Computational Plastic Capacity Analysis

Elsewhere on this website, documents are posted on hand calculation of plastic capacity of cross-sections, frames, and plates. There, the determination of the ultimate capacity of cross-sections, assuming the elastic-perfectly-plastic material model, is conducted by means of the lower-bound theorem. That is essentially an exercise in equilibrium. Conversely, the analysis of frames and plates applies the upper-bound theorem, and involves the assumption of plastic hinges for frames and yield lines for plates. This document describes an alternative to those hand calculation methods, following the computational approach described by Filippou & Fenves in their 2004 chapter in Bozorgnia and Bertero's book Methods of Analysis for Earthquake-Resistant Structures, following work published by Livesley in 1975. This approach starts by formulating the relationship between the degrees of freedom of the structure and the degrees of freedom of each element in its basic configuration. From the document on the computational stiffness method, that relationship is, for an individual element:

$$\mathbf{u}_{\mathrm{b}} = \mathbf{T}_{\mathrm{bl}}\mathbf{T}_{\mathrm{lg}}\mathbf{T}_{\mathrm{gf}}\mathbf{u}_{\mathrm{f}} \tag{1}$$

where $\mathbf{u}_{\mathrm{b}}$ includes only the rotational degrees of freedom, and the "all" configuration is omitted. The reason for both those facts is that we avoid the inclusion of axial deformations, and as a result, axial yielding. In other words, this formulation focuses on yielding in the form of plastic hinges, not axial crushing. It is the exclusion of axial deformations that lead to the omission of the "all" configuration and the need to establish $\mathbf{T}_{\mathrm{gf}}$ for each element, which requires special attention in computer implementations. That is because the input must reflect that the final degrees of freedom of the structure do *not* accommodate axial deformation. That is reflected in the Python code for computational plastic capacity analysis, posted near this document. For brevity, Eq. (1) is written

$$\mathbf{u}_{\mathrm{b}} = \mathbf{T}_{\mathrm{bf}}\mathbf{u}_{\mathrm{f}} \tag{2}$$

where $\mathbf{T}_{\mathrm{bf}}=\mathbf{T}_{\mathrm{bl}}\mathbf{T}_{\mathrm{lg}}\mathbf{T}_{\mathrm{gf}}$. Eqs. (1) and (2) represent kinematic compatibility. It is proven, via virtual work, in the document on the computational stiffness method, that equilibrium is represented by

$$\tilde{\mathbf{F}}_{\mathrm{f}} = \mathbf{T}_{\mathrm{bf}}^{\mathrm{T}}\mathbf{F}_{\mathrm{b}} \tag{3}$$

i.e., simply involving the transpose of $\mathbf{T}_{\mathrm{bf}}$. As described by Filippou & Fenves, the plastic capacity problem can be formulated either in terms of equilibrium, i.e., Eq. (3), or equivalently, in terms of kinematic compatibility, i.e., Eq. (2). Both approaches give the same result. Consider the equilibrium approach, and recall that the previously presented equations are expressed for a single element. The matrices $\mathbf{T}_{\mathrm{bf}}^{\mathrm{T}}$ for all elements are now stacked in the following manner:

$$\tilde{\mathbf{F}}_{\mathrm{f}} = \left[\mathbf{T}_{\mathrm{bf,1}}^{\mathrm{T}}, \mathbf{T}_{\mathrm{bf,2}}^{\mathrm{T}}, \mathbf{T}_{\mathrm{bf,3}}^{\mathrm{T}}, \cdots\right] \begin{Bmatrix} \tilde{\mathbf{F}}_{\mathrm{b,1}} \\ \tilde{\mathbf{F}}_{\mathrm{b,2}} \\ \tilde{\mathbf{F}}_{\mathrm{b,3}} \\ \cdots \end{Bmatrix} = \mathbf{T}_{\mathrm{bf,stacked}}^{\mathrm{T}} \tilde{\mathbf{F}}_{\mathrm{b,stacked}} \tag{4}$$

where the subscript "stacked" is attached to the larger stacked matrix and vector. The matrix $\mathbf{T}_{\mathrm{bf,stacked}}^{\mathrm{T}}$ connects the degrees of freedom of the structure to the basic degrees of freedom in all elements. Stated differently, and more appropriate for the equilibrium approach, $\mathbf{T}_{\mathrm{bf,stacked}}^{\mathrm{T}}$ connects the forces in the structure to all basic element forces. Equilibrium between externally applied loads along the final structural degrees of freedom and the basic element forces dictates that the external forces equals the internal forces, i.e., $\tilde{\mathbf{F}}_{\mathrm{f}}$, which means that

$$\lambda \cdot \mathbf{F}_{\mathrm{ref}} = \mathbf{T}_{\mathrm{bf,stacked}}^{\mathrm{T}} \tilde{\mathbf{F}}_{\mathrm{b,stacked}} \tag{5}$$

where λ=load factor and $\mathbf{F}_{\mathrm{ref}}$=reference load pattern. Interestingly, Eq. (5) can be regarded as a constraint in a linear programming optimization problem. To understand this, consider the following vector of unknowns: $\mathbf{x}=\{\lambda, \tilde{\mathbf{F}}_{\mathrm{b,1}}, \tilde{\mathbf{F}}_{\mathrm{b,2}}, \tilde{\mathbf{F}}_{\mathrm{b,3}}, \ldots\}^{\mathrm{T}}$, i.e., with the load factor prepended to the vector of basic element forces. The solution to a linear programming problem that maximizes λ subject to the equilibrium equality constraint in Eq. (5), and also to the inequality constraint that the absolute value of any basic element force cannot exceed the plastic capacity of the cross-section, M_u, gives the plastic capacity of the structure, in terms of λ, as well as the identification of the member ends that experience yielding at that load, i.e., the "mechanism," in terms of values of $\tilde{\mathbf{F}}_{\mathrm{b,1}}$, $\tilde{\mathbf{F}}_{\mathrm{b,2}}$, $\tilde{\mathbf{F}}_{\mathrm{b,3}}$,… This is shown in an example posted near this document, where it is also visualized that the columns of $\mathbf{T}_{\mathrm{bf,stacked}}^{\mathrm{T}}$ identify the independent plastic mechanisms of the structure. The number of mechanisms equal the number of final degrees of freedom of the structure, and it is the displacement degrees of freedom that give the most relevant mechanisms. The solution to the linear programming problem often consists of a combination of the independent mechanisms.