Tips and tricks for successful implicit analyses with LS-DYNA

Tobias Erhart
Stuttgart, 23. Februar 2016
**Explicit vs. Implicit**

**Explicit:** \[ x_{n+1} = f(x_n, \ldots) \]

- \( \text{Ma}_n = f_n^{\text{ext}} - f_n^{\text{int}} \)
- solution: directly
- decoupled: efficient, fast
- many small time steps
- conditionally stable (Courant)
  
  **short time dynamics:**
  high frequency response, wave propagation

  **impact, crash, ...**

**Implicit:**

\[ f(x_{n+1}, x_n, \ldots) = 0 \]

- \( \text{M}_n a_{n+1} + K_n u_{n+1} = f_n^{\text{ext}} - f_n^{\text{int}} - \text{Ma}_n \)
- solution: iteratively
- linearization necessary
- few large time/load steps
- unconditionally stable
  
  **equilibrium! convergence?**

**structural dynamics:**

low frequency response, vibration, oscillation

  **earthquake, machines, ...**
Introduction

Explicit vs. Implicit

- Explicit inevitably includes inertial effects and resolves high frequencies whether you want it or not
- Implicit can neglect inertial effects and the selected time step size determines the resolved frequency spectrum

Consequences for FE models

- "cleaner" models in implicit for the sake of convergence, e.g. no initial penetrations, smooth material curves, contact, accuracy, …
- expensive features are not so expensive anymore
- no restriction on element size (time step size) in implicit
- often more work to get "normal termination" in implicit

"Explicit is handcraft - implicit is skill"
Guidelines

Troubleshooting convergence problems

Convergence behavior is depending on the physics of the problem

- difference in physics → different method(s) for solving convergence issues
Guidelines

Possible reasons for convergence problems

Mesh
- Coarse meshes may result in poor element geometry and bad contact behavior

Time/load step size
- The applied load/displacement etc. in a single step may be too large or small

Rigid body motions
- Unconstrained d.o.f. due to missing BC/SPC, initial contact gaps, beams, …

Contact
- Initial penetrations, too large step sizes, large forces, …

Material properties
- “rough“ data, softening properties, discontinuities in curves, incompressibility, …
Recommendations

Use double precision of the code ( _d_ in the name)
- required for accurate linear analysis
- improved convergence behavior in nonlinear analysis

Use the most recent LS-DYNA version possible (e.g. R9 beta)
imPLICIT functionality is rapidly improving

Use command line option "memory=" to run job in-core
Verify using LPRINT=1 on *CONTROL_IMPLICIT_SOLVER or "<ctrl-c> lprint". The CPU penalty for out-of-core can be as high as 100 times the in-core simulation!!

Read Appendix P in the User’s manual and Chapter 37 of the latest draft version of the Theory Manual
Nice summary about LS-DYNA’s Implicit Solver
Guidelines

Recommendations

Element types
● for solids use type 1, -1, -2, 13, or 16 elements for non-linear analysis
● for shells use type 6, 16, or -16 elements for non-linear analysis
● try to avoid pentahedral solid elements

Contact
● try to avoid initial penetrations or try IGNORE=1
● use press-fit option (IGNORE=3/4) for intended initial penetrations
● switch contacts to tied (temporarily) in order to identify problems
● use Mortar contacts or try IGAP=2
● decrease contact stiffnesses, observe penetrations
● contact often requires small time steps in implicit, too
● make sure that finer mesh is slave side
● turn off viscous damping with **VDC=0.0**
Recommendations

General

- apply 2nd order stress update by setting \textbf{OSU=1} (*CONTROL\_ACCURACY)
- try to model displacement driven simulation instead of force driven simulation
- try to use IGS=1 (not default) on *CONTROL\_IMPLICIT\_GENERAL in case of convergence problems
- set \textbf{DNORM=1} on *CONTROL\_IMPLICIT\_SOLUTION, displacement tolerance can often be increased in that case, e.g. DCTOL=0.005
- try ABSTOL=1.e-20 on *CONTROL\_SOLUTION to improve accuracy
- Sometimes Full Newton (ILIMIT=1) improves convergence
- often dynamic solution more robust than static solution  
  → if static implicit fails to converge, try dynamic implicit
- try to avoid discontinuities, e.g. in material curves, geometry, ...
- use new accuracy option \textbf{IACC=1} on*CONTROL\_ACCURACY (R9)
## Guidelines

### *CONTROL_ACCURACY

<table>
<thead>
<tr>
<th>Card 1</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
</tr>
</thead>
<tbody>
<tr>
<td>Variable</td>
<td>OSU</td>
<td>INN</td>
<td>PIDOSU</td>
<td>IACC</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Type</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Default</td>
<td>1</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Implicit accuracy option IACC=1**

- Higher accuracy in selected material models
  - Fully iterative plasticity, tightened tolerances
- Strong objectivity and consistency in selected tied contacts
  - Physical (only ties to degrees of freedoms that are "real")
  - Finite rotation
- Strong objectivity and increased accuracy in selected elements
  - Finite rotation support for hypoelasticity

**In line with the general philosophy**

"Increased accuracy implies better convergence"
Guidelines

“Use new accuracy option IACC=1 on *CONTROL_ACCURACY”

Example: Plastic deformation of metal part

*MAT_024 with LCSS, DNORM=1, ENDTIM=0.014, DTMAX=0.001

New implicit accuracy flag for *MAT_024, *MAT_123, tied contacts, shell elements, … starting from release R9, see draft version of Keyword Manual.

![Diagram showing stress-strain curve]

- IACC=0: brief overshoot
- IACC=1: exactly on curve
- LCSS curve

(smaller steps would also help, or other material models)
“Set DNORM=1 on *CONTROL_IMPLICIT_SOLUTION”

Example: Compression of a foam block (*MAT_FU_CHANG_FOAM)

ENDTIM=20.0, DTMAX=1.0, DCTOL=0.005, ELFORM=1, IHQ=6, QM=1.0
Guidelines

Recommendations

Output / “Debugging“

- activate print flags (LPRINT/NLPRINT) to get more information
- check output in d3hspt / messag files
- in general, if problems occur when running an implicit model, then try to check the model using *CONTROL_IMPLICIT_EIGENVALUE
- Set MINFO=1 on *CONTROL_OUTPUT to get more informations about the mortar contact: penetrations, release, ...
- in case of convergence problems, dump iteration states via "<ctrl-c> iter" (residual forces in d3iter via RESPLT=1 on *DATABASE_EXTENT_BINARY)
Output of non-converged steps

With \textbf{D3ITCTL=1}, search directions for the nonlinear implicit solution are written to the \texttt{d3iter} database. If used together with \textbf{RESPLT=1} on \*DATABASE\_EXTENT\_BINARY, residual values can be fringed (Version R7):
Guidelines

Recommendations

For “typical” nonlinear analysis, start with the following keyword settings:

*CONTROL_ACCURACY
$  osu     inn
  1  4

*CONTROL_IMPLICIT_GENERAL
$  imflag   dt0   imform   nsbs   igs
  1   ...  (1)

*CONTROL_IMPLICIT_SOLUTION
$  nsolvr   ilimit  maxref  dctol  ectol  rctol  lstol  abstol
  12   6
$  dnorm   diverg  istif   nlprint  nlnorm  d3itctl
  1   1   (4)  (1)
$

$  lsmtd
  (5)

*CONTROL_IMPLICIT_AUTO
$  iauto   iteopt   itewin   dtmin   dtmax
  1   30   10  (term/20)

*CONTROL_IMPLICIT_DYNAMICS
$  imass
  (1)
Guidelines and Examples

New package on www.dynasupport.com:


… provided by Dynamore Nordic.

In this document, some basic control card settings suitable for different implicit analysis types are presented. The analysis types are also accompanied by some basic examples. The purpose is to reduce the effort of getting started with implicit analysis in LS-DYNA.

The package also includes a document about Implicit Mortar Contact Problems.
Examples

Rubber bearing

- Rubber confined by steel parts (diameter: 63mm, height: 40 mm)
- 1st phase: outer ring flanging
- 2nd phase: core shift by 2mm
- *MAT_027 for rubber ($\nu = 0.495$)
- Hexahedral elements (half model)

[LS-DYNA Version R9 beta MPP, double precision]
Examples

Rubber bearing: 1st run

*CONTROL_TERMINATION
$ endtim
  2.0

*CONTROL_IMPLICIT_GENERAL
$ imflag dt0
  1 0.05

*SECTION_SOLID
$ secid elform
  1 1

*HOURLGA$
$ hgid ihq qm
  1 6 1.0

*CONTACT_AUTOMATIC_SINGLE_SURFACE
$ ssid msid sstyp mstyp
  1 3
$ fs
  0.4

Nice convergence, but contact does not work at all!
Examples

Rubber bearing: 2nd run

<table>
<thead>
<tr>
<th>*CONTACT_SURFACE_TO_SURFACE</th>
</tr>
</thead>
<tbody>
<tr>
<td>$ ssid msid sstyp mstyp</td>
</tr>
<tr>
<td>2  1  0  0</td>
</tr>
<tr>
<td>$ fs</td>
</tr>
<tr>
<td>0.4</td>
</tr>
</tbody>
</table>

- Old contact with segment sets
- Maybe better suited for solid contact with nearly incompressible material

Contact works better now, but solver fails to find equilibrium at t=0.9 (near the end of flanging phase)

*** Warning 60124 (IMP+124)
6 negative eigenvalues detected
Examples

Rubber bearing: 3rd run

*CONTROL_IMPLICIT_SOLUTION
$  nsolvr  ilimit
    12    6
$  dnorm  nlprint
    1      1
$  lsmtd
    4

*CONTROL_IMPLICIT_AUTO
$  iauto  iteopt  itewin  dtmin  dtmax
    1      30     10  0.0001  -1234

*DEFINE_CURVE
    1234
    0.0        0.05
    1.0        0.05
    2.0        0.05

*CONTACT_AUTOMATIC_SINGLE_SURFACE_MORTAR
$  ssid  msid  sstyp  mstyp
    2      1      0      0
$  fs
    0.4

- Use all recommended implicit settings
- \( \text{DNORM} = 1 \)
- Automatic time stepping
- Mortar contact

Contact works correctly, good convergence, even manages large element distortions
Examples

Rubber bearing: 3rd run

Kink in originally curved surface
Examples

Rubber bearing: 4th run

- Make it more difficult: increase Poisson’s ratio from 0.495 to 0.499

*MAT_MOONEY-RIVLIN_RUBBER

\[
\begin{array}{cccc}
\text{mid} & \text{ro} & \text{pr} & \\
1 & 1.85E-9 & 0.499 & 0.31 & 0.031
\end{array}
\]

Convergence troubles at t=0.75:

```
... Iteration: 8  *|du|/|u| = 4.1805309E-01  *Ei/E0 = 1.6741033E-03
ITERATION LIMIT reached, automatically REFORMING stiffness matrix...
   *** Warning 60124 (IMP+124)    74 negative eigenvalues detected
Iteration:  9  *|du|/|u| = 1.0000000E+00  *Ei/E0 = 1.4155968E-03
Iteration: 10  *|du|/|u| = 1.0000000E+00  *Ei/E0 = 5.9733603E-04
Negative initial energy from quasi-Newton step,
automatically REFORMING stiffness matrix...
   *** Warning 60124 (IMP+124)    49 negative eigenvalues detected
Iteration: 11  *|du|/|u| = 3.7395361E-01  *Ei/E0 = 5.4974565E-04
Iteration: 12  *|du|/|u| = 1.0000000E+00  *Ei/E0 = 5.7415020E-04
...```

That situation improves by changing
LSMTD from 4 (default) to 6 (most robust)

d3iter: deformations and residual forces
Examples

T-joint component

*CONTACT_AUTOMATIC_SINGLE_SURFACE: overall contact

*MAT_138: adhesive bond with failure

*MAT_024: wooden blocks

*PART_INERTIA: $v_0 = 5 \text{ m/s}$

*MAT_024: DP 800

5 mm mesh for steel parts

*CONSTRAINED_RIGID_BODY: lower sheet and wooden block

[LS-DYNA Version R7.1.1 MPP, single and double precision]
Examples

Dynamic explicit
- Process time = 5 ms
- ~10,000 time steps
- 52 cohesive elements fail
- Low-frequency vibration and high-frequency response (wave propagation)

velocity [0 - 10 m/s]
Now, we want to do a static analysis of that process:

1. Start with explicit using a larger time period ("slow" loading)

2. Add implicit cards needed for dynamic implicit analysis ("slow" loading)

3. Remove dynamics and perform pure static analysis
Examples

Static (??) explicit

- Process time = 5 / 50 ms
- ~ 10,000 / 100,000 time steps
- No initial velocity, but prescr. motion
- 52 cohesive elements fail
- Response still dynamic
- Damping… ??

![Graph showing force in kN vs. displacement in mm](image)

velocity [0 - 3 m/s]
Examples

Dynamic implicit

- Process time = 50 ms ("slow")
- Compare to "slow" explicit run

velocity [0 - 3 m/s]
Examples

**Static implicit**
- Remove *CONTROL_IMPLICIT_DYNAMICS*
- No initial velocity, but prescr. motion
- “time“ not physical anymore
- *Real* static response
- statically defined !?!

![Graph showing force vs. displacement for explicit and implicit methods.](image)

*Time = 4.9292*

-force in kN
-displacement in mm

*no velocities!*
Examples

Implicit contact

- Contact is very important issue (especially) in implicit analysis
- User should know about IGAP options (“sticky contact“) and mortar contact (continuous tangent)
- Dynamic implicit shown here

![Graph showing force vs. displacement](image)

**Force in kN**

**Displacement in mm**

** Explicit (“slow“)**

- IGAP on MORTAR
- too early with IGAP

**Implicit**

- IGAP on

**MORTAR**
Examples

Static implicit with Mortar contact

- More realistic results with Mortar contact
- 5 different phases can be observed: no contact (i), tipping (ii), elastic bending (iii), adhesive softening (iv), and glue failure (v)
Examples

Static implicit with Mortar contact

- Convergence becomes more difficult
- Reason(s) for difficulties can be detected with special “iteration plot database” d3iter
- Evolution of out-of-balance forces during iteration process shows critical areas

Troubles from contact and damage evolution in cohesive material
Examples

Ideas for improvement

- Perhaps Full Newton better suited for this problem (ILIMIT=1)
- Modify other implicit settings (timestep size, tolerances, …) or contact parameters (IGAP, …)
- But maybe better to improve the model itself:
- Replacement for cohesive material (MAT_186 with smooth curve?)
- Mesh refinement in critical areas?
- Dynamic implicit – very slow
- …
Summary

- Explicit analysis runs into its limits for long duration processes or even real static load cases.

- Therefore, implicit analysis is often preferrable. Actually, computation time can be decreased in many cases.

- But: more demanding to get a solution, especially if large deformations, contact, and nonlinear material behavior is involved.

- Users must be aware of crucial differences between explicit (e.g. time step size) and implicit (e.g. “smooth“ model)

- Often greater effort is needed to obtain a functional model in implicit, but also the feeling of success is greater in the end