Z-mat Material model library

- Many pre-defined models
- User-extensible
- Object-oriented

After years of research, a large base of constitutive models for plasticity and viscoplasticity is now available with robust integration methods and advanced coefficient identification procedures. A joint effort was recently undertaken between **Centre des Matériaux Mines PARISTECH and NW Numerics, Inc, a Coventry, RI company to make these developments commercially available for use within ABAQUS.**

Z-mat includes several pieces of software which constitute an efficient set of tools for advanced material-oriented FE analysis with ABAQUS.

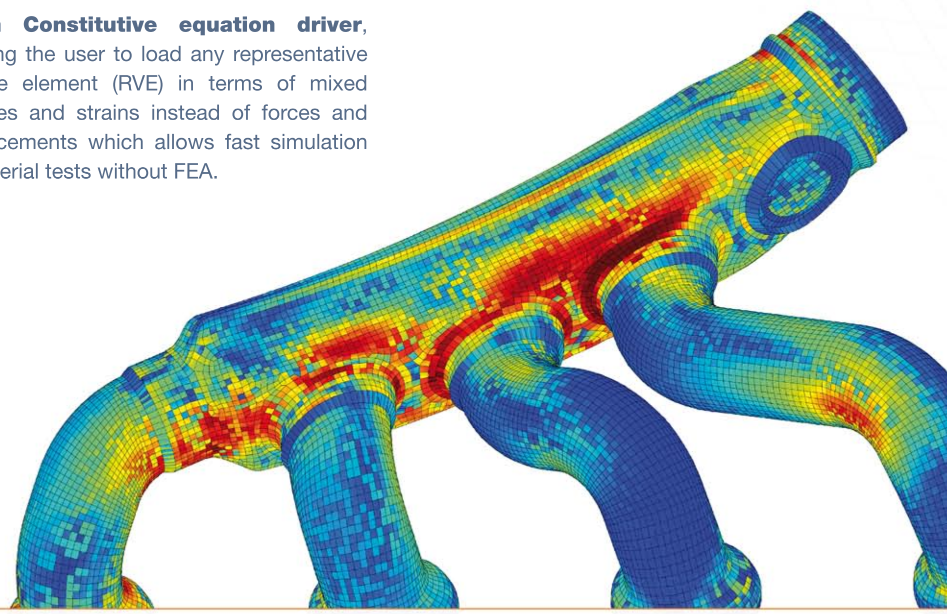
A dynamic library which extends the material modeling capabilities of ABAQUS and provides a more flexible, object-oriented interface for developing user models. There are a large number of proven constitutive equations built into the library, including sophisticated models for plasticity, viscoplasticity, coupled plasticity-viscoplasticity, crystal plasticity, creep, brittle and ductile damage. The nature of the library also allows all these models to be simultaneously available from a centralized location, so the user is not obliged to copy a large umat.f file to each calculation directory. Due to the modular design, the user can also build new models by himself from the input file, simply by combining several types of yield functions, flow rules and hardening rules. For example, plasticity-viscoplasticity does not exist as a predefined model; it is instead assembled on-the-fly in the input file by the user to combine simultaneously time-independent and viscoplastic strain components.

ZebFront High-level material programming language. More advanced users will be interested in programming additions to the library, which is particularly simple using the ZebFront pre-processor language. This language provides constitutive equation oriented data management commands on top of C++, gives simplified access to the integration methods (Runge-Kutta and implicit modified midpoint method), and interfaces with the many utility classes used throughout the library, which can be used as building bricks for fast prototyping of new models. We believe that the material library with ZebFront is the easiest way to implement User Material applications, in very high-level, object oriented programming. The C++ foundation for the library also directly provides that user extensions are seamlessly integrated, appearing as if they were in fact part of the original library. Again, many user extensions can exist simultaneously, simplifying version management, and facilitating end use.

Z-sim Constitutive equation driver, allowing the user to load any representative volume element (RVE) in terms of mixed stresses and strains instead of forces and displacements which allows fast simulation of material tests without FEA.

Note that the material model code is shared 100% between the simulation and FEA. Functionality is also available to plot yield or damage surfaces (actually any potential) in stress space, at different points in the loading history.

Z-opt Optimization including several classical methods such as SQP, simplex or Levenberg-Marquardt, and also genetic algorithms. In connection with Z-sim this is a powerful tool for the identification of material parameters, while keeping exactly the same user interface as Z-mat and Z-sim.



Courtesy of TENNECO

Abaqus input file

```
*********** ABAQUS INPUT FILE ***********
* NODE, NSET=all
1,0.,0.
...

****** Z-mat material declaration *******
*SOLID SECTION,ELSET=ALL,MATERIAL=steel
*MATERIAL,NAME=steel
*DEPVAR
13
*USER MATERIAL,CONSTANTS=1
0.0
**** End Z-mat material declaration *****
```

- The 6 lines below the Z-mat material declaration comment are the only commands to add to the input file to run an Abaqus analysis with a Z-mat material behavior.
- The only thing to change in this block of commands, depending on the particular analysis, is the number of SDV variables needed by the model (command DEPVAR) and the material name **steel** which refers to the name of a Z-mat material file (example included hereafter).

Z-mat input file

```
% ======== File steel =========
***material
  *integration theta_method_a 1.0 1.e-12 1500
***behavior gen_evp
 **elasticity
   young function
   200000.-100.*temperature;
   poisson 0.3
 **thermal_strain
   alpha          temperature
   0.12500E-04    0.0
   0.14500E-04    500.00
   0.14500E-04    1000.00
 **potential gen_evp ev
  *flow norton
   n 4.
   K              temperature
   200.           20.
   300.           650.
   250.           950.
  *isotropic  nonlinear
   R0             temperature
   400.           0.
   300.           1000.
   Q              -200.
   b              4.
  *kinematic  nonlinear
   D function 800.+2.5*(temperature-475.)^2.;
   C 75000.
  *kinematic  linear
   C 1050.
***return
```

- This example illustrates the definition of a typical viscoplastic model for thermo-mechanical analysis.
- Note that all material coefficients can depend upon internal variables or external parameters (eg temperature in present case).
- The dependence on temperature of the coefficients can be specified in a table or using mathematical functions.
- This particular model assembly includes nonlinear isotropic hardening and two kinematic hardenings, both linear and nonlinear (as many kinematic hardenings as needed can be implemented).
- Very complex models (including multi-potential models) can thus be defined by simply adding material bricks directly into the material input file.
- Input is completely free format, and syntax is 100% compatible with the identification tools (Z-sim and Z-opt).