SERVICE NEWSLETTER

ANSYS: Wissenswertes zum Lagrange-Kontaktalgorithmus

Ausgabe: 05 / 2004

Problem:

Dem ANSYS-Anwender stehen bei der Lösung von Kontaktproblemen
mit der KEYOPTION(2) fünf verschiedene Kontaktalgorithmen zur Verfügung. Auf die älteren Kontaktalgorithmen

- Augmented Method KEYOPT(2)=0
- Penalty Method KEYOPT(2)=1
- MPC Algorithm KEYOPT(2)=2

soll hier nicht mehr im Detail eingegangen werden. Wir konzentrieren uns in diesem Beitrag auf die beiden neuen Algorithmen

- Lagrange & Penalty KEYOPT(2)=3
- Lagrange Method KEYOPT(2)=4

Erläuterung:

Die Kontaktbedingung, die bei der Lösung von Kontaktproblemen immer
zu lösen ist lautet:

Die relative Bewegung von zwei sich in Kontakt befindlichen Oberflächen zueinander soll Null sein und zwar in normaler und tangentialer Richtung.

In Normalenrichtung heißt das, dass sich die Oberflächen möglichst gegenseitig nicht durchdringen sollen. In Tangentenrichtung bedeutet es, das beim Auftreten von Haftreibung möglichst kein Schlupf stattfinden soll.
Wir haben bei Kontaktproblemen also immer das strukturmechanische
Verhalten in normaler und tangentialer Richtung zu unterscheiden. Das machen auch die beiden neuen Algorithmen, *Langrange & Penalty* auf der einen und die *Langrange Method* auf der anderen Seite.
.

ANSYS: Wissenswertes zum Lagrange-Kontaktalgorithmus

Ausgabe: 05 / 2004

Erläuterung:

Wir wollen zunächst die prinzipiellen Unterschiede zwischen *Lagrange & Penalty* und der *Langrange Method* erklären:

<u>Lagrange & Penalty:</u>

In Normalenrichtung ist die Durchdringung immer (numerisch) Null. Man
benötigt also hier nicht mehr die Kontaktsteifigkeit in Normalenrichtung, die
mittels REAL-Konstante FKN vorgegeben werden kann und womit man beispielsweise bei der *Augmented Lagrange* oder *Penalty Method* die
verbleibende Durchdringung beeinflusst.

In Tangentenrichtung wird allerdings eine Penalty Formulierung verwendet. Das
Ergebnis in tangentialer Richtung wird hier also sehr wohl noch von der Kontaktsteifigkeit in Tangentenrichtung beeinflusst, welche man durch
die REAL-Konstante FKT angibt.

Folglich kann selbst bei Haftreibung (Kontaktstatus Sticking) eine tangentiale Relativbewegung, also Schlupf, numerisch enstehen.

<u>Lagrange Method</u>

Hier liegt eine reine Lagrange Formulierung in normaler und tangentialer
Richtung vor. Die REAL-Konstanten FKN und FKT, also die Kontaktsteifigkeit
in normaler und tangentialer Richtung werden nicht mehr benötigt.

Bei dieser Methode wird numerisch weder eine Durchdringung in Normalenrichtung noch ein Schlupf bei Haftreibung in Tangentenrichtung entstehen.

ENGINEERING SERVICES
CADFEM

ANSYS: Wissenswertes zum Lagrange-Kontaktalgorithmus

Ausgabe: 05 / 2004

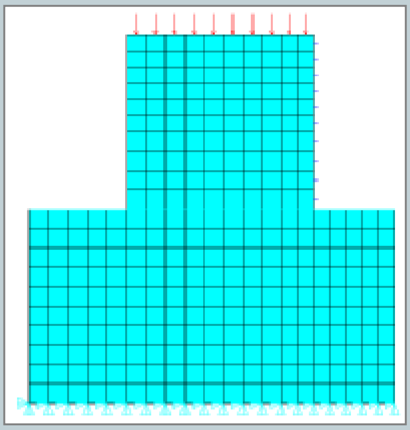
Prinzip-Beispiel: Ein durch Reibung gehaltene Scheibe in 2D

Eine Scheibe der Abmessung 10 mm mal 10 mm wird mit einem Druck von
10 MPa in einem ersten Lastfall auf eine Unterlage gedrückt. Die resultierende
Druckkraft ist demnach 100 N. In einem zweiten Lastfall wird an ihrer rechten
Kante mit einem Zug von 2 MPa gezogen, also mit 20 N. Da der Reibkoeffizient
gerade 0.25 beträgt, liegt nach dem Coulomb'schen Gesetz

R = m * F = 0.25 * 100 = 25

eine Wiederstandskraft von 25 N vor. Da wir nur mit 20 N von der Seite
belasten, muss die Scheibe noch im Zustand der Haftreibung sein.

Das Problem wurde mit quadratischen Elementen in ANSYS 8.1 abgebildet.



ANSYS: Wissenswertes zum Lagrange-Kontaktalgorithmus

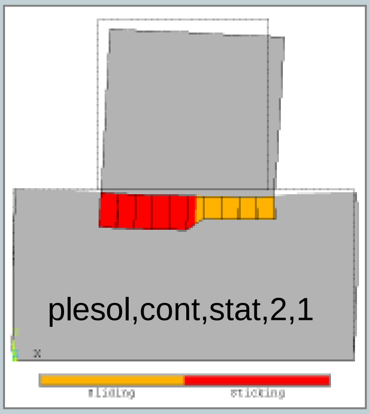
Ausgabe: 05 / 2004

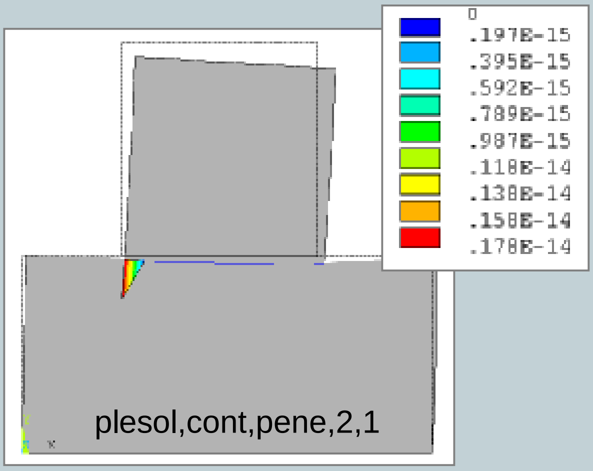
Ergebnis:

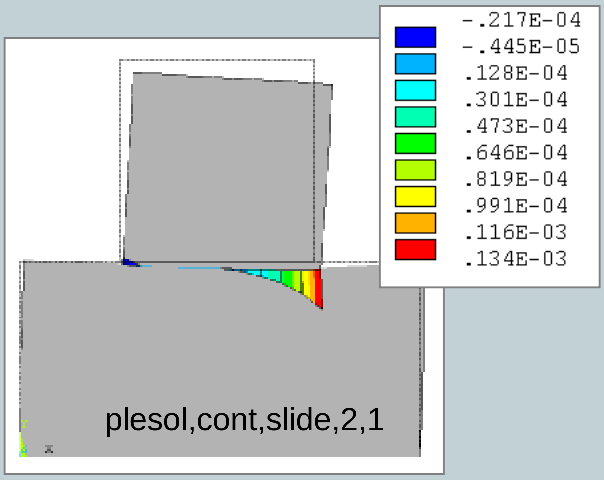
Auf den folgenden Seiten wird jeweils das Ergebnis für den Kontakt-Status, sowie für die Durchdringung und den Schlupf dargestellt. Beachten Sie, dass die Durchdringung bei Berührung und der Schlupf bei Haftreibung niemals exakt Null sein kann. Wir begnügen uns mit (numerisch) Null.
Dieses Ergebnis wird mit der *Lagrange Method* sofort berechnet. Wählt man *Lagrange & Penalty*, so wird das korrekte Ergebnis nur mit einem sehr großen Wert für die Kontaktsteifigkeit in Tangentialrichtung *FKT* erzielt.

Ergebnis: *Lagrange Method* (1000 fach überhöht)

In der Tat ist noch ein Teil der Struktur im Status Sticking, wie erwartet:







Die ***Durchdringung*** und der ***Schlupf im Bereich Sticking*** wird mit dieser Methode ***zu Null*** berechnet:

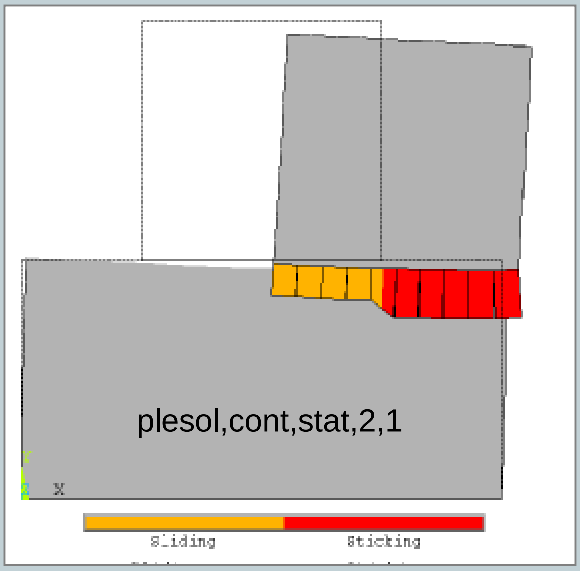
Durchdringung **Schlupf**

ANSYS: Wissenswertes zum Lagrange-Kontaktalgorithmus

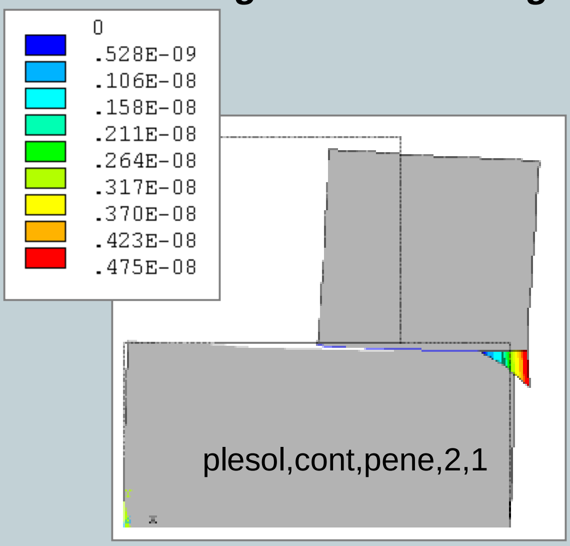
Ausgabe: 05 / 2004

Ergebnis: *Lagrange & Penalty* (1000 fach überhöht) à *<u>FKT=0.1</u>*

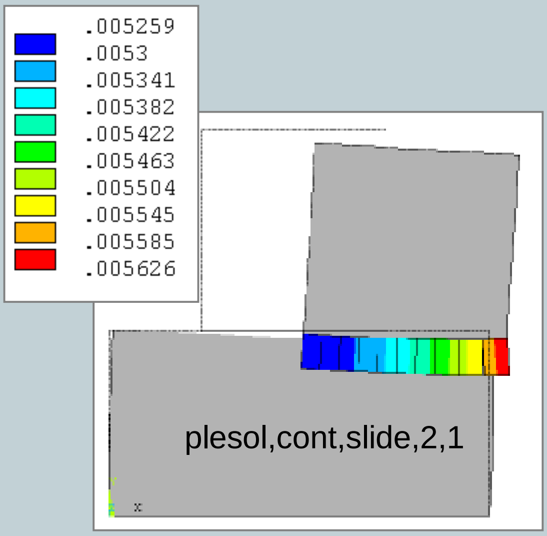
Hier hängt das Ergebnis sehr stark von der Wahl der Kontaktsteifigkeit in tangentialer Richtung ab. Wir beginnen mit FKT=0.1



Haftreibung ist hier deutlich nicht mehr zu erkennen. Die ***Durchdringung wird zu Null*** berechnet, der ***Schlupf im Bereich Sticking ist deutlich zu groß.***



Durchdringung

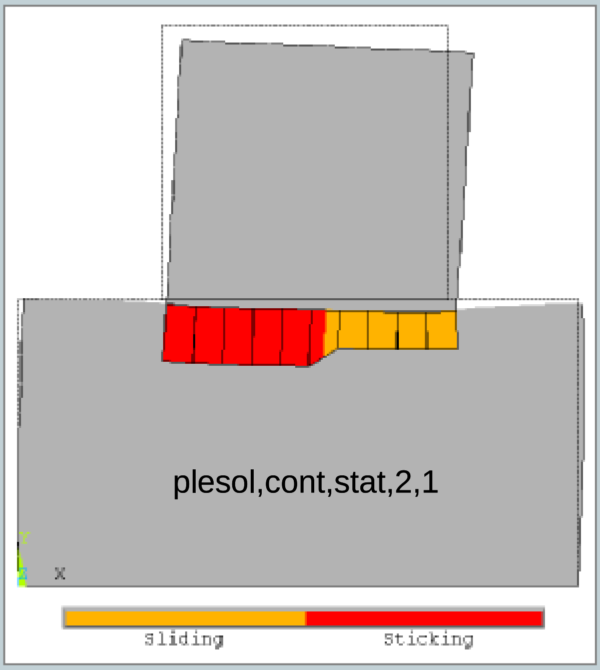


Schlupf

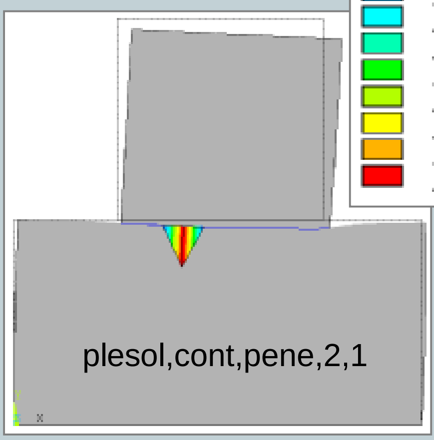
ANSYS: Wissenswertes zum Lagrange-Kontaktalgorithmus

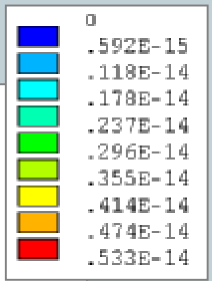
Ergebnis: *Lagrange & Penalty* (1000 fach überhöht) - *<u>FKT=1000</u>*

Wir rechnen jetzt mit FKT=1000

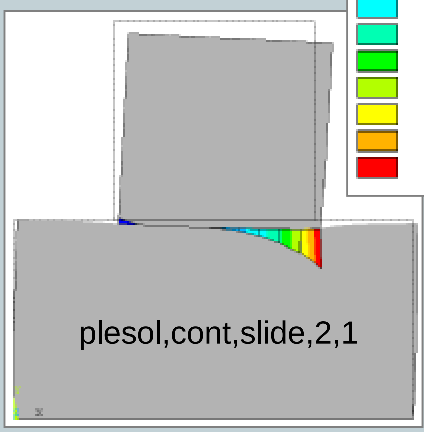


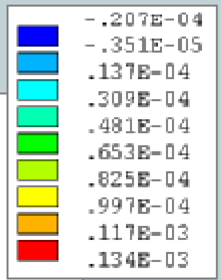
Haftreibung ist hier wieder deutlich zu erkennen. Die ***Durchdringung wird zu Null*** berechnet, der ***Schlupf im Bereich Sticking ist vernünftig klein.***





Durchdringung





Schlupf

ANSYS: Wissenswertes zum Lagrange-Kontaktalgorithmus

Eingabe-Datei für das Prinzip-Beispiel:

```
finish
/clear

KT_STIFF=1000.0
KT_STIFF=   0.1

/prep7
rectn,0,20, 0,10
rectn,5,15,10,20
et,1,182
mp,ex,1,210000
mp,prxy,1,0.3
mp,mu,2,0.25

esize,1
amesh,1
esize,1
amesh,2
eplot
```

```
et,2,169
et,3,172
keyopt,3,2,3
r,2,0,0,0,0,0,0,
rmore,,,,,,KT_STIFF

lsel,s,line,,3
nsll,s,1
type,2
real,2
mat,2
esurf

lsel,s,line,,5
nsll,s,1
type,3
real,2
mat,2
esurf

finish
```

```
/solu
nsel,s,loc,y,0
d,all,all,0
allsel
sfl,7,pres,10

time,10
autots,off
nsubst,10
outres,all,all
solve

sfl,6,pres,-2.0
time,20
autots,off
nsubst,10
solve

finish
```

```
/post1
/dscale,1,1000

set,last
plesol,cont,sfri,2,1
plesol,cont,slide,2,1
plesol,cont,pene,2,1
plesol,cont,stat,2,1
```

ANSYS: Wissenswertes zum Lagrange-Kontaktalgorithmus

Weitere Erläuterungen

1.)

Beachten Sie, dass bei der Verwendung eines Lagrange-Algorithmus‘ die KEYOPTION(4) immer größer als Null zu setzen ist. Das bedeutet, das die Kontaktprüfung nicht mehr am Gauß-Punkt sondern stets am Knoten des Elementes passiert.
Wenn man dieses nicht explizit setzt, so stellt ANSYS automatisch die KEYOPTION(4) auf den Wert 2 und der Algorithmus kann rechnen.

2.)

Ein Vorteil eines Lagrange-Algorithmus‘ zur Lösung von Kontaktproblemen ist, dass man ohne die Kontaktsteifigkeit rechnen kann. Die Kontaktsteifigkeit in normaler Richtung FKN wird bei keinem der hier diskutierten Verfahren mehr benötigt. Lediglich bei *Lagrange & Penalty* wird die Kontaktsteifigkeit in tangentialer Richtung noch benötigt.
Sie wissen sicher, dass die Kontaktsteifigkeit bislang der wichtigste Parameter zur Lösung von Kontaktproblemen war, jedenfalls, wenn man mit einem Algorithmus gearbeitet hat, der auf sie zurückgreifen musste, zum Beispiel die *Penalty Method* oder die *Augmented Method*.

Die Kontaktsteifigkeit beeinflusst gleichermaßen das Konvergenzverhalten und die Genauigkeit. Grosse Werte für die Kontaktsteifigkeit verbessern die Ergebnisgenauigkeit, verschlechtern jedoch leider die Konvergenz.
Die Lagrange-Verfahren zeigen daher auch meist ein besseres Konvergenzverfahren bei gleichzeitiger guter Ergebnisqualität.

ANSYS: Wissenswertes zum Lagrange-Kontaktalgorithmus

Ausgabe: 05 / 2004

Weitere Erläuterungen

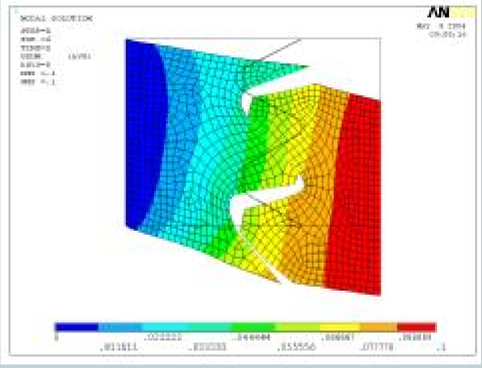
3.)
Bei dem momentanen Entwicklungsstand empfehlen wir die Lagrange-Verfahren für 2D-Probleme. Ferner haben Test-Rechnungen gezeigt, dass je nach Reibkoeffizient gerade die Lagrange Method auch einmal sehr lange rechnen kann. Insofern ist hier noch ein gewisses Entwicklungspotential vorhanden.

Für kleine 3D Probleme kann der Lagrange Algorithmus durchaus sinnvoll eingesetzt werden. Bei größeren Modellen zeigt sich aber ein deutlicher Effizienznachteil gegenüber den klassischen Methoden.

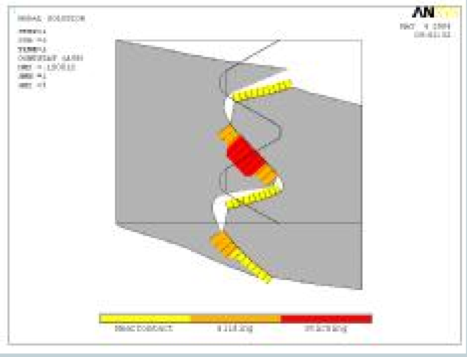
Schrauben-Beispiel in 2D:

Wir zeigen hier abschließend noch ein Beispiel, was mit *Lagrange & Penalty* und mit der *Lagrange Method* hervorragendes Konvergenzverhalten gezeigt hat und zwar mit den voreingestellten Parametern. Die *Augmented Method* und die *Penalty Method* konvergierten in Voreinstellung nicht.

Es handelt sich um ein Detail aus einer Schraubenverbindung, der Reibkoeffizient beträgt in diesem Beispiel μ = 0,1. Die Ergebnisse sind 10 fach überhöht dargestellt. Die Rechnung ist verschiebungsgesteuert:



Verschiebung



Kontakt-Status