DDM for Nonlinear Dynamic Problems

The equations presented in this document are implemented in the programming language Python in the G2 code posted on this website. The governing system of equations for nonlinear dynamics reads

$$\mathbf{M}\ddot{\mathbf{u}} + \mathbf{C}\dot{\mathbf{u}} + \tilde{\mathbf{F}}(\mathbf{u}) = \mathbf{F} \tag{1}$$

where the symbols are explained in the document on linear dynamic DDM, except $\tilde{\mathbf{F}}$, which is the internal resisting force vector. From the document on the DDM for nonlinear static problems, the implicit and explicit dependence of $\tilde{\mathbf{F}}$ on the parameter θ is noted. The time-discretization scheme from the document on linear dynamic DDM is adopted also here, meaning that the space- and time-discretized system of equilibrium equations reads

$$\begin{aligned}\mathbf{M}(a_1\mathbf{u}_{n+1} + a_2\mathbf{u}_n + a_3\dot{\mathbf{u}}_n + a_4\ddot{\mathbf{u}}_n) + \mathbf{C}(a_5\mathbf{u}_{n+1} + a_6\mathbf{u}_n + a_7\dot{\mathbf{u}}_n \\ + a_8\ddot{\mathbf{u}}_n) + \tilde{\mathbf{F}}_{n+1} = \mathbf{F}_{n+1}\end{aligned} \tag{2}$$

Combining the derivations from the documents on nonlinear static DDM and linear dynamic DDM, the following system of equations for the response sensitivities, $\partial\mathbf{u}_{n+1}/\partial\theta$, emerges:

$$\begin{aligned}&\left(a_1\mathbf{M} + a_5\mathbf{C} + \frac{\partial\tilde{\mathbf{F}}_{n+1}}{\partial\mathbf{u}_{n+1}}\right)\frac{\partial\mathbf{u}_{n+1}}{\partial\theta} \\ &= \frac{\partial\mathbf{F}_{n+1}}{\partial\theta} - \frac{\partial\mathbf{M}}{\partial\theta}(a_1\mathbf{u}_{n+1} + a_2\mathbf{u}_n + a_3\dot{\mathbf{u}}_n + a_4\ddot{\mathbf{u}}_n) \\ &\quad - \mathbf{M}\left(a_2\frac{\partial\mathbf{u}_n}{\partial\theta} + a_3\frac{\partial\dot{\mathbf{u}}_n}{\partial\theta} + a_4\frac{\partial\ddot{\mathbf{u}}_n}{\partial\theta}\right) \\ &\quad - \frac{\partial\mathbf{C}}{\partial\theta}(a_5\mathbf{u}_{n+1} + a_6\mathbf{u}_n + a_7\dot{\mathbf{u}}_n + a_8\ddot{\mathbf{u}}_n) \\ &\quad - \mathbf{C}\left(a_6\frac{\partial\mathbf{u}_n}{\partial\theta} + a_7\frac{\partial\dot{\mathbf{u}}_n}{\partial\theta} + a_8\frac{\partial\ddot{\mathbf{u}}_n}{\partial\theta}\right) - \left.\frac{\partial\tilde{\mathbf{F}}(\mathbf{u}_{n+1})}{\partial\theta}\right|_{\mathbf{u}\text{ fixed}}\end{aligned} \tag{3}$$

Compared to linear dynamics, the finite elements now need to produce $\partial\tilde{\mathbf{F}}_{n+1}/\partial\theta$ for fixed $\mathbf{u}_{n+1}$ instead of the derivative of the stiffness. It is also stressed that the stiffness $\partial\tilde{\mathbf{F}}_{n+1}/\partial\mathbf{u}_{n+1}$ in the coefficient matrix in Eq. (3) must be the updated algorithmically consistent tangent stiffness matrix. In many cases, the coefficient matrix in the left-hand side of Eq. (3) is identical to that of the response equations. However, certain damping models require a modification; that is addressed in other documents posted near this one. In summary, the Newton-Raphson algorithm is amended as follows in order to calculate exact response sensitivities in inelastic dynamic finite element analysis:

Increments, i.e., time steps:

Determine load vector, $\mathbf{F}$

Newton-Raphson iterations:

Calculate tangent stiffness, $\frac{\partial \tilde{\mathbf{F}}}{\partial \mathbf{u}}$, and effective stiffness, $a_1\mathbf{M} + a_5\mathbf{C} + \frac{\partial \tilde{\mathbf{F}}}{\partial \mathbf{u}}$

Solve linear system of equations for displacement increment, $\Delta\mathbf{u}$

Calculate the trial displacements, $\mathbf{u}_{i+1} = \mathbf{u}_i + \Delta\mathbf{u}$

Conduct state determination, i.e., calculate $\tilde{\mathbf{F}}(\mathbf{u}_{n+1})$

Keep iterating unless the selected measure of the residual is zero

Upon convergence:

Make sure the current tangent stiffness is available

Do an LU-decomposition of the effective stiffness, for efficiency

Loop over variables, θ, for which response sensitivities are sought

Phase 1: Assemble the right-hand side of Eq. (3) for the relevant θ, which involves calculating the conditional derivative of the internal forces for fixed displacement $\mathbf{u}_{n+1}$ but *not* fixed $\mathbf{u}_n$.

Solve Eq. (3) for $\partial\mathbf{u}_{n+1}/\partial\theta$

Phase 2: Send $\partial\mathbf{u}_{n+1}/\partial\theta$ to elements to calculate unconditional derivatives of history variables for the relevant θ

Commit the converged state, i.e., store both history variables and their unconditional derivatives