGE          =  /'NON',                                [DEFAULT]
                    /'OUI',
◆ ZONE             =  F(options d'appariement)

```

Les mots-clefs de ce paragraphe sont valables pour les formulations maillées :

- FORMULATION= 'DISCRETE'
- FORMULATION= 'CONTINUE'

3.1.1 Opérande APPARIEMENT

```
◆ APPARIEMENT = /'MAIT_ESCL' [DEFAULT]
               /'NODAT.'
```

Dans le cas des formulations discrètes l'appariement peut être nœud-facette ('MAIT_ESCL') ou nodal ('NODAL'). Pour l'appariement nodal on écrit une relation de non pénétration entre un nœud maître et un nœud esclave, alors que pour l'appariement nœud-facette on écrit cette relation entre un nœud esclave et sa projection sur la maille maître la plus proche (voir [R5.03.50] pour les détails de la méthode d'appariement).

L'appariement nodal est déconseillé car la méthode nœud-facette est plus générale et est la seule à permettre de prendre en compte les grands glissements de façon précise.

3.1.2 Opérandes MAILLE MAIT/GROUP MA MAIT/MAILLE ESCL/GROUP MA ESCL

```
◆ /MAILLE_MAIT      = 1_maille_mait      [1_maille]
  /GROUP_MA_MAIT    = 1_grma_mait        [1_gr_maille]
◆ /MAILLE_ESCL      = 1_maille_escl      [1_maille]
  /GROUP_MA_ESCL    = 1_grma_escl        [1_gr_maille]
```

Pour les formulations maillées l'utilisateur fournit la liste des mailles de contact potentielles de la surface maître (MAILLE_MAIT ou GROUP_MA_MAIT) et de la surface esclave (MAILLE_ESCL ou GROUP_MA_ESCL). Ces mailles doivent être surfaciques ou linéiques en dimension 3 (QUAD9, QUAD8, QUAD4, TRIA7, TRIA6, TRIA3, SEG3, SEG2), linéiques et concentrées en dimension 2 (SEG3, SEG2 et POI1).

Attention :

Il est important de vérifier que la connectivité de ces mailles est telle que la normale est sortante à la structure (pour ce faire, utilisez MODI_MAIILLAGE mot clé ORIE_PEAU_2D, ORIE_PEAU_3D, ORIE_NORM_COQUE [U4.23.04]).

Par ailleurs, il faut s'assurer que les structures «ne tiennent pas que par le contact» (notamment dans le cas d'un chargement en force imposée) : les mouvements de corps rigide doivent être bloqués par des conditions aux limites appropriées. Une bonne façon de le vérifier est d'effectuer un calcul avec l'opérateur `MECA_STATIQUE` ou avec les opérateurs `STAT_NON_LINE` / `DYNA_NON_LINE` en enlevant les conditions de contact du chargement.

Dans toute la suite, on utilisera le concept maître-esclave : les nœuds de la surface esclave ne peuvent pas « pénétrer » dans les facettes (ou les nœuds) de la surface maître. Dans le cas de l'appariement de type 'MAIT_ESCL', la surface maître est celle définie par 'MAILLE_MAIT' ou 'GROUP_MA_MAIT'. Dans le cas de l'appariement de type 'NODAL' (disponible uniquement pour les formulations discrètes), la surface maître est celle qui doit comporter le plus de nœuds. Si ce n'est pas le cas l'utilisateur est arrêté par un message d'erreur et invité à intervertir les deux surfaces.

Remarques :

- Il est impossible de mélanger les modélisations purement bidimensionnelles (contraintes planes *C_PLAN*, déformations planes *D_PLAN* et axisymétriques *AXIS*) avec les modélisations tri-dimensionnelles. Les surfaces maître et esclave doivent être de même nature (2D/2D ou 3D/3D). Un message d'erreur vous arrêtera dans le cas contraire. Notons qu'une poutre, une plaque ou une coque sont de dimension 3 et qu'il est donc possible de faire du contact poutre/3D ou poutre/plaque.
- Les surfaces de contact en vis-à-vis doivent être disjointes. Si ce n'est pas le cas, il faut utiliser le mot-clé décrit dans le paragraphe suivant pour éliminer les nœuds communs du contact.

3.1.3 Opérandes SANS_NOEUD/SANS_GROUP_NO

◇ SANS_NOEUD	=	l_snoeud	[l_noeud]
◇ SANS_GROUP_NO	=	l_sgrno	[l_gr_noeud]

Ces opérandes permettent d'exclure des nœuds de la liste des nœuds esclaves, opération qui est recommandée pour des nœuds soumis à des conditions aux limites dans la direction attendue du contact (exemple encastrement).

3.1.4 Opérandes TYPE_APPA/DIRE_APPA

◇ TYPE_APPA	=	/'PROCHE' /'FIXE'	[DEFAULT]
◆ DIRE_APPA	=	(Yx, Yy, Yz)	[R]

Le choix de la maille maître appariée au nœud esclave se fait par une opération de minimisation de la distance entre le nœud esclave et les mailles maîtres. C'est l'option par défaut *TYPE_APPA*='PROCHE'. Cette procédure peut provoquer des oscillations, par exemple, si la projection sur une maille n'est pas unique, ce qui peut être le cas si la maille maître est convexe. Pour éviter ces oscillations, l'utilisateur peut renseigner une direction d'appariement fixe via l'option *TYPE_APPA*='FIXE', la direction étant alors donnée par un vecteur dans *DIRE_APPA*.

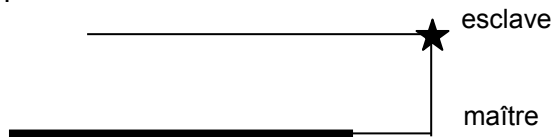
3.1.5 Opérandes TOLE_APPA et TOLE_PROJ_EXT

◇ TOLE_APPA	=	/-1.0 /tole_appa	[DEFAULT] [R]
◇ TOLE_PROJ_EXT	=	/0.50 /tole_proj_ext	[DEFAULT] [R]

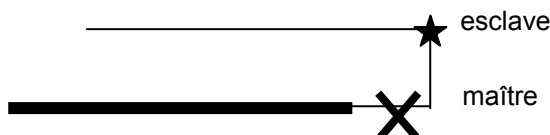
Lors de la recherche de la maille maître appariée au nœud esclave courant, il est possible de restreindre le choix par utilisation du mot-clé *TOLE_APPA*. Si *TOLE_APPA*=-1 (valeur par défaut), alors toutes les mailles maîtres données dans la zone de contact sont susceptibles d'être appariées avec le nœud esclave. Si *TOLE_APPA*=val avec val un réel positif, alors seules les mailles maîtres situées en 3D dans la sphère, en 2D dans le cercle de rayon val centré(e) sur le nœud esclave peuvent être appariées.

Dans certaines situations, il peut être nécessaire d'étendre de manière fictive les mailles de la surface maître. Prenons le cas du contact en 2D (les surfaces de contact sont donc des segments), on se place sur le bord de la surface de contact.

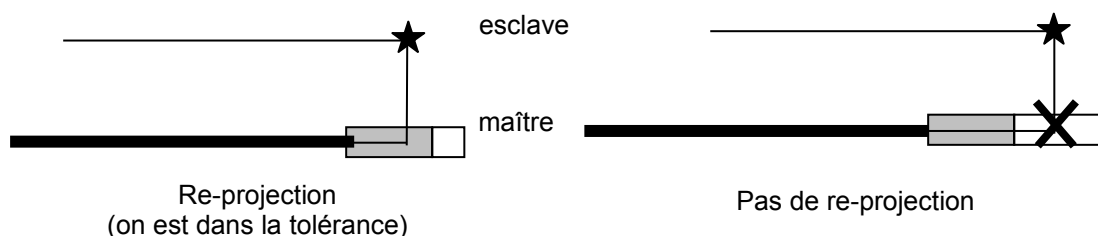
Code_Aster procède à la projection d'un nœud esclave sur la surface maître, cette projection tombe en dehors de celle-ci :



Une solution consiste à ne pas appairer le nœud, ce qui revient à l'exclure du contact :



Cependant, cette solution ne tient pas compte des cas limites et peut provoquer des interpénétrations intempestives si jamais le maillage n'est pas « optimal » (c'est à dire pas assez fin, ce qui est difficile à assurer dans le cadre des grandes transformations). On a donc opté pour une solution intermédiaire en limitant l'extension de la surface maître susceptible d'entraîner une reprojexion.



La valeur limite de cette re-projection est fixée par le mot-clef 'TOLE_PROJ_EXT' qui prend pour argument la valeur (rapportée à l'élément de référence) de l'extension de la maille maître dans laquelle on autorise la re-projection. Par défaut, cette valeur est fixée à 0,50. Par exemple en 2D, cela signifie que tout nœud esclave se projetant à plus de 25% à droite ou à gauche de la longueur de la maille maître ne sera pas reprojété (dans le cas d'un segment, l'élément de référence est de longueur 2, cf. [R3.01.01]). Pour interdire complètement la re-projection, il suffit de fixer TOLE_PROJ_EXT négatif. Cet opérateur est valable en 2D et en 3D (dans ce dernier cas, il s'agit de l'extension d'une maille surfacique de contact).

3.1.6 Choix des normales

Par défaut, la normale prise par Code_Aster pour évaluer le jeu entre les deux surfaces est la normale extérieure à la maille maître appariée au nœud esclave. Il est toutefois possible de changer ce choix.

3.1.6.1 Type de normale (NORMALE)

◆ NORMALE = / 'MAIT' [DEFAULT]
/ 'MAIT_ESCL'
/ 'ESCL'

En premier lieu, il est possible de choisir le type de la normale :

- la normale extérieure à la maille maître (NORMALE= 'MAIT', par défaut) ;
- la normale intérieure à la maille esclave (NORMALE= 'ESCL') ;
- une moyenne entre les deux normales maître et esclave (NORMALE= 'MAIT_ESCL').

3.1.6.2 Choix des normales maître ou esclave (VECT_MAIT/VECT_ESCL)

◆ VECT_MAIT = / 'AUTO' [DEFAULT]
/ 'FIXE'
/ 'VECT_Y'
◆ MAIT_FIXE = (Yx, Yy, Yz) [R]

```

♦      MAIT_VECT_Y      =      (Yx,Yy,Yz)      [R]

♦  VECT_ESCL      =      / 'AUTO'      [DEFAULT]
                        / 'FIXE'
                        / 'VECT_Y'

♦  ESCL_FIXE      =      (Yx,Yy,Yz)      [R]
♦  ESCL_VECT_Y    =      (Yx,Yy,Yz)      [R]

```

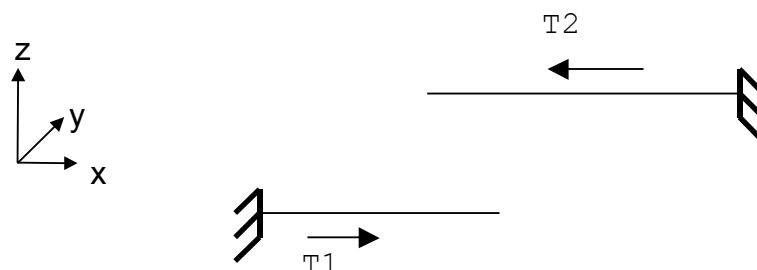
Le choix de la normale (sur la maille esclave ou sur la maille maitre) peut se faire de plusieurs manières différentes :

- Automatiquement, la normale est alors calculée par l'utilisation des fonctions de forme de l'élément, c'est l'option ' AUTO '.
- Fixe et donnée directement par l'utilisateur, c'est l'option `FIXE`. On entre alors la normale avec le mot-clef `MAIT_FIXE` ou `ESCL_FIXE`.
- Variable et donnée indirectement par la tangente, c'est l'option `VECT_Y`. L'utilisateur donne alors la direction du second vecteur tangent par le mot-clef `MAIT_VECT_Y` ou `ESCL_VECT_Y`. Le code reconstitue alors la normale à partir de la première tangente de la maille.

Une direction fixe (option `VECT_*= 'FIXE'`) est nécessaire si on doit évaluer la normale sur une maille de type `POI1` .

Une direction variable donnée par la seconde tangente (`VECT_MAIT='VECT_Y'`) est particulièrement utilisée dans le cas des poutres qui ne se déforment que dans un plan. Dans ce cas, l'utilisation de l'option de type `VECT_Y` permet de réactualiser continuellement la normale pendant que la poutre se déforme.

$$\text{VECT_Y} \wedge T = N$$



Comme ici $T1=(1,0,0)$ et $T2=(-1,0,0)$, avec $\text{VECT_Y}=(0,0,1)$, on obtient la normale souhaitée pour chaque poutre : $N1=(0,1,0)$ et $T2=(0,-1,0)$. Attention au fait que tout ceci est entièrement lié à l'orientation de chaque poutre.

On peut fixer l'orientation de chaque poutre à l'aide du mot-clé `ORIE_LIGNE` de l'opérateur `MODI_MAILLAGE` [U4.23.04] .

3.1.6.3 Lissage des normales (LISSAGE)

```

♦  LISSAGE      =      / 'NON',      [DEFAULT]
                        / 'OUI',

```

L'opérande `LISSAGE` permet de lisser les normales aux surfaces de contact intervenant dans le calcul de la matrice de contact. On notera Q un nœud quelconque des surfaces de contact (maître ou esclave), P un nœud de la surface esclave et M le nœud maître obtenu par projection du nœud P .

Le lissage se fait en deux étapes :

- la première étape du lissage consiste à effectuer une moyenne des normales aux mailles qui contiennent le nœud Q ,
- la seconde étape consiste à calculer une moyenne des normales aux sommets de la maille contenant M . Cette moyenne étant pondérée par les fonctions de forme associées à M .

Le lissage prend en compte les options de normales décidées par les mots-clés `VECT_MAIT` et `VECT_ESCL`. Attention, ce paramètre est global et non pas zoné.

3.1.7 Modification du jeu

Le jeu est toujours calculé comme étant la distance minimale entre le nœud esclave et la projection sur la maille maître la plus proche, modulo les options de choix de cette normale (voir paragraphe précédent). Il est cependant possible de définir des valeurs de jeu « en dur », par exemple, pour simuler la présence d'un trou ou d'une bosse non représentée par le maillage, ou pour prendre en compte les caractéristiques géométriques des éléments de structure qui utilisent la notion de fibre ou de surface neutre.

3.1.7.1 Opérandes DIST_MAIT/DIST_ESCL

◇ DIST_MAIT	=	dist_mait	[fonction]
◇ DIST_ESCL	=	dist_escl	[fonction]

Ces opérandes permettent de prendre en compte des « trous » ou des « bosses » non maillés dans le cas des formulations maillées ou l'épaisseur des coques (les relations de contact sont écrites entre les deux feuillets moyens) pour les surfaces maîtres (DIST_MAIT) ou esclaves (DIST_ESCL). On compte la distance positivement dans le sens de la normale sortante à la structure (cf. [R5.03.50]). Les grandeurs renseignées sont nécessairement des fonctions des variables d'espace. Si l'utilisateur désire une valeur fixe, il doit définir une fonction constante (voir l'opérateur DEFI_CONSTANTE).

3.1.7.2 Opérandes DIST_POUTRE/DIST_COQUE/CARA_ELEM

◇ DIST_POUTRE	=	/'NON' /'OUI'	[DEFAULT]
◇ DIST_COQUE	=	/'NON' /'OUI'	[DEFAULT]
◆ CARA_ELEM	=	carac	[cara_elem]

Semblables aux deux mots-clés précédents, les mot-clefs DIST_POUTRE/DIST_COQUE (et le renseignement obligatoire du mot_clef CARA_ELEM qui leur est associé), permet d'introduire un jeu fictif qui repose sur la description de l'élément de structure dans le concept produit par AFFE_CARA_ELEM que l'utilisateur a utilisé :

- pour les éléments de poutre (modélisation POU_*), le mot-clef DIST_POUTRE autorise le code à prendre un jeu supplémentaire correspondant au rayon de la section **circulaire** de la poutre.
- pour les éléments de plaque ou de coque (modélisation DKT ou COQUE_3D par exemple), le mot-clef DIST_COQUE autorise le code à prendre un jeu supplémentaire correspondant à la demi-épaisseur autour du feuillet moyen d'une coque.

Attention, lorsque l'on utilise ces mots-clefs, le jeu fictif n'est rajouté que sur la surface esclave. Si l'on modélise un contact entre deux éléments de structures, il faudra donc aussi utiliser DIST_MAIT.

3.2 Opérations d'appariement spécifiques à la formulation CONTINUE

3.2.1 Opérandes RACCORD_LINE_QUAD/GROUP_NO_RACC/NOEUD_RACC

◆ RACCORD_LINE_QUAD=	/'NON' /'OUI'	[DEFAULT]
◆ /NOEUD_RACC	= l_rnoeud	[l_noeud]
/GROUP_NO_RACC	= l_rgrno	[l_gr_noeud]

Dans le cas d'une formulation continue et lorsque l'on a au sein du même modèle des maillages non compatibles avec des ordres d'interpolation EF différents (ici l'un est linéaire, l'autre est quadratique), des conditions de raccord y sont alors appliquées afin d'assurer la continuité des déplacements et éviter les « trous » en surface, ce qui peut engendrer des relations surabondantes (systèmes matriciels singuliers et donc pivots nuls).

On peut alors utiliser l'option RACCORD_LINE_QUAD='OUI' pour traiter ce conflit entre ces conditions de raccordement surfacique cinématique linéaire/quadratique et les conditions de contact.

On renseigne alors la liste des nœuds sur lesquels portera le traitement théorique précisé dans la documentation de référence [R5.03.52] par l'intermédiaire des mot-clés `GROUP_NO_RACC` ou `NOEUD_RACC`.

3.2.2 Opérandes `FOND_FISSURE`/`GROUP_NO_FOND`/`NOEUD_FOND`/`GROUP_MA_FOND`/`MAILLE_FOND`

◆ <code>FOND_FISSURE</code>	=	<code>/'NON'</code>	[DEFAULT]
		<code>/'OUI'</code>	
◆ <code>/NOEUD_FOND</code>	=	<code>l_fnoeud</code>	[<code>l_noeud</code>]
<code>/GROUP_NO_FOND</code>	=	<code>l_fgrno</code>	[<code>l_gr_noeud</code>]
<code>/MAILLE_FOND</code>	=	<code>l_fmaille</code>	[<code>l_maille</code>]
<code>/GROUP_MA_FOND</code>	=	<code>l_fgrma</code>	[<code>l_gr_maille</code>]

Dans le cas d'une formulation continue et lorsqu'un modèle comporte un fond de fissure avec l'utilisation de la technique de Barsoum pour le calcul des coefficients de concentration des contraintes, on peut avoir des problèmes d'incompatibilités lors de la résolution du contact entre les lèvres de fissure. En effet :

- l'interpolation polynomiale n'étant plus quadratique à cause du déplacement des nœuds milieux au quart, on doit veiller à contrôler l'influence des éléments de Barsoum sur l'intégration des termes de contact. Les schémas d'intégration de type Simpson ou Newton-Cotes avec subdivision en sous-éléments d'intégration virtuels permettent d'intégrer proprement les contributions élémentaires des éléments de contact en fond de fissure.
- les degrés de liberté de contact aux nœuds de fond de fissure n'ont aucune signification physique car appartenant à la fois aux surfaces maître et esclave.
- la transformation géométrique en certains nœuds particuliers des mailles modifiées en fond de fissure est singulière.

On peut alors utiliser l'option `FOND_FISSURE='OUI'` pour traiter ces problèmes d'incompatibilités entre les conditions de contact et la technique de Barsoum en fond de fissure.

On renseigne la liste des nœuds (2D/3D) ou des mailles (3D) sur lesquels portera le traitement théorique précisé dans la documentation de référence [R5.03.52] par l'intermédiaire des mot-clés `GROUP_NO_FOND`, `NOEUD_FOND`, `GROUP_MA_FOND` ou `MAILLE_FOND`.

3.2.3 Opérandes `SANS_NOEUD_FR`/`SANS_GROUP_NO_FR`

◆ <code>/SANS_NOEUD_FR</code>	=	<code>l_sfnoeud</code>	[<code>l_noeud</code>]
<code>/SANS_GROUP_NO_FR</code>	=	<code>l_sgrno</code>	[<code>l_gr_noeud</code>]
◆ <code>EXCL_FROT_1</code>	=	<code>(Yx, Yy, Yz)</code>	[<code>R</code>]
◆ <code>EXCL_FROT_2</code>	=	<code>(Yx, Yy, Yz)</code>	[<code>R</code>]

Pour la formulation continue, ces mots-clefs permettent à l'utilisateur d'exclure des directions de frottement qui risquent d'entrer en conflit avec d'autres conditions aux limites de Dirichlet. Les directions de frottement exclues sont indiqués par `EXCL_FROT_1` (en 2D) et `EXCL_FROT_1/EXCL_FROT_2` (en 3D). L'exclusion de la direction de frottement permet néanmoins de garder le caractère contactant d'un nœud.

3.2.4 Opérande `EXCLUSION_PIV_NUL`

◆ <code>EXCLUSION_PIV_NUL</code>	=	<code>/'NON'</code>	[DEFAULT]
		<code>/'OUI'</code>	

Pour la formulation continue, ce mot-clef une fois activé (`EXCLUSION_PIV_NUL='OUI'`) permet d'exclure automatiquement d'éventuelles redondances entre les conditions aux limites de Dirichlet et les conditions de contact. Ce mécanisme ne permet pas de détecter tous les cas, en particulier, il est inefficace sur les conditions aux limites non homogènes, impliquant plus de deux nœuds ou non écrites dans le repère principal (X, Y, Z) . En pratique, il n'est donc efficace que pour `DDL_IMPO` et `LIAISON_DDL`. Ce traitement peut s'avérer coûteux lorsque le nombre de nœuds esclaves est grand. L'utilisateur peut utiliser la fonctionnalité `SANS_NOEUD` / `SANS_GROUP_NO` pour indiquer lui-même les nœuds susceptibles d'être en conflit.

3.3 Choix des méthodes de résolution

Des opérandes permettent de sélectionner une méthode de calcul suivant le type de contact (2D/3D et avec ou sans frottement) que l'on veut traiter et le type de formulation.

3.3.1 Frottement

◆ FROTTEMENT = / 'SANS', [DEFAULT]
/ 'COULOMB',

Pour affecter du frottement de Coulomb aux zones de contact, utilisez le mot-clef FROTTEMENT='COULOMB'. Ce mot-clef est **global** (valable pour toutes les zones), mais il est possible d'affecter du frottement zone par zone en jouant sur la valeur du coefficient de Coulomb (on met ce coefficient à 0 quand on ne veut pas de frottement).

3.3.2 Choix de la méthode pour le contact en formulation DISCRETE

◆ ALGO_CONT = / 'CONTRAINTTE' [DEFAULT]
/ 'GCP'
/ 'PENALISATION'
/ 'LAGRANGIEN'
/ 'VERIF'
◆ ALGO_FROT = / 'PENALISATION' [DEFAULT]
/ 'LAGRANGIEN'

Pour sélectionner le type d'algorithme dans la formulation DISCRETE, on utilise les mots-clefs ALGO_CONT et ALGO_FROT. ALGO_FROT est modifiable seulement si FROTTEMENT='COULOMB'. Toutes les combinaisons ne sont pas possibles:

	ALGO_FROT='PENALISATION'	ALGO_FROT='LAGRANGIEN'
ALGO_CONT='CONTRAINTTE'	NON	NON
ALGO_CONT='GCP'	NON	NON
ALGO_CONT='VERIF'	NON	NON
ALGO_CONT='PENALISATION'	OUI	NON
ALGO_CONT='LAGRANGIEN'	OUI	OUI

Les différentes méthodes de résolution sont :

- **CONTRAINTTE** : par défaut on traite le problème du contact unilatéral exact avec la méthode des contraintes actives (voir [R5.03.50]). **Pas de frottement possible.**
- **GCP** : c'est une méthode itérative proche de la méthode 'CONTRAINTTE' mais qui est particulièrement adaptée aux cas où le nombre de liaisons de contact est très élevé. **Pas de frottement possible.**
- **VERIF** : la méthode de vérification permet de contrôler si deux surfaces s'interpénètrent ou pas sans imposer les conditions de contact. C'est donc une méthode qui ne se préoccupe que de l'aspect géométrique et qui est peu coûteuse en terme de temps CPU. On peut l'utiliser par exemple pour contrôler que les deux lèvres d'une fissure ne s'interpénètrent pas. **Pas de frottement possible.**
- **PENALISATION** : la méthode pénalisée permet de traiter des problèmes de contact pénalisé avec ou sans frottement 2D ou 3D.
- **LAGRANGIEN** : la méthode lagrangienne permet de traiter de façon exacte par multiplicateurs de Lagrange des problèmes de contact avec ou sans frottement en 2D et 3D (voir [R5.03.50])

Bien que les différents algorithmes soient donnés par zone, il n'est pas possible de mélanger plusieurs combinaisons de méthodes dans le même DEFI_CONTACT.

3.3.3 Bref descriptif de chaque méthode

3.3.3.1 Méthode CONTRAINTTE

Cette permet de résoudre des problèmes de contact sans frottement de manière exacte (aucune interpénétration n'est tolérée). Elle est particulièrement rapide et robuste, sa convergence est prouvée. La résolution du système d'inéquations résultant du contact s'appuyant sur la construction explicite d'un complément de Schur, son utilisation est limitée à quelques centaines de liaisons de contact. Au-delà les coûts mémoire et calcul deviennent trop important.

3.3.3.2 Méthode GCP

Cette méthode permet de résoudre des problèmes de contact sans frottement. Elle résout avec une précision réglable (éventuellement très élevée) les conditions de contact à l'aide de multiplicateurs de Lagrange. Elle est en tout point semblable à la méthode 'CONTRAINTE' à la différence près qu'elle est de nature totalement itérative et donc très peu gourmande en mémoire. En d'autres termes, le surcoût de stockage lié à la prise en compte du contact est quasiment nul. Cette spécificité a pour effet de la rendre particulièrement adaptée aux cas impliquant des nombres importants de liaisons de contact.

3.3.3.3 Méthodes PENALISATION

Cette méthode est une méthode de résolution du contact/frottement par régularisation. Elle est non exacte dans le sens où il y a **toujours** interpénétration lorsque le contact est établi. Si l'on utilise une formulation discrète avec pénalisation, il convient donc de renseigner le ou les coefficients de pénalisation.

3.3.3.4 Méthodes LAGRANGIEN

C'est une méthode qui diffère de la méthode CONTRAINTE par le fait qu'elle ne dispose pas de preuve de convergence (la mise à jour des contraintes actives se faisant par paquet) et qu'elle est utilisable en FROTTEMENT.

3.3.4 Paramètres locaux (zone par zone) de la formulation DISCRETE

◆ `ZONE` = `_F(paramètres locaux)`

Les mots-clefs de ce paragraphe sont valables pour la formulation DISCRETE. Les paramètres sont définis par zone.

3.3.4.1 Méthode VERIF

◆ `GROUP_MA_FOND` = `l_grma_fond` [`l_gr_maille`]

On peut indiquer le fond de fissure par le mot-clef `GROUP_MA_FOND`. Ce mot-clef permet d'exclure du contact tous les nœuds qui se trouvent en fond de fissure (et qui risqueraient donc d'être communs à la surface maître et la surface esclave).

3.3.4.2 Méthode PENALISATION

◆ `E_N` = `e_n` [R]
◆ `E_T` = `e_t` [R]

`e_n` est le coefficient de pénalisation sur l'interpénétration pour la méthode pénalisée. Il est homogène à la raideur de ressorts placés entre les surfaces de contact pour empêcher l'interpénétration. Une valeur de l'ordre du plus grand module d'Young des solides en contact est initialement recommandée. On augmentera la valeur du coefficient jusqu'à l'obtention de résultats stables. En outre il est possible de contrôler les distances d'interpénétration et donc d'affiner son choix de coefficient, ce qui n'est pas le cas en présence de frottement, puisque l'on ne sait pas a priori quelles sont les zones glissantes et non glissantes alors qu'en cas de contact on peut vérifier que les distances d'interpénétration sont petites devant les dimensions caractéristiques des zones de contact.

`e_t` est le coefficient de pénalisation sur le glissement pour la méthode pénalisée. Il n'est nécessaire que lorsque le frottement est actif. Il faut augmenter la valeur du coefficient jusqu'à l'obtention de résultats stables.

3.3.4.3 Paramètres spécifiques pour le frottement

◆ COULOMB	=	coulomb	[R]
◆ COEF_MATR_FROT	=	coef_matr_frot	[R]

Le paramètre `COULOMB` permet de régler le coefficient de frottement.

Dans le cas d'une formulation pénalisée en frottement en 3D, le paramètre `COEF_MATR_FROT`, compris entre 0 et 1, permet de modérer l'effet déstabilisant de la partie négative de la matrice de glissement (qui est ajoutée à la rigidité tangente en 3D). Plus ce coefficient est grand, meilleure est la convergence lorsque l'on est proche de l'équilibre et plus la résolution est difficile loin de l'équilibre. Une valeur de 0.5 est donc initialement conseillée. Le défaut de 0 assure la convergence systématique pour un temps de calcul plus long.

Ce coefficient est utilisé pour traiter des contacts surfaciques avec frottement en 3D. Il n'est pas utilisé le reste du temps.

3.3.5 Paramètres globaux de contrôle de la formulation DISCRETE

3.3.5.1 Méthode VERIF

◆ STOP_INTERP	=	/'NON',	[DEFAULT]
		/'OUI'	
◆ TOLE_INTERP	=	/0.,	[DEFAULT]
		/tole_interp	[R]

Cette méthode réalise un contrôle de l'interpénétration de deux surfaces sans imposer les conditions de contact (s'il y a interpénétration, elle restera). S'il y a interpénétration, on aura une alarme. Le paramètre `STOP_INTERP` permet d'arrêter le calcul au lieu d'alarmer l'utilisateur. `TOLE_INTERP` règle la valeur d'interpénétration tolérée (homogène à une longueur).

3.3.5.2 Méthode GLISSIERE

◆ GLISSIERE	=	/'NON'	[DEFAULT]
		/'OUI'	
◆ ALARME_JEU	=	/0	[DEFAULT]
		/alarme_jeu	[R]

Cette option est disponible uniquement pour la méthode 'CONTRAINTE'. Elle permet d'activer le mode de contact bilatéral ou en glissière, dans lequel deux surfaces se trouvant en contact restent « collées » (c'est-à-dire avec un jeu nul) quelque soit l'évolution du chargement. Elle autorise de grands glissements relatifs et le mode glissière n'est pas activé avant que les surfaces soient effectivement en contact (elle ne colle pas a priori deux surfaces distantes d'un jeu non nul si le chargement ne l'implique pas).

L'opérande 'ALARME_JEU' permet d'activer une alarme dès que l'algorithme détecte que, sans la méthode glissière, il y aurait décollement des deux surfaces (un jeu virtuel supérieur à zéro). Sa valeur est réglée par défaut à 0, ce qui alarme l'utilisateur dès que les surfaces auraient dû se décoller sans l'option activée.

3.3.5.3 Méthodes CONTRAINTE et LAGRANGIEN

◆ STOP_SINGULIER	=	/'OUI',	[DEFAULT]
		/'NON'	
◆ NB_RESOL	=	/10,	[DEFAULT]
		/nb_resol	[I]

Ces deux paramètres globaux sont accessibles pour les méthodes discrètes s'appuyant sur la dualisation des conditions de contact-frottement et la construction d'un complément de Schur.

`NB_RESOL` est le nombre de résolutions simultanées pour la construction du complément de Schur. Réaliser plusieurs résolutions simultanées permet de manipuler des matrices par blocs. Augmenter `nb_resol` accélère la construction du complément de Schur mais fait perdre de la place mémoire. `nb_resol=10` est un bon compromis. Ce paramètre est réservé à un usage **d'expert**.

STOP_SINGULIER permet de désactiver l'erreur fatale apparaissant si le complément de Schur est singulier suite à une perte importante de décimales dans la factorisation (8 décimales par défaut). On renseigne pour cela STOP_SINGULIER='NON'. Ce paramètre est réservé à un usage **d'expert**.

3.3.5.4 Méthode GCP

◆ RESI_ABSO	=	/resi_abso	[R]
◆ REAC_ITER	=	/3,	[DEFAULT]
		/reac_iter	
◆ ITER_GCP_MAXI	=	/0,	[DEFAULT]
		/iter_gcp_maxi	
◆ PRE_COND	=	/'SANS',	[DEFAULT]
		/'DIRICHLET'	
◆ ITER_PRE_MAXI	=	/0,	[DEFAULT]
		/iter_pre_maxi	
◆ COEF_RESI	=	/1.,	[DEFAULT]
		/coef_resi	[R]
◆ RECH_LINEAIRE	=	/'ADMISSIBLE',	[DEFAULT]
		/'NON_ADMISSIBLE'	

Le mot-clé RESI_ABSO permet de régler la précision de la résolution des inéquations de contact (c'est le critère d'arrêt sur les jeux de l'algorithme itératif). RESI_ABSO (qu'il faut comprendre par « résidu absolu ») représente le niveau d'interpénétration toléré pour les corps en contact. D'un point de vue pratique, il est conseillé lors de la mise au point d'une étude de partir d'une valeur assez grossière (typiquement de l'ordre de la taille des éléments au voisinage des surfaces de contact) et de la diminuer pour obtenir une solution plus précise.

Comme toute méthode de résolution itérative, la méthode 'GCP' peut être accélérée par l'usage d'un préconditionneur. Un seul est aujourd'hui disponible et il s'agit d'un préconditionneur de Dirichlet PRE_COND='DIRICHLET'. Son usage peut dans certains cas accélérer et diminuer sensiblement le temps de résolution. On peut également le désactiver par PRE_COND='SANS'.

La phase de préconditionnement est réalisée par la résolution itérative d'un problème auxiliaire. Le mot clé COEF_RESI permet de spécifier la précision de la résolution de ce problème par : COEF_RESI*RESI_ABSO.

La méthode de résolution 'GCP' nécessite une phase appelée recherche linéaire. Deux variantes sont disponibles : admissible ou pas, mot-clef RECH_LINEAIRE.

La méthode de résolution 'GCP' utilise des directions de recherche conjuguées les unes aux autres (dans la même logique qu'un gradient conjugué classique). Toutes les reac itérations (mot-clef REAC_ITER), ces directions sont réinitialisées.

3.3.6 Paramètres locaux (zone par zone) de la formulation CONTINUE

◆ ZONE = _F(paramètres locaux)

Les mots-clefs de ce paragraphe sont valables pour la formulation CONTINUE. Les paramètres sont définis par zone.

3.3.6.1 Coefficients de la formulation CONTINUE

◆ ALGO_CONT	=	/'STANDARD'	[DEFAULT]
		/'AVANCE'	
		/'PENALISATION'	
◆ COEF_CONT	=	/100.	[DEFAULT]
		/coef_cont	[R]
◆ COEF_REGU_CONT	=	/100.	[DEFAULT]
		/coef_regu_cont	[R]
◆ COEF_STAB_CONT	=	/100.	[DEFAULT]
		/coef_stab_cont	[R]

```

♦ COEF_PENA_CONT      = /100.                                [DEFAULT]
                        /coef_pena_cont                       [R]
♦ ALGO_FROT            = /'STANDARD'                          [DEFAULT]
                        /'AVANCE'
                        /'PENALISATION'
♦ COEF_FROT            = /100.                                [DEFAULT]
                        /coef_frot                             [R]
♦ COEF_REGU_FROT       = /100.                                [DEFAULT]
                        /coef_regu_frot                        [R]
♦ COEF_STAB_FROT       = /100.                                [DEFAULT]
                        /coef_stab_frot                        [R]
♦ COEF_PENA_FROT       = /100.                                [DEFAULT]
                        /coef_pena_frot                        [R]

```

La formulation continue est une généralisation des méthodes lagrangiennes. Dans la documentation de référence [R5.05.32], on explique les différentes versions. La formulation continue est une formulation lagrangienne pénalisée avec augmentation. Chacun de ces termes (augmentation, régularisation et pénalisation) est donnée par six coefficients (trois pour le contact et trois pour le frottement).

Si on choisit `ALGO_CONT='STANDARD'` et `ALGO_FROT='STANDARD'`, la formulation continue est alors équivalente à une formulation lagrangienne augmentée classique dont le coefficient d'augmentation est donné, respectivement, par `COEF_CONT` et `COEF_FROT`.

Si on choisit `ALGO_CONT='AVANCE'` et `ALGO_FROT='AVANCE'`, on peut alors accéder individuellement aux différents termes de la formulation lagrangienne augmentée (termes de régularisation, de stabilisation et de pénalisation) par `COEF*_CONT` et `COEF*_FROT`. Ces paramètres sont à réserver aux **experts**.

Les coefficients de type `REGU` et `STAB` sont relatifs à la régularisation des lois de contact/frottement. Ils peuvent prendre des valeurs de l'ordre de grandeur du pas de temps en dynamique (10^{-5} , 10^{-6} ...) jusqu'à beaucoup plus importante (500 par exemple). Ces coefficients agissent sur la vitesse de convergence mais pas sur la précision du résultat.

Les coefficients de type `PENA` permettent d'ajouter un terme de stabilisation qui renforce la coercivité de l'opérateur. Leurs valeurs n'influencent que la convergence. Pour plus de détails, voir [R5.03.52].

Si on choisit `ALGO_CONT='PENALISATION'` et `ALGO_FROT='PENALISATION'`, la formulation est alors purement pénalisée. On n'accède qu'aux termes `COEF_PENA_CONT` et `COEF_PENA_FROT`, les autres paramètres `COEF*_CONT` et `COEF*_FROT` sont automatiquement affectés à la valeur 0 (on remarque que l'on est ici dans un cas particulier des choix `ALGO_CONT='AVANCE'` et `ALGO_FROT='AVANCE'`).

La méthode pénalisée est décrite dans le cas de la méthode discrète au §3.3.4.2, `COEF_PENA_CONT` y correspond à `E_N` et `COEF_PENA_FROT` à `E_T`.

3.3.6.2 Opérande INTEGRATION

```

♦ INTEGRATION          = /'NOEUD'                             [DEFAULT]
                        /'GAUSS'
                        /'SIMPSON'
                        /'SIMPSON1'
                        /'SIMPSON2'
                        /'NCOTES'
                        /'NCOTES1'
                        /'NCOTES2'

```

L'opérande 'INTEGRATION' permet de sélectionner une méthode d'intégration numérique pour les termes de contact et de frottement. Plusieurs méthodes sont implémentées ; 'NOEUD' pour un schéma d'intégration aux nœuds, 'GAUSS' pour le schéma classique de Gauss, 'SIMPSON' pour le schéma de Simpson (intégration aux nœuds et aux milieux des éléments) et 'COTES' pour un schéma adaptatif dans le cas du contact linéaire/quadratique. L'intégration aux nœuds est la plus générale et la plus efficace. Ce paramètre est à réserver aux **experts**.

3.3.6.3 Opérande COMPLIANCE

```

♦ COMPLIANCE           = /'NON'                                [DEFAULT]

```

◆	ASPERITE	=	/'OUI'	[DEFAULT]
◆	E_N	=	e_n	[DEFAULT]
◆	E_V	=	/0	[DEFAULT]
			/e_v	[R]

Cet opérateur permet d'activer le modèle de compliance. Ce modèle prend en compte les aspects microscopiques des surfaces (aspérités) et permet une régularisation du modèle de contact de Signorini. En dynamique, l'apport de ce modèle consiste en la possibilité d'introduire une densité de percussion amortissante qui correspond à la dissipation de l'énergie du choc. La loi de compliance introduite dans *Code_Aster* est une loi polynomiale (voir doc [R5.03.52]). Les trois paramètres de la loi de compliance sont ASPERITE, E_N et E_V.

3.3.6.4 Opérateur USURE

◆	USURE	=	/'SANS'	[DEFAULT]
			/'ARCHARD'	
◆	K	=	/usure_k	[R]
◆	H	=	/usure_h	[R]

Cet opérateur permet, lorsque le frottement est activé, d'utiliser le modèle d'usure d'Archard. Les deux paramètres de la loi d'Archard sont K et H.

3.3.6.5 Opérateur GLISSIERE

◆	GLISSIERE	=	/'NON'	[DEFAULT]
			/'OUI'	

Contrairement à la formulation discrète (voir § 21), il est possible d'activer cette option zone par zone .

3.3.7 Paramètres locaux (zone par zone) de la formulation XFEM

◆	ZONE	=	_F(paramètres locaux)
---	------	---	-----------------------

Les mots-clefs de ce paragraphe sont valables pour la formulation XFEM . Les paramètres sont définis par zone.

3.3.7.1 Opérateur FISS_MAIT

◆	FISS_MAIT	=	fiss_mait	[fiss_xfem]
---	-----------	---	-----------	-------------

Pour la formulation XFEM, l'utilisateur fournit la fissure de contact potentielle (FISS_MAIT). Ces fissures sont issues de l'opérateur DEFI_FISS_XFEM.

3.3.7.2 Coefficients de la formulation XFEM

◆	ALGO_CONT	=	/'STANDARD'	[DEFAULT]
			/'AVANCE'	
			/'PENALISATION'	
◆	COEF_CONT	=	/100.	[DEFAULT]
			/coef_cont	[R]
◆	COEF_REGU_CONT	=	/100.	[DEFAULT]
			/coef_regu_cont	[R]
◆	COEF_STAB_CONT	=	/100.	[DEFAULT]
			/coef_stab_cont	[R]
◆	COEF_PENA_CONT	=	/100.	[DEFAULT]
			/coef_pena_cont	[R]
◆	ALGO_FROT	=	/'STANDARD'	[DEFAULT]
			/'AVANCE'	
			/'PENALISATION'	

◆	COEF_FROT	=	/100.	[DEFAULT]
			/coef_frot	[R]
◆	COEF_REGU_FROT	=	/100.	[DEFAULT]
			/coef_regu_frot	[R]
◆	COEF_STAB_FROT	=	/100.	[DEFAULT]
			/coef_stab_frot	[R]
◆	COEF_PENA_FROT	=	/100.	[DEFAULT]
			/coef_pena_frot	[R]

La formulation XFEM est une écriture de type continue. Le choix des coefficients est le même que dans le cas de la formulation continue (§ 22).

3.3.7.3 Opérande GLISSIERE

◇	GLISSIERE	=	/'NON'	[DEFAULT]
			/'OUI'	

Contrairement à la formulation discrète (voir § 21), il est possible d'activer cette option zone par zone.

3.3.7.4 Opérande INTEGRATION

◇	INTEGRATION	=	/'FPG4'	[DEFAULT]
			/'FPG2'	
			/'FPG3'	
			/'FPG6'	
			/'FPG7'	
			/'NOEUD'	
			/'GAUSS'	
			/'SIMPSON'	
			/'SIMPSON1'	
			/'NCOTES'	
			/'NCOTES1'	
			/'NCOTES2'	

Ce mot-clé a la même signification que celui de la formulation CONTINUE, cf. §23. C'est un paramètre d'expert.

3.3.7.5 Opérande ALGO_LAGR

◇	ALGO_LAGR	=	/'VERSION1'	[DEFAULT]
			/'VERSION2'	
			/'NON'	

Il existe deux autres paramètres spécifiques à la formulation XFEM. Le premier est le choix de l'algorithme d'élimination des Lagrange de frottement pour satisfaire la condition LBB (voir [R7.02.12]). Ce paramètre est d'un usage destiné aux experts.

3.3.7.6 Opérande COEF_ECHELLE

◇	COEF_ECHELLE	=	/1.E6	[DEFAULT]
			/coef_echelle	[R]

La méthode XFEM étant basée sur une formulation hybride à deux champs (déplacements et pressions), dans certaines situations, une grande différence entre les différents termes conduit à une matrice de rigidité mal conditionnée. Le paramètre 'COEF_ECHELLE' permet d'appliquer un multiplicateur sur les termes correspondants aux Lagrangs de contact. Ce paramètre est d'un usage destiné aux experts.

3.3.7.7 Opérande RELATION

◇	RELATION	=	/'NON'	[DEFAULT]
			/'CZM_XFEM'	

Le mot-clé `RELATION='CZM_XFEM'` permet d'activer la prise en compte de forces de cohésion lors de l'ouverture d'une interface X-FEM. La cohésion est modélisée par la loi cohésive `CZM_EXP_REG` déjà existante en méthode des éléments finis classiques (voir [U2.05.07]).

Les données nécessaires du champ matériau sont fournies dans l'opérateur `DEFI_MATERIAU` [U4.43.01], sous le mot clé `RUPT_FRAG`.

3.4 Contrôle de l'algorithme de résolution

Toutes les formulations avec réactualisation géométrique utilisent un algorithme de type point fixe pour résoudre la non-linéarité d'appariement. Les algorithmes résolvant les problèmes en formulation `CONTINUE` ou `XFEM` utilisent trois boucles de résolution imbriquées pour résoudre la non-linéarité d'appariement, de seuil sur le frottement et de contact.

Il convient de noter que la formulation `XFEM` est particulière à deux points de vue :

1. Le fait d'activer ou pas la réactualisation géométrique (`REAC_GEOM`) entraîne deux formulations différentes (voir [R5.03.53]) : `XFEM` grands glissements et `XFEM` petits glissements.
2. Dans le cas où l'on est en `XFEM` grands glissements, il n'y a pas de boucle de point fixe sur les seuils de frottement et l'on résout le problème non-linéaire complètement (ce qui entraîne l'usage d'une matrice non-symétrique).

3.4.1 Boucle de réactualisation géométrique `REAC_GEOM`

◇ <code>REAC_GEOM</code>	=	<code>/'AUTOMATIQUE',</code> <code>/'CONTROLE',</code> <code>/'SANS'</code>	<code>[DEFAULT]</code>
◇ <code>ITER_GEOM_MAXI</code>	=	<code>/5,</code> <code>/iter_geom_maxi,</code>	<code>[DEFAULT]</code> <code>[I]</code>
◇ <code>RESI_GEOM</code>	=	<code>/0.05,</code> <code>/0.0001,</code> <code>/resi_geom,</code>	<code>[DEFAULT]</code> <code>[DEFAULT]</code> <code>[R]</code>
◇ <code>NB_ITER_GEOM</code>	=	<code>/2,</code> <code>/nb_iter_geom,</code>	<code>[DEFAULT]</code> <code>[I]</code>

Cet opérande indique sur quelle configuration géométrique est traité le problème de contact :

- `REAC_GEOM='AUTO'` : on réactualise automatiquement la géométrie *i.e.* le nombre de cycles « réactualisation géométrique-itérations » jusqu'à convergence n'est pas fixé par avance mais obéit à un critère interne de convergence géométrique. C'est l'option par défaut, conseillée pour résoudre correctement la non-linéarité d'appariement. Elle assure que les conditions de contact ont été imposées sur une configuration (initiale) qui diffère de moins de `resi_geom` de la configuration trouvée (5% par défaut pour la formulation `DISCRETE` ou 0.01% par défaut pour les formulations `CONTINUE` et `XFEM`).
- `REAC_GEOM='SANS'` : on travaille sur la géométrie initiale. Cette option n'est valable que dans l'hypothèse de petites perturbations ou quand la normale aux surfaces de contact ne change pas au cours du calcul.
- `REAC_GEOM='CONTROLE'` : lorsque le critère automatique ne parvient pas à être satisfait ou lorsque l'utilisateur souhaite une meilleure précision (c'est-à-dire inférieure à `resi_geom`), il doit contrôler lui-même la réactualisation géométrique et pour cela il doit indiquer :
 - `NB_REAC_GEOM=n` : C'est le nombre de cycles de réactualisations géométriques qui seront effectués par pas de charge. Plaçons-nous à un pas de charge donné :
 - La valeur 1 indique qu'à convergence, on réactualise la géométrie et on passe au pas de charge suivant.
 - La valeur 2 indique qu'à convergence, on ne passe pas au pas de charge suivant. On réactualise la géométrie et on réitère jusqu'à convergence.
 - La valeur $n > 2$ indique que l'on fait n cycles « réactualisation géométrique-itérations » jusqu'à convergence.

Si vous avez sélectionné `REAC_GEOM='AUTO'`, le paramètre `ITER_GEOM_MAXI` est le nombre maximal de cycles de réactualisations géométriques. Si le critère portant sur `RESI_GEOM` n'est pas satisfait au bout de `iter_geom_maxi` cycles, alors on s'arrête en erreur.

Remarque :

Si vous avez choisi une réactualisation contrôlée avec $n > 1$ et que Code_Aster détecte la nécessité d'une réactualisation géométrique, il vous en avertira par une alarme. Charge à l'utilisateur de décider si l'erreur commise (nécessairement supérieure à `resi_geom`) est acceptable ou non. Il y a en effet un risque d'erreur d'appariement (une maille a été appariée sur une configuration qui a bougé) et donc d'interpénétration. Dans le cas où l'on résout sur la géométrie initiale ou bien avec un seul cycle de réactualisation géométrique, il n'y a pas d'avertissement car on ne peut pas calculer d'erreur. L'utilisateur doit donc vérifier la validité de son choix.

3.4.2 Boucle de réactualisation du frottement REAC_FROT

◇ REAC_FROT	=	/'AUTOMATIQUE',	[DEFAULT]
		/'CONTROLE',	
		/'SANS',	
◇ ITER_FROT_MAXI	=	/2,	[DEFAULT]
		/iter_frot_maxi,	[I]
◇ RESI_FROT	=	/0.0001,	[DEFAULT]
		/resi_frot,	[R]
◇ NB_ITER_FROT	=	/2,	[DEFAULT]
		/nb_iter_frot,	[I]

Pour la formulation `CONTINUE` et la formulation `XFEM` (en petits glissements), le problème de Coulomb est résolu par une succession de points fixes sur le seuil de Tresca. Pour contrôler cette boucle, le mécanisme est le même que sur la boucle de réactualisation géométrique. Plusieurs remarques :

- Pour `XFEM` en grands glissements (`REAC_GEOM != 'SANS'`), il n'y a pas de boucle sur les seuils de Tresca ;
- Pour la formulation `DISCRETE`, chaque algorithme (`LAGRANGIEN`, `PENALISATION`) gère en interne sa propre boucle sur les seuils de Tresca (voir [R5.03.50]) ;
- Le problème de frottement de Coulomb étant résolu par une succession de boucles sur les seuils fixes de Tresca, il est possible de simuler une loi de type Tresca en renseignant `SEUIL_INIT` et en forçant le nombre d'itérations de frottement à 1 (`NB_ITER_FROT=1`).

3.4.3 Boucle de point fixe pour les statuts de contact

◇ ITER_CONT_TYPE	=	/'MAXI',	[DEFAULT]
		/'MULT'	
◇ ITER_CONT_MULT	=	/4,	[DEFAULT]
		/iter_cont_mult,	[I]
◇ ITER_CONT_MAXI	=	/30,	[DEFAULT]
		/iter_cont_maxi,	[I]

On peut contrôler la troisième boucle sur les statuts de contact de deux manières différentes :

- En donnant le nombre **absolu** maximum d'itérations de contact, `ITER_CONT_TYPE='MAXI'` puis `ITER_CONT_MAXI` ;
- En donnant le nombre **relatif** maximum d'itérations de contact par un coefficient multiplicateur, `ITER_CONT_TYPE='MULT'` puis `ITER_CONT_MULT` ; dans ce cas, le nombre d'itérations sur le statut de contact sera égal au produit `iter_cont_mult` par le nombre de nœuds esclaves.

Ces deux mécanismes correspondent à deux manières de gérer les statuts. Dans la méthode `CONTRAINTE`, le nombre maximum d'itérations de contact est donné par un résultat théorique (il est inférieur à deux fois le nombre de nœuds esclaves). Par extension, on peut utiliser ce mécanisme pour les formulations `DISCRETE` avec des méthodes de résolution de type `LAGRANGIEN`. Pour les autres formulations (`CONTINUE` et `XFEM`), les statuts sont changés par blocs et le nombre d'itérations maximum est donc plus logiquement renseigné en valeur absolue. Dans tous les cas, la modification de ces paramètres est réservée aux **experts** car elle peut conduire à des résultats **faux** si on le frottement est activé.

Si on dépasse le nombre maximum d'itérations de contact, on obtient le message d'erreur 'Echec dans le traitement du contact'. On peut alors tenter de raffiner le maillage, subdiviser le pas de temps, ou, en dernier recours, changer la valeur de `ITER_MULT_MAXI`.

3.4.4 Opérandes **CONTACT_INIT/SEUIL_INIT**

◇ <code>CONTACT_INIT</code>	=	<code>/'NON'</code>	[DEFAULT]
		<code>/'OUI'</code>	
◇ <code>SEUIL_INIT</code>	=	<code>/0.</code>	[DEFAULT]
		<code>/seuil_init</code>	[R]

Pour les formulations `CONTINUE` et `XFEM`, les opérandes `CONTACT_INIT` et `SEUIL_INIT` permettent de fixer respectivement les seuils de démarrage des boucles du contact et du frottement. En effet, l'hypothèse de base de la résolution par boucles imbriquées est de toujours chercher à résoudre d'abord un problème sans contact (tous les points sont supposés non contactants) et sans frottement (tous les points glissent). Il est néanmoins possible de forcer l'algorithme à prendre en compte le contact initial en renseignant `CONTACT_INIT='OUI'` et donner un seuil de glissement différent de zéro en renseignant `SEUIL_INIT`.

Attention `CONTACT_INIT='OUI'` active tous les statuts sans exception. S'il existe des jeux initiaux positifs, cette option peut donc engendrer des difficultés de convergence, voire un comportement inattendu.

3.5 Structure de données **VALE_CONT**

Toutes les méthodes de contact avec ou sans frottement produisent une structure de données de type `VALE_CONT` avec les composantes suivantes en chaque nœud esclave :

- `CONT` : indicateur de contact frottant
 - 0 : pas de contact
 - 1 : contact adhérent (uniquement si `FROTTEMENT='COULOMB'`)
 - 2 : contact glissant
- `JEU` : valeur du jeu
- `RN` : norme de la réaction normale de contact
- `RNX` : composante suivant DX de la réaction normale de contact
- `RNY` : composante suivant DY de la réaction normale de contact
- `RNZ` : composante suivant DZ de la réaction normale de contact
- `GLIX` : composante suivant t_1 du glissement tangentiel (repère local)
- `GLIY` : composante suivant t_2 du glissement tangentiel (repère local)
- `GLI` : norme du glissement tangentiel
- `RTAX` : composante suivant DX de la force tangentielle d'adhérence
- `RTAY` : composante suivant DY de la force tangentielle d'adhérence
- `RTAZ` : composante suivant DZ de la force tangentielle d'adhérence
- `RTGX` : composante suivant DX de la force tangentielle de glissement
- `RTGY` : composante suivant DY de la force tangentielle de glissement
- `RTGZ` : composante suivant DZ de la force tangentielle de glissement
- `RX` : composante suivant DX de la force de contact frottant ($RNX+RTAX+RTGX$)
- `RY` : composante suivant DY de la force de contact frottant ($RNY+RTAY+RTGY$)
- `RZ` : composante suivant DZ de la force de contact frottant ($RNZ+RTAZ+RTGZ$)
- `R` : norme de la force de contact frottant
- `HN` : jeu d'usure en formulation `CONTINUE`
- `I` : percussion de la résultante R de la force de contact frottant
- `IX` : percussion de la composante suivant DX de la force de contact frottant
- `IY` : percussion de la composante suivant DY de la force de contact frottant
- `IZ` : percussion de la composante suivant DZ de la force de contact frottant
- `PT_X` : coordonnées du point d'intersection en formulation `XFEM`
- `PT_Y` : coordonnées du point d'intersection en formulation `XFEM`
- `PT_Z` : coordonnées du point d'intersection en formulation `XFEM`

Elle s'imprime comme suit sous forme de table :

```
MATABLE=POST_RELEVE_T( ACTION=_F(INTITULE='INFOS FROTTEMENT',  
                                GROUP_NO='ESCLAVE',  
                                RESULTAT=U,  
                                INST=10.,  
                                TOUT_CMP='OUI',  
                                NOM_CHAM='VALE_CONT',  
                                OPERATION='EXTRACTION',),),);  
  
IMPR_TABLE(TABLE=MATABLE);
```

Il convient de remarquer que les quantités correspondant aux réactions de contact/frottement sont des efforts nodaux (en unité de force) au sens de la formulation éléments finis et non des pressions de contact. En modélisation axisymétrique, tout comme pour l'option **FORC_NODA** ou **REAC_NODA**, il faut donc multiplier les valeurs obtenues par 2π pour obtenir la résultante (cf. [U4.81.02]).

Plus généralement, pour avoir accès aux pressions de contact, deux méthodes sont possibles suivant la formulation :

- si l'on est dans le cas d'une formulation discrète, il faut récupérer les contraintes à l'intérieur des éléments finis (**SIEF_ELGA**) et les extrapoler aux nœuds sur la peau (il y a donc approximation).
- si l'on est dans le cas d'une formulation continue, la pression de contact et la pression de frottement sont accessibles directement comme inconnues du problème (composantes **LAGS_C**, **LAGR_F1** et **LAGR_F2** du champ **DEPL**). Attention la densité d'effort surfacique tangentielle s'obtient en multipliant **LAGR_F*** par **LAGS_C**.

3.6 Formulation **LIAISON_UNIL**

La formulation **LIAISON_UNIL** est utilisable pour définir une condition unilatérale (inéquation) de type nodal sur des degrés de liberté quelconques tel que $\sum \alpha_i(t) p_i < r(t)$ avec p_i la valeur du degré de liberté d'un nœud (déplacement, pression, température), $r(t)$ une valeur numérique imposée (fonction) et α_i un coefficient réel quelconque. On remarquera que le paramètre α_i permet aussi, si son signe est négatif, d'inverser le sens de l'inégalité.

3.6.1 Paramètres locaux (zone par zone) de la formulation **LIAISON_UNIL**

◆ **ZONE** = **_F(paramètres locaux)**

Les mots-clefs de ce paragraphe sont valables pour la formulation **LIAISON_UNIL**. Les paramètres sont définis par zone.

3.6.1.1 Opérands **MAILLE/GROUP_MA/NOEUD/GROUP_NO**

◆ /NOEUD	=	l_noeud	[l_noeud]
/GROUP_NO	=	l_grno	[l_gr_noeud]
/MAILLE	=	l_maille	[l_maille]
/GROUP_MA	=	l_grma	[l_gr_maille]

La condition unilatérale s'exprime sur les nœuds du maillage donnés sous les mot-clefs **NOEUD** ou **GROUP_NO**. On peut néanmoins donner les mailles portant les nœuds grâce aux mots-clefs **MAILLE** et **GROUP_MA**. Contrairement au cas du contact « classique », comme les conditions sont nodales, il est inutile d'orienter les normales aux mailles de peau dans le cas **LIAISON_UNIL**.

3.6.1.2 Opérands **SANS_NOEUD/SANS_GROUP_NO**

◆ SANS_NOEUD	=	l_snoeud	[l_noeud]
◆ SANS_GROUP_NO	=	l_sgrno	[l_gr_noeud]

Ces opérands permettent d'exclure des nœuds de la liste des nœuds de la même manière que pour les formulations **CONTINUE** ou **DISCRETE** (voir §14).

3.6.1.3 Opérande NOM_CMP

◆ NOM_CMP = l_cmp [l_TXM]

Liste des composantes p_i (degrés de liberté) sur lesquelles s'exerce la relation unilatérale. Ce peut être n'importe quel degré de liberté porté par le nœud. Par exemple : PRE, PRE1, PRE2, TEMP ou encore DX, DY ou DZ.

3.6.1.4 Opérandes COEF_IMPO et COEF_MULT

◆ COEF_IMPO = l_c_impo [fonction]
◆ COEF_MULT = l_c_mult [l_fonction]

COEF_IMPO est la valeur $r(t)$ imposée à droite de la relation unilatérale. COEF_MULT est la liste des coefficients multiplicateurs $\alpha(t)$ utilisés devant chacun des ddls. Les longueurs des listes COEF_MULT et NOM_CMP doivent bien entendu être identiques.

Les coefficients $r(t)$ et $\alpha(t)$ sont forcément des **fonctions**. Pour définir des coefficients constants, il conviendra d'utiliser la commande DEFI_CONSTANTE.

3.6.2 Exemple

On veut imposer la condition $1.3*PRE1-5.2*PRE2 < 4.0$, on aura alors :

```
coef_i = DEFI_CONSTANTE (VALE=4.0)
coef_m1 = DEFI_CONSTANTE (VALE=1.3)
coef_m2 = DEFI_CONSTANTE (VALE=-5.2)

NOM_CMP = ('PRE1', 'PRE2')
COEF_IMPO = coef_i
COEF_MULT = (coef_m1, coef_m2)
```