
# Implementation of advanced numerical solvers in FLAC3D thermal and fluid implicit formulation

Andrey V. Pyatigorets, David B. Russell

# FLAC3D thermal and fluid logic

- *FLAC3D* thermal / fluid logic adopt energy-balance / fluid mass-balance equations formulated at each gridpoint:

$$\frac{dT_n}{dt} = -\frac{1}{\sum m_{th}^n} Q_{th}^n \qquad \frac{dP_n}{dt} = -\frac{1}{\sum m_{fl}^n} Q_{fl}^n$$

- *FLAC3D* thermal and fluid modules have two separate solvers: explicit and implicit
- Explicit solver uses forward difference scheme: $T_{new} = T_{old} + \Delta t \cdot f(T_{old})$

**Timestep $\Delta t$ must be small (often $10^{-4} \leftrightarrow 10^{-6}$ sec)!**

- Current implicit solver uses:
  - Central difference (Crank-Nicolson) scheme to obtain a system of linear equations for each gridpoint: $A_{nj}\Delta T_{j\langle t\rangle} = b_{n\langle t\rangle}$
  - Jacobi method (iterative) to solve it

**Jacobi solver easily diverges (fails) when $\Delta t_{implicit} \gtrsim [10 \leftrightarrow 100]\, \Delta t_{explicit}$**

Need to find an approach which will allow using much larger timestep in the implicit scheme!

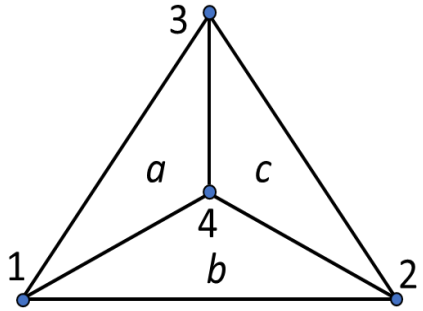
# New approach – use global matrix

- Global matrix is assembled for the whole model based on previously calculated local matrices:

$$\begin{bmatrix} Q_1^a \\ Q_4^a \\ Q_3^a \end{bmatrix} = \begin{bmatrix} c_{11}^a & c_{14}^a & c_{13}^a \\ & c_{44}^a & c_{43}^a \\ & & c_{33}^a \end{bmatrix} \begin{bmatrix} \Delta T_1 \\ \Delta T_4 \\ \Delta T_3 \end{bmatrix}$$

$$\begin{bmatrix} Q_1^b \\ Q_2^b \\ Q_4^b \end{bmatrix} = \begin{bmatrix} c_{11}^b & c_{12}^b & c_{14}^b \\ & c_{22}^b & c_{24}^b \\ & & c_{44}^b \end{bmatrix} \begin{bmatrix} \Delta T_1 \\ \Delta T_2 \\ \Delta T_4 \end{bmatrix}$$

$$\begin{bmatrix} Q_2^c \\ Q_3^c \\ Q_4^c \end{bmatrix} = \begin{bmatrix} c_{22}^c & c_{23}^c & c_{24}^c \\ & c_{33}^c & c_{34}^c \\ & & c_{44}^c \end{bmatrix} \begin{bmatrix} \Delta T_2 \\ \Delta T_3 \\ \Delta T_4 \end{bmatrix}$$




$$\begin{bmatrix} Q_1 \\ Q_2 \\ Q_3 \\ Q_4 \end{bmatrix} = \begin{bmatrix} Q_1^a + Q_1^b \\ Q_2^b + Q_2^c \\ Q_3^a + Q_3^c \\ Q_4^a + Q_4^b + Q_4^c \end{bmatrix} =$$

$$= \begin{bmatrix} c_{11}^a + c_{11}^b & c_{12}^b & c_{13}^a & c_{14}^a + c_{14}^b \\ & c_{22}^b + c_{22}^c & c_{23}^c & c_{24}^b + c_{24}^c \\ & & c_{33}^a + c_{33}^c & c_{43}^a + c_{34}^c \\ & & & c_{44}^a + c_{44}^b + c_{44}^c \end{bmatrix} \begin{bmatrix} \Delta T_1 \\ \Delta T_2 \\ \Delta T_3 \\ \Delta T_4 \end{bmatrix}$$

- Real global matrix is very sparse, symmetric and positive definite!

# New solvers

- Two new solvers from INTEL MKL are adopted:
  - **Preconditioned Conjugate Gradient (PCG) Solver** – iterative solver for very large sparse systems (e.g. millions of equations). Uses simple Jacobi (diagonal) preconditioner.
  - **Direct Sparse Solver (DSS)** – direct solver for smaller sparse systems (equivalent to matrix inversion; LU / Chlolesky).

- New solvers operate on ***global matrix*** representing the whole model. The matrix must be assembled in compressed sparse format (CSR3) - done once when user calls new solvers.

- Global matrix must be symmetric and positive definite. To check this and to find matrix condition number $\kappa$, extremal eigenvalues are calculated (Krylov-Schur method). Limit on number if iteration in PCG:

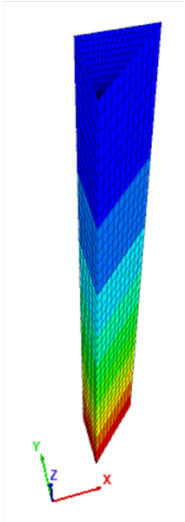
$$\kappa = \frac{\lambda_{max}}{\lambda_{min}}, \qquad n \leq \frac{1}{2}\kappa \log\frac{2}{\varepsilon}$$

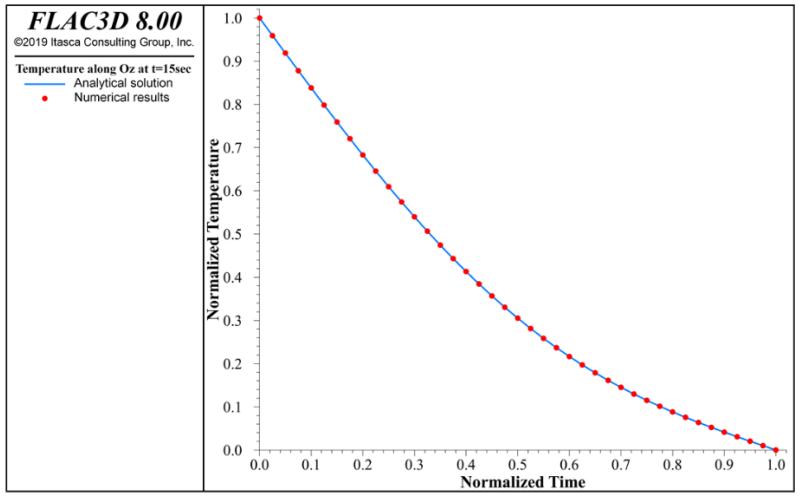
- **Solvers are unconditionally stable! Solution can always be found up to the specified accuracy (providing the time step is below the characteristic time of the model)!**

# Performance

- Simple diffusion problem: $T_{bot}$ = 100C, $T_{top}$ = 0.

Results of temperature distribution after 15sec:






| Scheme / Solver | Explicit | Implicit / Jacobi | Implicit / precond. CG | Implicit / direct |
|---|---|---|---|---|
| Timestep | 1.28856e-5 | 1e-4 | 0.5 | 0.5 |
| Number of steps | 1164094 | 150000 | 30 | 30 |
| Error | 2e-5 | 5e-4 | 9e-5 | 9e-5 |
| Relative speed-up (runtime) | 1X (11m02s) | 2.1X (5m13s) | 441X (1.5sec) | 662X (1sec) |
| Time/Step | 5.68e-4 | 2.09e-3 (3.7x) | 5e-2 (88x) | 3.33e-2 (58x) |

Overall speed-up over the explicit logic is

$10^2$ - $10^6$

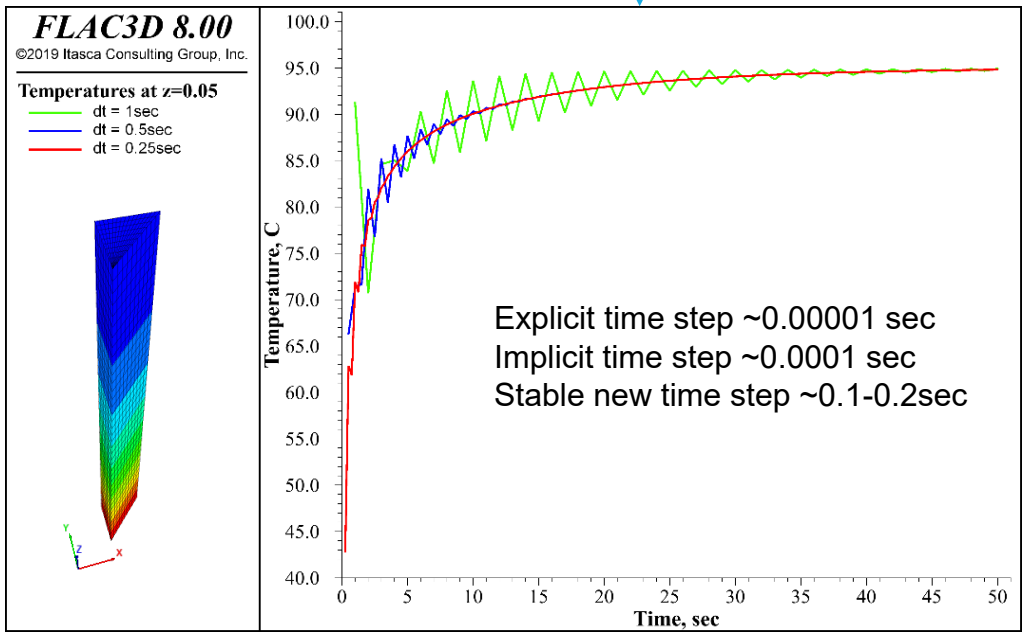
(for thermal diffusion problems involving large time step and large calculation time)

New solvers have been tested on:

- heat conduction,
- thermo-mechanical coupling,
- constant and transient boundary conditions,
- heat sources/sinks,
- convective boundaries.

# Challenges / limitations

- Currently there is an issue within thermal/fluid Attach logic which does not properly account for presence of interfaces within same zone (when multiple attach conditions have to be resolved).
- Possible oscillations (from C-N scheme) for large time steps (so small number of cycles)!
- The global matrix has to be re-assembled every time the model geometry or physical properties of gp/zones change. Thus, new solvers may be inefficient for:
  - Problems in large strain mode
  - Problems with moving boundaries
  - Problems requiring small time step

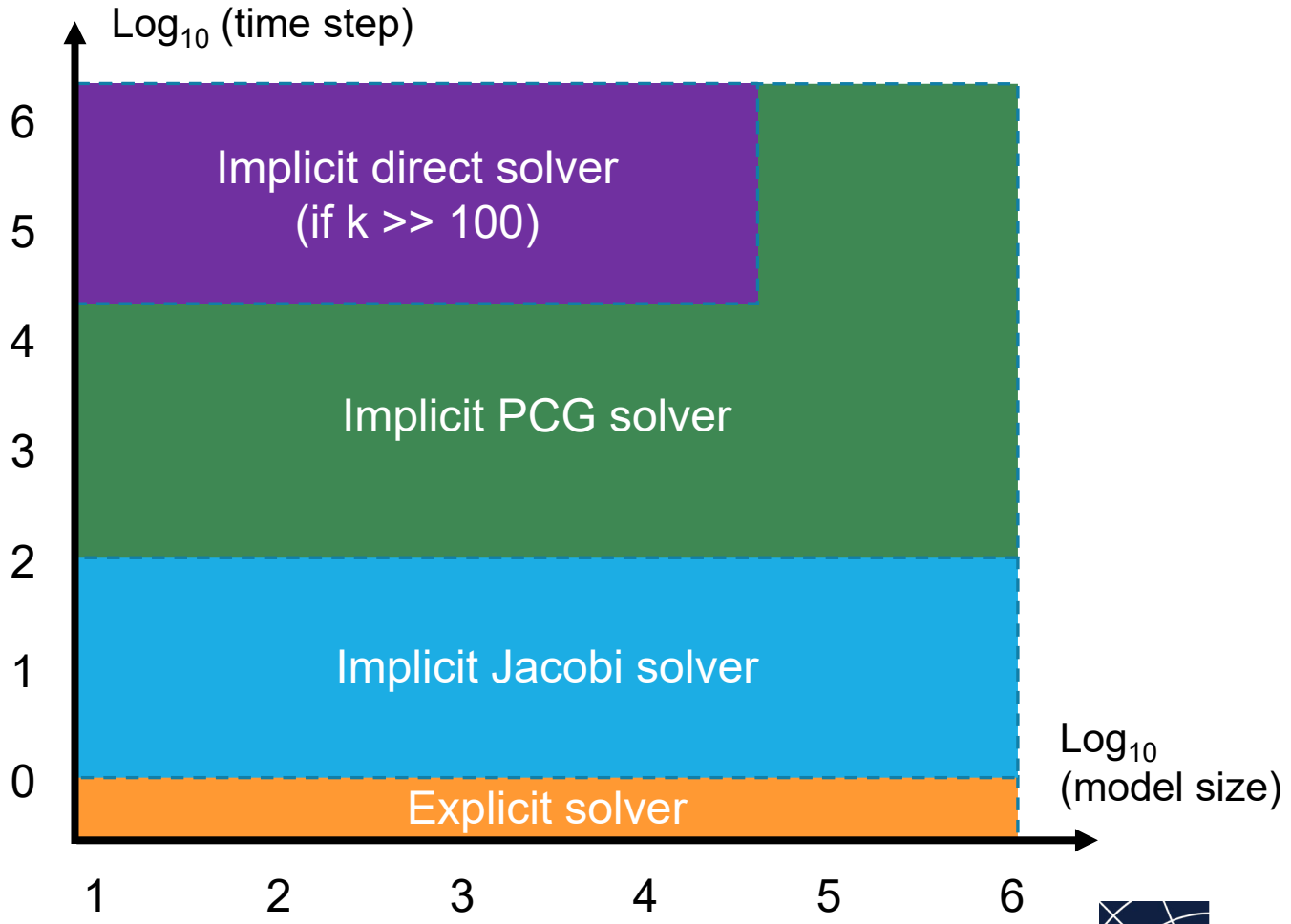


# Applicability

- Choice of proper solver is done by user:
  - Explicit method
  - Implicit Jacobi solver
  - Implicit PCG solver
  - Implicit direct solver

- It depends on:
  - Physical parameters (i.e. time-dependent)
  - Boundary conditions (i.e. time-dependent)
  - Timestep
  - Model size

Generic solvers applicability for best performance



# Command syntax

```
model configure thermal
model thermal active on
```

- **zone** thermal implicit on
- **zone** thermal implicit on solver-jacobi
- **zone** thermal implicit on solver-pcg [*preserve*]
- **zone** thermal implicit on solver-direct [*preserve*]

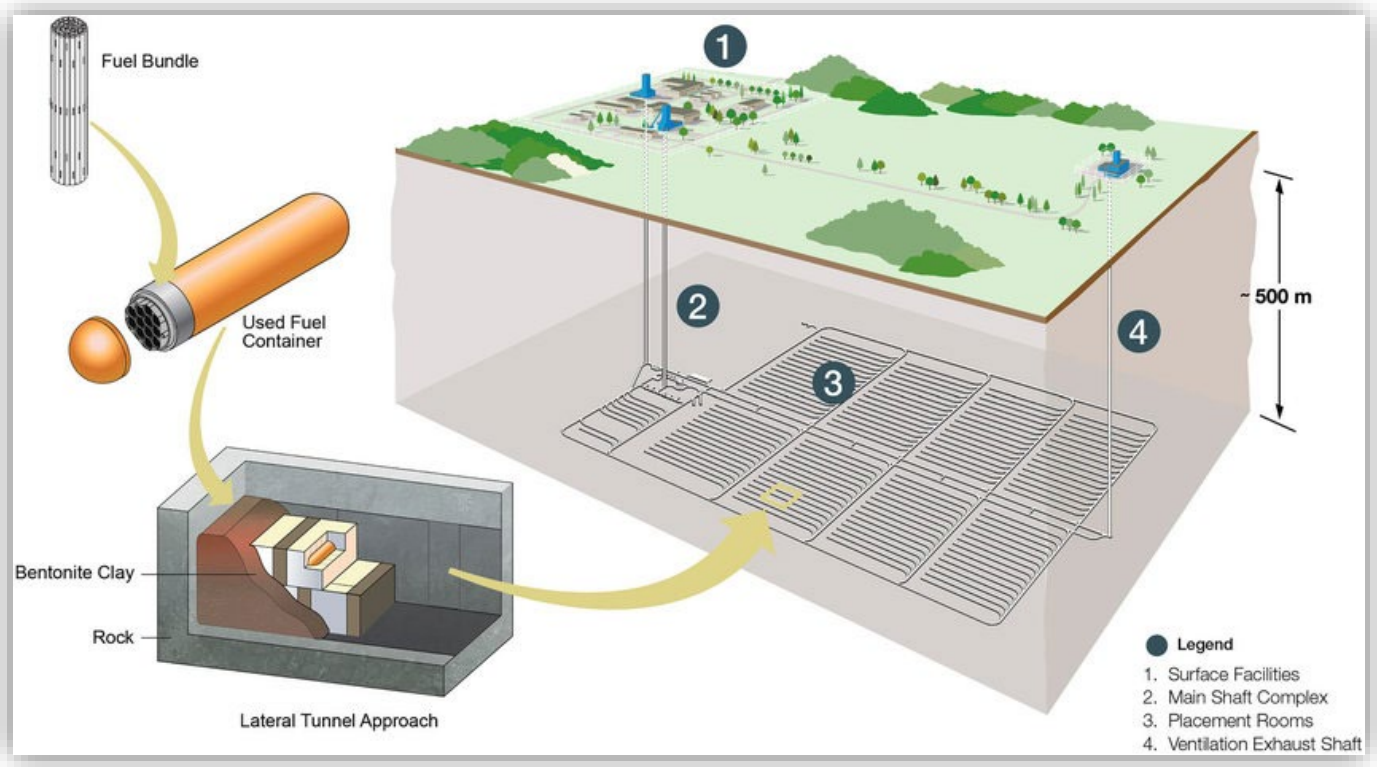
```
model thermal timestep fix …
model cycle … / solve time …
```

FLAC3D 7: calls Jacobi solver only

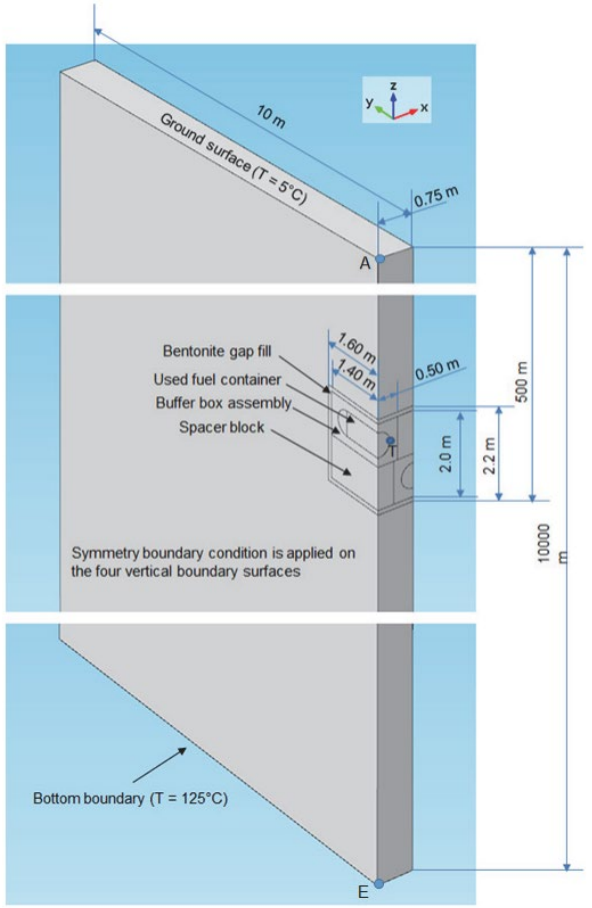
FLAC3D 8: calls Jacobi solver; if diverges – automatically switches to PCG solver

# NWMO project

## Long-Term Stability Analysis of APM Conceptual Repository Design in Sedimentary and Crystalline Rock Settings (2015) (NWMO-TR-2015-27)




Hall, Keech (2017). Corr. Eng., Science and Techn.




**Ruiping Guo**
NWMO-TR-2016-03

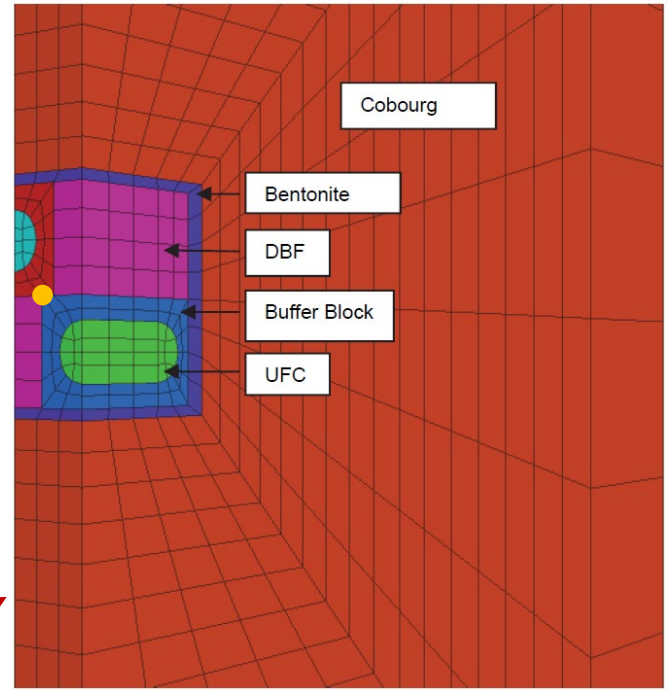
# NWMO project

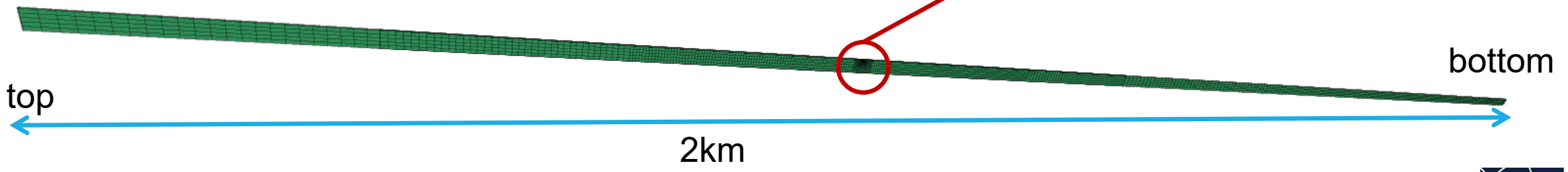
**Long-Term Stability Analysis of APM Conceptual Repository Design in Sedimentary and Crystalline Rock Settings (2015) (NWMO-TR-2015-27)**

Temperature simulation in a nuclear waste repository over 1 million years

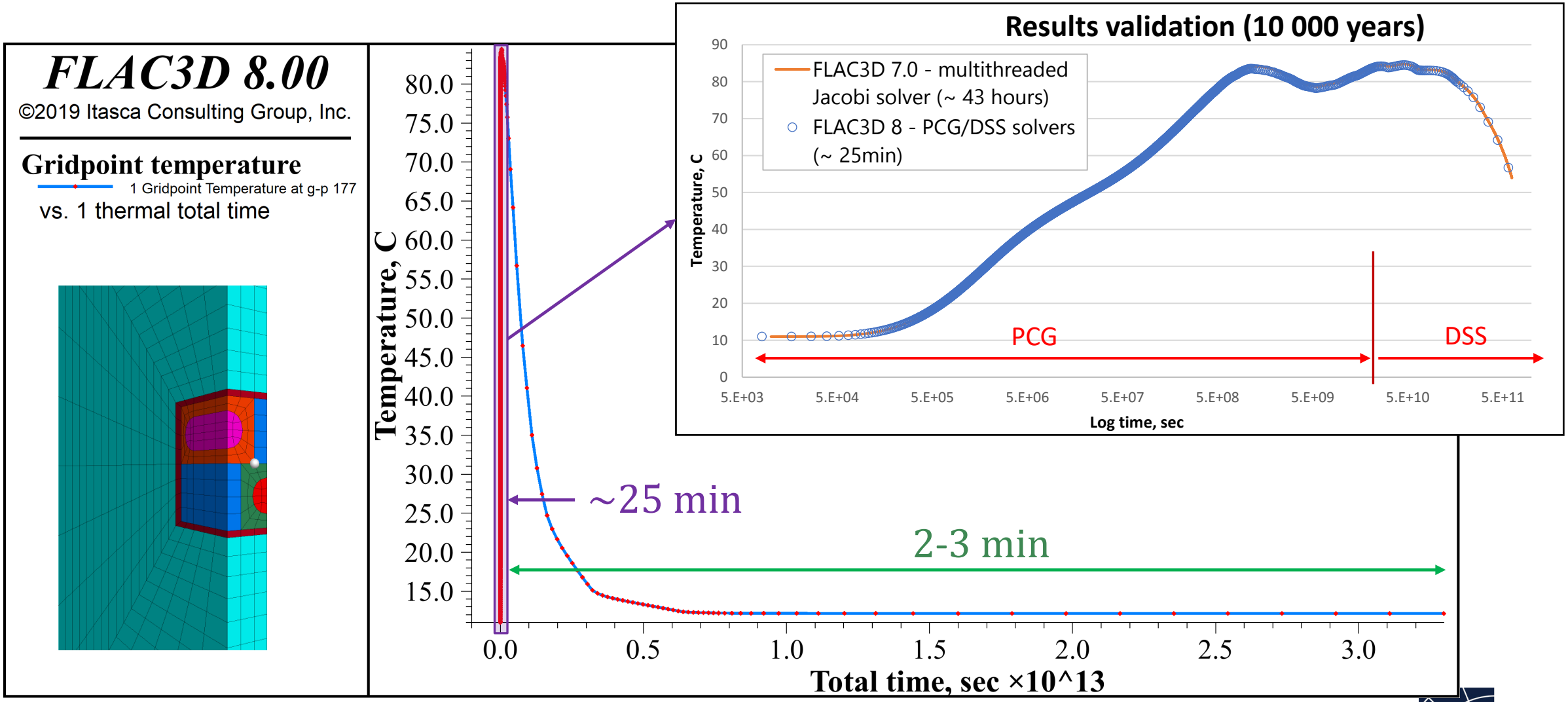
- FLAC3D 5 runtime is ~ **1 month** with single-thread thermal logic
- FLAC3D 7 runtime ~ **2-4 weeks** with multi-threaded thermal logic
- FLAC3D 8 runtime is ~ **25-30 minutes** using combination of new solvers

- The largest stable timestep for Jacobi solver (old technique) ~ 30 days
- The largest timestep used in new solvers ~ 1200 years (limited artificially, solvers allow larger)



# NWMO results



# NWMO results