Continuation methods for non-linear analysis
FR : Méthodes de pilotage de chargement

Code_Aster, Salome-Meca course material
GNU FDL licence (http://www.gnu.org/copyleft/fdl.html)
Outline

- Definition of continuation methods
- Theoretical elements for continuation methods
- Solving non-linear problems with continuation methods
- Using continuation methods in Code_Aster
Definition of continuation methods
Continuation method – What?

Linear mechanical problem definition:
- Unknowns: displacement, Lagrange multiplier for boundary conditions
- Loadings: displacement (Dirichlet), forces (Neumann)
- Unique solution (elliptical differential equation): \( Ku = F \) if \( \det K \neq 0 \)

Non-linear problem definition:
- Unknowns: displacement, Lagrange multiplier for boundary conditions, temperature, pressure, stress and internal variables
- Loadings: displacement (Dirichlet), forces (Neumann), contact/friction
- Parameterization: \( t \) is not real time (quasi-static problem)
- Sequence of linearized solutions

General non-linear continuation method
- Some external loading and prescribed displacements are partially unknowns by user: directions are known, intensity are unknown \( \rightarrow \) continuation method
Continuation methods

Example 1: tensile stress test of the notched specimen

Isotropic fragile damage law

(ENOD_ISO_T_BETON)
Continuation method – Where?

Where to use:

1. External loading and prescribed displacements are partially unknowns by user: direction, application are **known**, intensity are **unknown**

2. Solution of an unstable problem => impossibility to follow system evolution continuously => **Newton method fails**

---

**1.** Experimental setup controlled by extensometer

- Applied force at point A is controlled by displacement at point B

**2.** Damage of the notched bar

- Sound state
- Completely damaged state
Continuation method – Loading control

Nooru-Mohamed concrete fracture test:
1) Loading is applied via rigid mobile platform by piston displacement
2) Efforts are controlled on the stable effort control cell
3) Unstable concrete fracture

Rigid mobile platform
Hydraulic Piston
Tested Sample
Effort control cell

2m
Continuation method – Instability = snap-back

Bi-material with close characteristics

In 1d tensile test “weak-chain” is damaged first

Solution post peak: equilibrium => stress equality

\[
\sigma = E\varepsilon_1 = E(\varepsilon_f - \varepsilon_2) ; \quad \varepsilon_1 = \frac{\Delta l_1}{l_1}, \quad \varepsilon_2 = \frac{\Delta l_2}{l_2}
\]

\[
\Delta l \equiv \Delta l_1 + \Delta l_2 = l_1\sigma / E + l_2(\varepsilon_f - \sigma / E)
\]

Global post peak force-displacement response:

\[
\sigma = E(\Delta l - l_2\varepsilon_f) / (l_1 - l_2)
\]

Snap-back if \( l_1 > l_2 \)

\( l_1 << l_2 \)
Continuation method – Instability = snap-back

Damage of the bar

Global force-displacement response:

Simplified damage law:

\[
\sigma = \begin{cases} 
E \varepsilon & \text{si} \quad \varepsilon < \varepsilon_c \\
E(\varepsilon_f - \varepsilon) & \text{si} \quad \varepsilon > \varepsilon_c 
\end{cases}
\]
Continuation methods: General FU curve

General form of the force-displacement curve

Example: Snap-through for shell buckling

BC Prescription: forces
- Multiplicity: one force -> several displacements

BC Prescription: displacements
- Multiplicity: one displacement -> several forces
- Horizontal tangent matrices: singular (slope vanishing)
Continuation method – Why?

Continuation method in non-linear problems:

- Choosing a solution for incomplete model
  \[ F^{ext} = F^{ext}_{impo} + \eta F^{ext}_{piolo} \]

- Yield-point analysis: critical loading
- Follow physical solution for “ill defined” problem:
  \[ K^n \delta u^{n+1} = R^n \quad \text{si} \quad \det K^n = 0 \]

Multiple solutions coming from computation

- From constitutive laws. Non-elliptic condition as softening, damage, geo-mechanic laws:
  \text{ENDO\_SCALAIRE, ENDO\_ISOT\_BETON, ENDO\_ORTH\_BETON, CZM\_EXP, ROUSSELIER, VENDOCHAB, ...}
- From equilibrium equations: buckling, structural instabilities
- From Coulomb’s friction
Continuation methods

Choosing single solution for partially defined loading

- Partially unknown loadings: direction, application are **known**, intensity are **unknown**

- **Goal:** $u$
- **Parameter:** intensity of force $\eta$
- **Direction of force:** known !

- **Goal:** all domain $\Omega$ is plastified
- **Parameter:** intensity of pressure $\eta$
- **Yield-point analysis**
Solving mechanical problems with continuation
Solving linear problem with continuation

Electric pylon stability control

- Boundary conditions: known and unknown parts of prescribed forces

- Goal: $u$
- Parameter: intensity of force $\eta$
- Direction of force: known!

Linear Elasticity:

$$Ku = F^{\text{ext}}$$

where

$$F^{\text{ext}} \equiv F_{\text{impo}}^{\text{ext}} + \eta F_{\text{pilo}}^{\text{ext}}$$

Formal solution:

$$u = K^{-1}F_{\text{impo}}^{\text{ext}} + \eta K^{-1}F_{\text{pilo}}^{\text{ext}} = u_{\text{impo}} + \eta u_{\text{pilo}}$$

Cable length control ⇒ new equation for $\eta$

$$\|u\| = \|u_{\text{impo}} + \eta u_{\text{pilo}}\| = \Delta \tau_{\text{crit}}$$
Solving non-linear problems with continuation

- Boundary conditions: known and unknown parts of prescribed displacement or prescribed forces

\[
\begin{align*}
\{L^{ext}\} &= \{L^{ext}_{impo}\} + \eta \{L^{ext}_{pilo}\} \\
\{u^d\} &= \{u^d_{impo}\} + \eta \{u^d_{pilo}\}
\end{align*}
\]

- New unknown \(\eta\) \(\Rightarrow\) new equation

\[
P\{u\} = \Delta \tau
\]
Solving non-linear problems with continuation

- Newton's method for equilibrium equation:

\[
[K]\{\delta u\} = \{L_{\text{ext}}^{\text{impo}}\} + \eta \cdot \{L_{\text{ext}}^{\text{pilo}}\} - \{L^{\text{int}}\}
\]

- Solution: linear continuation equation \( \Rightarrow \) separation

\[
\{\delta u\} = \{\delta u_{\text{impo}}\} + \eta \cdot \{\delta u_{\text{pilo}}\}
\]

- Two parts of the solution (independent)

\[
\{\delta u_{\text{impo}}\} = [K]^{-1} \cdot \left(\{L_{\text{ext}}^{\text{impo}}\} - \{L^{\text{int}}\}\right)
\]

\[
\{\delta u_{\text{pilo}}\} = [K]^{-1} \cdot \left(\{L_{\text{ext}}^{\text{pilo}}\} - \{L^{\text{int}}\}\right)
\]
Solving non-linear problems with continuation

How to find unknown load parameter $\eta$?

Continuation equation:
- Build on displacements, strain or stress
- Should be easy to solve (linear, quadratic)
- Using only one scalar parameter

Continuation methods list by goal function:
- Degree of freedom: \texttt{DDL\_IMPO}
- Norm of displacement: \texttt{LONG\_ARC}
- Displacement jump: \texttt{SAUT\_IMPO} (XFEM)
- Norm of displacement jump: \texttt{SAUT\_LONG\_ARC} (XFEM)
- Work of exterior forces (yield-point analysis): \texttt{ANA\_LIM}
- Strain increment: \texttt{DEFORMATION}
- Elastic prediction: \texttt{PRED\_ELAS}

\[
\|\mathbf{u}\| = \|\mathbf{u}_{\text{impo}} + \eta \mathbf{u}_{\text{pielo}}\| = \Delta \tau_{\text{crit}}
\]

\[
P(\{\Delta u\}) = \frac{\Delta t}{C}
\]
Solving non-linear problems with continuation

- Continuation by degree of freedom - dof (DDL_IMPO)
- Equation

\[ P(\{\Delta u\}) = \Delta u^{dof} = \frac{\Delta t}{C} \]

- Using rules:
  - Control displacement \textit{increment} of one dof
  - The controlled node must be important for movement
  - \( C \) is a constant given by user in \texttt{STAT\_NON\_LINE}

Good goal: what is \( \eta \) for a given vertical displacement of A node?

Bad goal: what is \( \eta \) for a given displacement of B node?

\textbf{B doesn’t move!}
Solving non-linear problems with continuation

- Continuation by norm of displacement (**LONG_ARC**) – Extended RIKS method (1972)
- Equation

\[ P(\{\Delta u\}) = \|\Delta u\| = \frac{\Delta t}{C} \]

- Using rules:
  - Control **norm** displacement **increment** of **several** dof and **several** nodes
  - The controlled nodes must be important for movement
  - \( C \) is a constant given by user in **STAT_NON_LINE**
  - Resulted equation is quadratic: two solutions -> need selection criterion **RESIDU**, **ANGL_INCR_DEPL**, **NORM_INCR_DEPL** (see documentation)
Solving non-linear problems with continuation

Continuation by norm of displacement (LONG_ARC) – Extended RIKS method

Arc-length: construct successive circles to follow loading path

Very useful for complex path (snap-through for instance)

\[ P(\{\Delta u\}) = \|\Delta u\| \Rightarrow \|\Delta u_{impo} + \eta . \Delta u_{pilo}\|^2 = \left(\frac{\Delta t}{C}\right)^2 \]

Quadratic equation for \(\eta\)
Solving non-linear problems with continuation

- Continuation by strain increment (DEFORMATION)

- Equation

\[ P(\{u\}) = \text{Max}_{\text{gauss}} \left( \frac{\varepsilon_i^g : \Delta \varepsilon_i^g}{\| \varepsilon_i^g \|} \right) = \frac{\Delta t}{C} \]

- \( \{ \varepsilon_i^g \} \) Strains at the previous load step
- \( \{ \Delta \varepsilon_i^g \} \) Increment of strains at the current step

- Using rules
  - At least, one point where strain is increasing
  - No indication on plasticity state
  - Need a reference state with deformation (\( \| \varepsilon_i^g \| \neq 0 \)): first computation without continuation method to establish this state
  - Impossibility to follow the snap-back solutions: impossible loading-unloading transition.
Solving non-linear problems with continuation

- Continuation by strain increment (DEFORMATION)

**Equation**

\[
P(\{u\}) = \text{Max}_{\text{gauss}} \left( \frac{\varepsilon_{i-1}^g : \Delta \varepsilon_{i-1}^g}{\|\varepsilon_{i-1}^g\|} \right) = \frac{\Delta t}{C}
\]

\[
\begin{align*}
\{\varepsilon_{i-1}^g\} & \quad \text{Strains at the previous load step} \\
\{\Delta \varepsilon_{i}^g\} & \quad \text{Increment of strains at the current step}
\end{align*}
\]

**Using rules**

- Resultant equation could have two solutions =>
- =>need selection criterion

RESIDU, ANGL_INCR_DEPL, NORM_INCR_DEPL
(see documentation)
Solving non-linear problems with continuation

- Continuation method by elastic prediction ([PRED_ELAS])
- Available for Yield function constitutive laws: plasticity, damage
- Equations:

\[ P\{u\} = \text{Max}_{gauss} \left( \varphi_{gauss}\left(d_{i-1}^g, \varepsilon_{i-1}^g + \Delta\varepsilon^g\right) \right) = \frac{\Delta t}{C} \quad \text{for elasto-plasticity laws} \]

\[ P\{u\} = \text{Max}_{gauss} \left( \varphi_{gauss}\left(d_i^g + \frac{\Delta t}{C}, \varepsilon_{i-1}^g + \Delta\varepsilon^g\right) \right) = 0 \quad \text{for damage laws} \]

\[ \{\varepsilon_{i-1}^g\} \quad \text{Strains at the previous load step} \]

\[ \{\Delta\varepsilon^g\} \quad \text{Increment of strains at the current step} \]

\[ \{d_{i-1}^g\} \quad \text{Damage at the previous step} \]

- \( \Delta t \) control either magnitude of Yield function overflow, or damage increment.
Solving non-linear problems with continuation

- Continuation method by elastic prediction (**PRED_ELAS**)

- Using rules
  - At least, one point which passes through initial yield surface
  - Depend on behavior law: **ENDO_SCALAIRE, ENDO_FRAGILE, ENDO_ISOT_BETON, ENDO ORTH_BETON, VMIS_ISOT_*, CZM_*** and **BETON_DOUBLE_DP**
  - Criterion C: increasing ratio of damage or strain
  - Resultant equation should have two solutions -> need selection criterion **RESIDU, ANGL_INCR_DEPL, NORM_INCR_DEPL** (see documentation)
Using continuation methods in *Code_Aster*
Using continuation methods in *Code_Aster*

- As continuation methods is using parameter for determination of loading path, you must avoid direct or indirect using of time in your model:
  - No dynamic (only `STAT_NON_LINE` where *t* is *pseudo*-time)
  - No time for loadings: no `FONC_MULT`, no `AFFE_CHAR_MECA_F` with parameter `INST`
  - No « command variables » as temperature in `AFFE_MATERIAU/AFFE_VARC`

- Contact/friction is **not possible** except for specific XFEM methods (with CZM, see documentation) or discrete element (`DIS_CHOC`)

- Line search is possible only for some continuation methods
Using continuation methods in Code_Aster

- Definition of loads in **AFFE_CHAR_MECA**

- Definition of continuation load in **STAT_NON_LINE/EXCIT**
  
  EXCIT/TYPE_CHARGE='**FIXE_PILLO**'

- Definition of the parameters for continuation method in **STAT_NON_LINE/PILOTAGE**

- Post-processing: parameter could been found in result (**ETA_PILOTAGE**)
Continuation method: command file

Fixed charge

Driven charge

Pseudo-time

Resulting charge

Application

Continuation parameters

```c
# ETABLISSEMENT DE CONDITIONS LIMITES:
# AXE REVU ET REDUITE HORIZONTAL.
> CHARSY=AFPE_CHAR_MECA(MODELE=MO,
79  DDL_IMPO=_F(GROUP_MA='AXE', DX=0.));

# ETABLISSEMENT DE CONDITION DU DX CHARGEMENT PILOTE EN DEPLACEMENT
> EFFORT=AFPE_CHAR_MECA(MODELE=MO,
83  FACE_IMPO={_F(GROUP_MA='HAUT', DY=1),
84  _F(GROUP_MA='BAS', DY=-1)}
85  );

# DEFINITION DE LISTE DE PAS DE TEMPS
# SÉRIE LINAIRE 0,1,2,...INS_FIN
# la valeur de pas de temps (dT) sert à définir
# l'incrémentation d'endommagement en pilotage delta_d = dT/COEF_MUL
2 TEMPS=DEFI_LIST_REEL(DEBUT=0,
93  INTERVALLE=_F(JUSQU_A=1000,
94  NOMBRE=1000));

EVAL = STAT_NON_LINE(
97  MODELE=MO,
98  CHAM_MATER=CHMAT,
99  EXCIT={_F(CHARGE=CHARSY)
101  _F(CHARGE=EFFORT),
102  },
103  TYPE_CHARGE='FIXE_PILO')

COMP_INCR=_F(RELATION='ENDO_SCALAIRE',TOUT='OUI'),
105  INCRÉMEN=_F(LIST_INST=TEMPS),
106  NEWTON=_F(MATRICE='TANGENTE'),
107
115  PILOTAGE=_F(TYPE='REDU_ELAS',
119  COEF_MULT=10.,
120  ETA_PILO_MAX=etaMax,
121  ETA_PILO_R_MAX=2.,
122  ETA_PILO_R_MIN=0.,
123  SELECTION='MIXTE',
124  PROJ_BORNES = 'OUI',
125  );
```
Continuation methods: PRED_ELAS

Example 1: tensile stress test of the notched specimen

Isotropic fragile damage law (ENDO_ISOT_BETON)
Continuation methods: PRED_ELAS

Example 2: Impact on damaged concrete

Anisotropic fragile damage law (ENDO_ORTH_BETON)
Continuation methods : PRED_ELAS

Example 3 : tensile stress test for the perforated plate

Cohesive Zone Model (CZM)
Continuation methods: LONG_ARC

Example 4

Buckling of a shell/column

SSNS101B: comparaisons Aster - Inca - ref[2]
Continuation methods: DDL_IMPO

Example 5: stability of the gravity dam

Cohesive Zone Model (JOINT_MECA RUPT)
Using continuation methods in Code_Aster

Documentation:
- General documentation about non-linear solver [R5.03.01]
- General documentation about continuation methods [R5.03.80]
- Using continuation method, syntax in [U4.51.03]

Examples:
- See [V6.03.114], forma03d test-case for general example
- See [V6.01.101], ssna119b test-case for fragile damage (Elastic prediction)
- See [V6.04.124], ssnv124 test-case for yield-point analysis
- See [V6.05.101], ssns101 test-case for shell buckling (Riks method)
End of presentation

Is something missing or unclear in this document?
Or feeling happy to have read such a clear tutorial?

Please, we welcome any feedbacks about Code_Aster training materials.
Do not hesitate to share with us your comments on the Code_Aster forum dedicated thread.