Solvers for Computational Mechanics

Arnaud Delaplace - David Ryckelynck

Outline

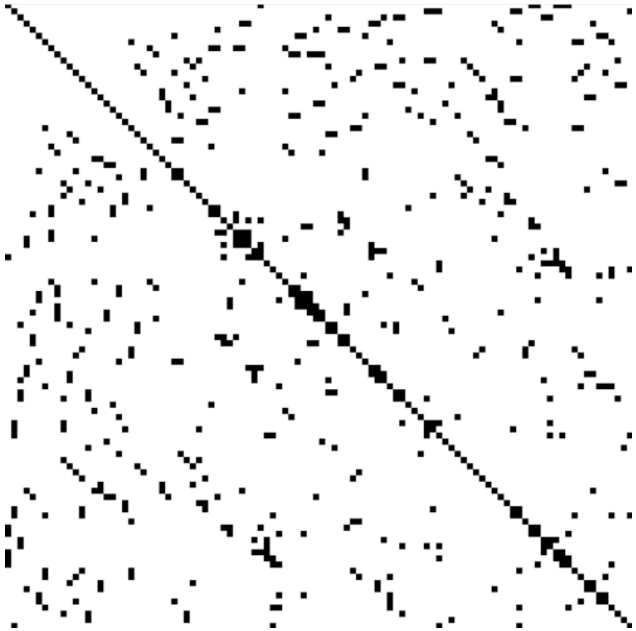
Discretized form of the static equations:

$$\mathbf{K}(\mathbf{u}_t)\mathbf{u}_t = \mathbf{r}_t \tag{1}$$

Solved by using either a *direct* method or an *iterative* one.

Matrix storage

In computational mechanics, $\mathbf{K}$ is usually a sparse-symmetric matrix. One needs to store only the non zero-elements (sparse storage, skyline storage...).



Direct solvers

- Gauss-Jordan elimination,
- LU decomposition,
- Inversion by partitionning,
- Cholesky decomposition...

Very efficient, robust, free libraries are available!!!

$\rightarrow$ LAPACK (Linear Algebra PACKage) with BLAS (Basic Linear Algebra Subprograms), ATLAS (Automatically Tuned Linear Algebra System), SuiteSparse...

Iterative solvers

- Mainly based on the conjugate gradient method family, and a *preconditioner*
- Efficient for large system with sparse operator

Algorithm

1. Initialisation: $\mathbf{u}_0^k$ arbitrary, $\mathbf{g}_0^k = \mathbf{K}\mathbf{u}_0^k - \mathbf{f}^k$, $\mathbf{z}_0^k = \mathbf{P}^{-1}\mathbf{g}_0^k$
2. $i \Leftarrow 1$
3. **REPEAT**
4. $\boldsymbol{\omega}_i^k = \mathbf{z}_i^k + \gamma_{i-1}^k \boldsymbol{\omega}_{i-1}^k$ $\qquad \gamma_{i-1}^k = -\frac{(\mathbf{z}_i^k, \mathbf{K}\boldsymbol{\omega}_{i-1}^k)}{(\mathbf{K}\boldsymbol{\omega}_{i-1}^k, \boldsymbol{\omega}_{i-1}^k)}$
5. $\mathbf{u}_{i+1}^k = \mathbf{u}_i^k + \alpha_i^k \boldsymbol{\omega}_i^k$
6. $\mathbf{g}_{i+1}^k = \mathbf{g}_i^k + \alpha_i^k \mathbf{K}\boldsymbol{\omega}_i^k$ $\qquad \alpha_i^k = -\frac{(\mathbf{g}_i^k, \mathbf{z}_i^k)}{(\mathbf{K}\boldsymbol{\omega}_i^k, \boldsymbol{\omega}_i^k)}$
7. $\mathbf{z}_{i+1}^k = \mathbf{P}^{-1}\mathbf{g}_{i+1}^k$
8. $i \Leftarrow i + 1$
9. **UNTIL** convergence

- For a nonlinear problem, the solution is obtained with an iterative process, where a succession of linear systems are solved.
- Examples: Newton-Raphson solver, radial return algorithm (mainly dedicated to plasticity problem)

Newton-Raphson solver

We search for:

$$F_i(x_1, x_2, ..., x_N) = 0 \text{ for } i = 1, 2, ..., N$$

Taylor series expansion:

$$F_i(\mathbf{x} + \delta\mathbf{x}) = F_i(\mathbf{x}) + \sum_{j=1}^{N} \frac{\partial F_i}{\partial x_j} \delta x_j + \mathcal{O}(\delta\mathbf{x}^2)$$

Using the Jacobian matrix $J_{ij} = \partial F_i / \partial x_j$, the matrix notation leads to:

$$\mathbf{F}(\mathbf{x} + \delta\mathbf{x}) = \mathbf{F}(\mathbf{x}) + \mathbf{J}\delta\mathbf{x} + \mathcal{O}(\delta\mathbf{x}^2)$$

If the term of order $\delta\mathbf{x}^2$ are neglected, one has to solve the system:

$$\mathbf{J}\delta\mathbf{x} = -\mathbf{F}$$

The corrections are added to the solution vector until convergence:

$$\mathbf{x}_{new} = \mathbf{x}_{old} + \delta\mathbf{x}$$

Radial return algorithm

For a return mapping algorithm, the first (trial) solution is computed assuming an elastic step. If the yield criterion is violated, then the solution is projected back to the plastic yield surface to give the updated stress.

Dynamic problems

$\longrightarrow$ Inertia terms must be included in the equations of equilibrium

- Wave propagation problems (response governed by high frequency modes)
  ex.: shock response, impact loading...
- Inertia problems (response governed by a small number of low frequency modes)
  ex.: seismic response...

Governing equations:

$$\mathbf{f}^i(t) + \mathbf{f}^D(t) + \mathbf{f}^{int}(t) = \mathbf{f}^{ext}(t)$$

- $\mathbf{f}^i(t)$: inertia forces
- $\mathbf{f}^D(t)$: damping forces
- $\mathbf{f}^{int}(t)$: internal forces
- $\mathbf{f}^{ext}(t)$: external forces

Equations of motion for linear structural dynamics

$$\mathbf{M}\ddot{\mathbf{u}}_t + \mathbf{C}\dot{\mathbf{u}}_t + \mathbf{K}\mathbf{u}_t = \mathbf{r}_t \tag{2}$$

- $\mathbf{M}$: discrete mass matrix
- $\mathbf{C}$: viscous damping matrix
- $\mathbf{K}$: linear stiffness matrix
- $\mathbf{r}_t$: vector of external discrete forces
- $\ddot{\mathbf{u}}_t, \dot{\mathbf{u}}_t, \mathbf{u}_t$: nodal acceleration, velocity and displacement vectors

We search the time-history response $\mathbf{u}_t$ of the structure.

Initial conditions

$$\mathbf{u}(0) = \mathbf{u}_0$$

$$\dot{\mathbf{u}}(0) = \dot{\mathbf{u}}_0$$

The initial acceleration is obtained using the equations of motion:

$$\ddot{\mathbf{u}}_0 = \mathbf{M}^{-1} \left(\mathbf{r}(0) - \mathbf{C}\dot{\mathbf{u}}_0 - \mathbf{K}\mathbf{u}_0 \right)$$

Equations of motion for nonlinear structural dynamics

$$\mathbf{M}\ddot{\mathbf{u}}_t + \mathbf{C}\dot{\mathbf{u}}_t + \mathbf{n}(\mathbf{u}_t) = \mathbf{r}_t \tag{3}$$

- $\mathbf{M}$: discrete mass matrix
- $\mathbf{C}$: viscous damping matrix
- $\mathbf{n}$: vector of nonlinear internal forces
- $\mathbf{r}_t$: vector of external discrete forces
- $\ddot{\mathbf{u}}_t, \dot{\mathbf{u}}_t, \mathbf{u}_t$: nodal acceleration, velocity and displacement vectors

Vector of nonlinear internal forces:

$$\mathbf{K}_t(\mathbf{u}) = \frac{\partial \mathbf{n}(\mathbf{u})}{\partial \mathbf{u}}$$

- $\mathbf{K}_t$: tangent stiffness matrix

$$\mathbf{n}(\mathbf{u}) = \sum_e \int_V \mathbf{B}^T \boldsymbol{\sigma}(\varepsilon) dV$$

Stability

The solution of a problem can be expressed in a recurrence relation form:

$$\mathbf{u}_{t+\Delta t} = \mathbf{A}\mathbf{u}_t + \ldots$$

with $\mathbf{A}$ the amplification matrix. The stability of the scheme is obtained if the spectral radius of $\mathbf{A}$ verifies:

$$\rho(\mathbf{A}) \leq 1$$

where $\rho(\mathbf{A}) = \max(\lambda_i)$ and λ_i are the eigenvalues of $\mathbf{A}$.

Accuracy

For a resolution over the time range $[t_0, t_f]$, with a time increment Δt corresponding to $n = (t_f - t_0)/\Delta t$, the global error of a scheme is evaluated as:

$$e = \mathbf{u}_{t_f} - \mathbf{u}_n$$

The order of accuracy p of the scheme is defined as:

$$e = \mathcal{O}(\Delta t^p) \text{ when } \Delta t \to 0$$

Explicit schemes

- Solution at time $t + \Delta t$ is obtained from equilibrium equations at time t
- Do not require the factorization of the stiffness matrix
  ($\rightarrow$ and no additional storage for a diagonal mass matrix)
- Efficient for short load durations (impact, explosions...)

But

- Conditionally stable
- Require generally small time steps (inversely proportional to the highest frequency of the discrete system)

- Central different methods, Runge-Kutta methods, Predictor-corrector methods, Taylor series schemes...

$$\dot{\mathbf{u}}_t = \frac{1}{\Delta t}(\mathbf{u}_{t+\Delta t} - \mathbf{u}_{t-\Delta t}) \tag{4}$$

$$\ddot{\mathbf{u}}_t = \frac{1}{\Delta t^2}(\mathbf{u}_{t+\Delta t} - \mathbf{u}_t + \mathbf{u}_{t-\Delta t}) \tag{5}$$

Using equation (3),

$$\left(\tfrac{1}{\Delta t^2}\mathbf{M} + \tfrac{1}{2\Delta t}\mathbf{C}\right)\mathbf{u}_{t+\Delta t} = \mathbf{r}_t - \left(\mathbf{n}(\mathbf{u}_t) - \tfrac{2}{\Delta t^2}\mathbf{M}\mathbf{u}_t\right) - \left(\tfrac{1}{\Delta t^2}\mathbf{M} - \tfrac{1}{2\Delta t}\mathbf{C}\right)\mathbf{u}_{t-\Delta t}$$

(with $\mathbf{n}(\mathbf{u}_t) = \mathbf{K}\mathbf{u}_t$ for linear problem)

Without physical damping ($\mathbf{C} = \mathbf{0}$),

$$\mathbf{u}_{t+\Delta t} = \Delta t^2 \mathbf{M}^{-1}\left(\mathbf{r}_t - \mathbf{n}(\mathbf{u}_t)\right) + 2\mathbf{u}_t - \mathbf{u}_{t-\Delta t}$$

Remark

If damping is included, one prefers the *backward* difference scheme, where equation (4) is replaced by:

$$\dot{\mathbf{u}}_t = \frac{1}{\Delta t}(\mathbf{u}_t - \mathbf{u}_{t-\Delta t})$$

Then,

$$\mathbf{u}_{t+\Delta t} = \mathbf{M}^{-1}\left(\Delta t^2(\mathbf{r}_t - \mathbf{n}(\mathbf{u}_t)) - \Delta t\mathbf{C}(\mathbf{u}_t - \mathbf{u}_{t-\Delta t})\right) + 2\mathbf{u}_t - \mathbf{u}_{t-\Delta t}$$

Algorithm

1. Initial calculations
   - 1.1 Initialize $\mathbf{u}_0$ and $\dot{\mathbf{u}}_0$.
   - 1.2 Form structural mass matrix $\mathbf{M}$ and damping matrix $\mathbf{C}$.
   - 1.3 Select time step such that $\Delta t < \Delta t_{cr}$.
2. For each time step
   - 2.1 Calculate effective load at time t: $\hat{\mathbf{r}}_t = \mathbf{r}_t - \mathbf{C}(\mathbf{u}_t - \mathbf{n}(\mathbf{u}_t)$
   - 2.2 Solve for accelerations at time t: $\mathbf{M}\ddot{\mathbf{u}}_t = \hat{\mathbf{r}}_t$
   - 2.3 Evaluate (for the first time step): $\mathbf{u}_{t-\Delta t} = \mathbf{u}_t - \Delta t \dot{\mathbf{u}}_t + \frac{\Delta t^2}{4}\ddot{\mathbf{u}}_t$
   - 2.4 Evaluate displacement and velocity at time $t + \Delta t$

$$\mathbf{u}_{t+\Delta t} = -\mathbf{u}_{t-\Delta t} + 2\mathbf{u}_t + \Delta t^2 \ddot{\mathbf{u}}_t$$

$$\dot{\mathbf{u}}_{t+\Delta t} = \left[\mathbf{u}_{t+\Delta t} - \mathbf{u}_t\right]/\Delta t$$

   - 2.5 $t \leftarrow t + \Delta t$ and go to 2.1.

Fourth order Runge-Kutta method

$$\mathbf{u}_{t+\Delta t} = \mathbf{u}_t + \Delta t \dot{\mathbf{u}}_t + \frac{\Delta t^2}{6}(\mathbf{a}_0 + \mathbf{a}_1 + \mathbf{a}_2) + \mathcal{O}(t^5)$$

$$\dot{\mathbf{u}}_{t+\Delta t} = \dot{\mathbf{u}}_t + \frac{\Delta t}{6}(\mathbf{a}_0 + 2\mathbf{a}_1 + 2\mathbf{a}_2 + \mathbf{a}_3) + \mathcal{O}(t^5)$$

$$\mathbf{a}_0 = \ddot{\mathbf{u}}(t, \mathbf{u}_t)$$

$$\mathbf{a}_1 = \ddot{\mathbf{u}}(t + \frac{\Delta t}{2}, \mathbf{u}_t + \frac{\Delta t}{2}\dot{\mathbf{u}}_t)$$

$$\mathbf{a}_2 = \ddot{\mathbf{u}}(t + \frac{\Delta t}{2}, \mathbf{u}_t + \frac{\Delta t}{2}\dot{\mathbf{u}}_t + \frac{\Delta t^2}{4}\mathbf{a}_0)$$

$$\mathbf{a}_3 = \ddot{\mathbf{u}}(t + \Delta t, \mathbf{u}_t + \Delta t \dot{\mathbf{u}}_t + \frac{\Delta t^2}{2}\mathbf{a}_1)$$

- Self-starting scheme
- Accurate evaluation of the solution

- Acceleration vector must be evaluated four times per time step
- Requires a small time step
- Generally (computationally) slower than the central difference scheme

Predictor:

$$\mathbf{u}^{p}_{t+\Delta t} = \mathbf{u}_t + \frac{\Delta t}{2}(3\dot{\mathbf{u}}_t - \dot{\mathbf{u}}_{t-\Delta t})$$

Iterate with the corrector until convergence:

$$\mathbf{u}^{m+1}_{t+\Delta t} = \mathbf{u}^{m}_{t+\Delta t} + \frac{\Delta t}{2}(\dot{\mathbf{u}}_{t+\Delta t} + \dot{\mathbf{u}}_t)$$

Implicit schemes

- Solution at time $t + \Delta t$ involves the velocities and the accelerations at time $t + \Delta t$
- (Generally) unconditionally stable
- Efficient for long load durations (seismic loading...)

But

- Require the solution of the stiffness matrix
- Require more computational effort per time step

- Newmark family methods, Wilson-θ methods, HHT method...

Newmark time integration scheme

$$\mathbf{u}_{t+\Delta t} = \mathbf{u}_t + \Delta t \dot{\mathbf{u}}_t + \frac{\Delta t^2}{2}\left[(1-2\beta)\ddot{\mathbf{u}}_t + 2\beta\ddot{\mathbf{u}}_{t+\Delta t}\right]$$

$$\dot{\mathbf{u}}_{t+\Delta t} = \dot{\mathbf{u}}_t + \Delta t\left[(1-\gamma)\ddot{\mathbf{u}}_t + \gamma\ddot{\mathbf{u}}_{t+\Delta t}\right]$$

β and γ are the Newmark parameters which determine the stability and accuracy of the algorithm.

Equations of motion

$$\begin{aligned}\left(\tfrac{1}{\beta(\Delta t)^2}\mathbf{M} + \tfrac{\gamma}{\beta\Delta t}\mathbf{C}\right)\mathbf{u}_{t+\Delta t} + \mathbf{n}(\mathbf{u}_{t+\Delta t}) &= \mathbf{r}_{t+\Delta t} \\ +\mathbf{C}\left(\tfrac{\gamma}{\beta\Delta t}\mathbf{u}_t + (\tfrac{\gamma}{\beta}-1)\dot{\mathbf{u}}_t + \Delta t(\tfrac{\gamma}{2\beta}-1)\ddot{\mathbf{u}}_t\right) & \\ +\mathbf{M}\left(\tfrac{1}{\beta(\Delta t)^2}\mathbf{u}_t + \tfrac{1}{\beta\Delta t}\dot{\mathbf{u}}_t + (\tfrac{1}{2\beta}-1)\ddot{\mathbf{u}}_t\right) & \end{aligned} \quad (6)$$

Newmark family methods

Trapezoidal rule ($\beta = 1/4$ and $\gamma = 1/2$):

$$\mathbf{u}_{t+\Delta t} = \mathbf{u}_t + \Delta t \dot{\mathbf{u}}_t + \frac{\Delta t^2}{4} [\ddot{\mathbf{u}}_t + \ddot{\mathbf{u}}_{t+\Delta t}]$$

$$\dot{\mathbf{u}}_{t+\Delta t} = \dot{\mathbf{u}}_t + \frac{\Delta t}{2} [\ddot{\mathbf{u}}_t + \ddot{\mathbf{u}}_{t+\Delta t}]$$

Central difference scheme ($\beta = 0$ and $\gamma = 1/2$):

$$\mathbf{u}_{t+\Delta t} = \mathbf{u}_t + \Delta t \dot{\mathbf{u}}_t + \frac{\Delta t^2}{4} \ddot{\mathbf{u}}_t$$

$$\dot{\mathbf{u}}_{t+\Delta t} = \dot{\mathbf{u}}_t + \frac{\Delta t}{2} [\ddot{\mathbf{u}}_t + \ddot{\mathbf{u}}_{t+\Delta t}]$$

Linear acceleration method ($\beta = 1/6$ and $\gamma = 1/2$)...

Stability aspects

method	type	β	γ	stability
Trapezoidal rule	implicit	$1/4$	$1/2$	unconditional
Linear acceleration	implicit	$1/6$	$1/2$	$\Omega_{crit} = 2\sqrt{3}$
Fox-Goodwin	implicit	$1/12$	$1/2$	$\Omega_{crit} = \sqrt{6}$
Central difference	explicit	0	$1/2$	$\Omega_{crit} = 2$

Rem.: all of these methods are 2-order of accuracy.

The Newmark family methods are *unconditionally* stable if $\gamma \geq 1/2$ and $\beta \geq (\gamma + 1/2)^2/4$.
The Newmark family methods are *conditionally* stable for $\gamma \geq 1/2$, $\beta < \gamma/2$ and $\omega \Delta t \geq \Omega_{crit}$

Newmark family methods

- Positive damping is introduced if $\gamma > 1/2$ (but the scheme looses its second-roder accuracy).
- Negative damping is introduced if $\gamma < 1/2$ (leading eventually to an unbounded response).

Wilson-θ method

The acceleration is assumed to be linear from time t to time $t+\theta\Delta t$, with $\theta \geq 0$. The method is unconditionnaly stable if $\theta \geq 1.37$. For $0 \leq \tau \leq t+\theta\Delta t$,

$$\ddot{\mathbf{u}}_{t+\tau} = \ddot{\mathbf{u}}_t + \frac{\tau}{\theta\Delta t}\left[\ddot{\mathbf{u}}_{t+\theta\Delta t} - \ddot{\mathbf{u}}_t\right]$$

$$\dot{\mathbf{u}}_{t+\tau} = \dot{\mathbf{u}}_t + \tau\ddot{\mathbf{u}}_t + \frac{\tau^2}{2\theta\Delta t}\left[\ddot{\mathbf{u}}_{t+\theta\Delta t} - \ddot{\mathbf{u}}_t\right]$$

$$\mathbf{u}_{t+\tau} = \mathbf{u}_t + \tau\dot{\mathbf{u}}_t + \frac{\tau^2}{2}\ddot{\mathbf{u}}_t + \frac{\tau^3}{6\theta\Delta t}\left[\ddot{\mathbf{u}}_{t+\theta\Delta t} - \ddot{\mathbf{u}}_t\right]$$

HHT method

Equation of motion

$$\mathbf{M}\ddot{\mathbf{u}}_{t+\Delta t} + (1+\alpha)\mathbf{C}\dot{\mathbf{u}}_{t+\Delta t} - \alpha\mathbf{C}\dot{\mathbf{u}}_t + (1+\alpha)\mathbf{K}\mathbf{u}_{t+\Delta t} - \alpha\mathbf{K}\mathbf{u}_t = (1+\alpha)\mathbf{r}_{t+\Delta t} - \alpha\mathbf{r}_t$$

$\alpha \in [-1/3, 0]$ is a dissipative parameter, $\beta = \frac{1}{4}(1-\alpha)^2$ and $\gamma = \frac{1}{2} - \alpha$.

$\Rightarrow$ Dissipation of the high frequencies

Algorithm (1/2)

1. Initialize $\mathbf{u}_0$, $\dot{\mathbf{u}}_0$ and $\ddot{\mathbf{u}}_0$.
2. Calculate the following constants:

$$a_0 = 1/(\beta \Delta t^2);\ a_1 = \gamma/(\beta \Delta t);\ a_3 = 1/(2\beta) - 1$$

$$a_4 = \gamma/\beta - 1;\ a_5 = \Delta t/2(\gamma/\beta - 2)$$

3. Form $\mathbf{M}$, $\mathbf{C}$ and $\mathbf{K}$.
4. Form the effective stiffness matrix, initially assuming a liner behavior: $\hat{\mathbf{K}} = a_0\mathbf{M} + a_1\mathbf{C} + \mathbf{K}$
5. Form the effective load vector:

$$\hat{\mathbf{r}}_{t+\Delta t} = \mathbf{r}_{t+\Delta t} + \mathbf{M}(a_2\dot{\mathbf{u}}_t + a_3\ddot{\mathbf{u}}_t) + \mathbf{C}(a_4\dot{\mathbf{u}}_t + a_5\ddot{\mathbf{u}}_t) - \mathbf{n}(\mathbf{u}_t)$$

6. Solve for the displacement increments: $\delta\mathbf{u} = \hat{\mathbf{K}}^{-1}\hat{\mathbf{r}}_{t+\Delta t}$

Algorithm (2/2)

7. iterate for dynamic equilibrium:
   - 7.1 $i = i + 1$
   - 7.2 Evaluate the $(i-1)$-th approximation to the displacements, velocities and accelerations:

$$\mathbf{u}_{t+\Delta t}^{i-1} = \mathbf{u}_t + \delta\mathbf{u}^{i-1}; \qquad \dot{\mathbf{u}}_{t+\Delta t}^{i-1} = a_1\delta\mathbf{u}^{i-1} - a_4\dot{\mathbf{u}}_t - a_5\ddot{\mathbf{u}}_t$$

$$\ddot{\mathbf{u}}_{t+\Delta t}^{i-1} = a_0\delta\mathbf{u}^{i-1} - a_2\dot{\mathbf{u}}_t - a_3\ddot{\mathbf{u}}_t$$

   - 7.3 Evaluate the i-th residual force: $\psi_{t+\Delta t}^{i-1} = \mathbf{r}_{t+\Delta t} - (\mathbf{M}\ddot{\mathbf{u}}_{t+\Delta t}^{i-1} + \mathbf{C}\dot{\mathbf{u}}_{t+\Delta t}^{i-1} + \mathbf{n}(\mathbf{u}_{t+\Delta t}^{i-1})$
   - 7.4 Solve for the i-th corrected displacement: $\Delta\mathbf{u}^i = \hat{\mathbf{K}}^{-1}\psi_{t+\Delta t}^{i-1}$
   - 7.5 Evaluated the corrected displacement increments: $\delta\mathbf{u}^i = \delta\mathbf{u}^{i-1} + \Delta\mathbf{u}^i$
   - 7.6 Check for convergence of the iteration process $|\Delta\mathbf{u}^i|/|\mathbf{u}_t + \delta\mathbf{u}^i| \leq \varepsilon$