Finite Element Analysis Flow Chart

Variational Calculus,
Essential BC
(Euler 1750)

Governing Partial Differential Equation,
Essential, Non-essential Boundary Conditions

Virtual Work
Essential BC
(Bernoulli 1740)

Methods of Weighted Residuals:
Least Squares Method (1900)
Galerkin Method (1930)
Method of Moments

. . .

Governing Integral Form (includes non-essential BC), Essential BC

$$I = \int_\Omega f(\vec{x})\, d\Omega \;+\; \int_\Gamma g(\vec{x})\, d\Gamma \;\begin{cases} = 0, \\ or \\ \rightarrow minimum \end{cases}$$

Assume spatial form in $\boldsymbol{\Omega}$, with unknown constants, *that satisfy EBC*.
Substitute it and integrate that form to create a matrix system.
Solve matrix system for unknown constants.

- -Math methods -
- -FEA methods (1960) -

Mesh the domain, Ω, (and its boundary, Γ) with a non-overlapping union of elements:
$$\boldsymbol{\Omega} = \cup_e \boldsymbol{\Omega^e}, \qquad \boldsymbol{\Gamma} = \cup_b \boldsymbol{\Gamma^b}, \;\; \boldsymbol{\Omega^e} = finite\ element, \qquad \boldsymbol{\Gamma^b} \subset \boldsymbol{\Omega^e}$$

$$I = \sum_e \int_{\Omega^e} \boldsymbol{f}\, d\boldsymbol{\Omega} \;+\; \sum_b \int_{\Gamma^b} \boldsymbol{g}\, d\boldsymbol{\Gamma}\, , plus\ \text{EBC:} \;\begin{cases} = \boldsymbol{0}, \\ or \\ \rightarrow \boldsymbol{minimum} \end{cases}$$

$$I = \sum_e I^e + \sum_b I^b$$

Each element has nodes. A connection list of nodes on each element defines the mesh.
The degrees of freedom (DOF) at each node define the system unknowns, $\boldsymbol{\delta}\ (\boldsymbol{n_d \times 1})$
and the sub-set of element unknowns, $\boldsymbol{\delta^e}\ (\boldsymbol{n_i \times 1})$, so $\boldsymbol{\delta^e} \subset_e \boldsymbol{\delta}.$
The element sub-set of DOF is defined by its connection list:

$$\boldsymbol{\delta^e} = \boldsymbol{\beta_e\, \delta}, \;\; \boldsymbol{\beta_e} \Leftarrow Boolean\ connection\ list\ for\ element\ e$$
$$\boldsymbol{(n_i \times 1)} = \boldsymbol{(n_i \times n_d)\ (n_d \times 1)}$$

In an element, and on its boundary, the solution is interpolated[*] from the nodal DOF, $\boldsymbol{\delta^e}$:
$$\boldsymbol{u(\vec{x}) = H(\vec{x})\, \delta^e = u(\vec{x})^T = \delta^{eT} H(\vec{x})^T}$$
$$\boldsymbol{(1 \times 1) = (1 \times n_i)\ (n_i \times 1)}$$
Unknown at $\vec{x}$ = (interpolation functions at $\vec{x}$) times (the element nodal DOF).

Finite Element Analysis Flow Chart, 2

That interpolation in space also defines the gradient of the solution, in n_s physical spatial dimensions, within that element domain, $\mathbf{\Omega}^e$:

$$\frac{\partial \boldsymbol{u}(\vec{\boldsymbol{x}})}{\partial \vec{\boldsymbol{x}}} = \vec{\nabla}\,\boldsymbol{u}(\vec{\boldsymbol{x}}) = \frac{\partial \boldsymbol{H}(\vec{\boldsymbol{x}})\,\boldsymbol{\delta}^e}{\partial \vec{\boldsymbol{x}}} \equiv \boldsymbol{B}^e(\vec{\boldsymbol{x}})\,\boldsymbol{\delta}^e = \boldsymbol{\delta}^{e^T}\boldsymbol{B}^e(\vec{\boldsymbol{x}})^T$$

$$(\boldsymbol{n_s} \times \boldsymbol{1}) = (\boldsymbol{n_s} \times \boldsymbol{n_i})\,(\boldsymbol{n_i} \times \boldsymbol{1})$$

Substitute the spatial solution and its gradient into the integral form.
For PDE coefficients $c_1, c_2,$ and c_3, typically:

$$f \longrightarrow f^e = \boldsymbol{\delta}^{e^T}\boldsymbol{B}^e(\vec{\boldsymbol{x}})^T\,c_1\boldsymbol{B}^e(\vec{\boldsymbol{x}})\,\boldsymbol{\delta}^e + \boldsymbol{\delta}^{e^T}\boldsymbol{H}(\vec{\boldsymbol{x}})^T c_2\boldsymbol{H}(\vec{\boldsymbol{x}})\,\boldsymbol{\delta}^e + \boldsymbol{\delta}^{e^T}\boldsymbol{H}(\vec{\boldsymbol{x}})^T c_3$$

Often a boundary differential equation, with coefficients b_1 and b_2, defines the non-essential boundary condition. Typically, $\boldsymbol{H}^b(\vec{\boldsymbol{x}}) \subset \boldsymbol{H}(\vec{\boldsymbol{x}})\ and\ \boldsymbol{\delta}^b \subset_b \boldsymbol{\delta}$:

$$g \longrightarrow g^b = \boldsymbol{\delta}^{b^T}\boldsymbol{H}^b(\vec{\boldsymbol{x}})^T\,b_1\boldsymbol{H}^b(\vec{\boldsymbol{x}})\,\boldsymbol{\delta}^b + \boldsymbol{\delta}^{b^T}\boldsymbol{H}^b(\vec{\boldsymbol{x}})^T b_2$$

The constants $\boldsymbol{\delta}^e$ and $\boldsymbol{\delta}^b$ move outside the integrals. Integrating these terms (usually numerically) converts the integral form to the matrix form:

$$\boldsymbol{S}^e = \int_{\Omega^e}[\boldsymbol{B}^e(\vec{\boldsymbol{x}})^T\,c_1\boldsymbol{B}^e(\vec{\boldsymbol{x}}) + \boldsymbol{H}(\vec{\boldsymbol{x}})^T c_2\boldsymbol{H}(\vec{\boldsymbol{x}})\,]\,d\Omega, \quad \boldsymbol{C}^e = \int_{\Omega^e}\{\,\boldsymbol{H}(\vec{\boldsymbol{x}})^T c_3\}\,d\Omega$$

$$\boldsymbol{S}^e = \int_{\Gamma^b}[\boldsymbol{H}^b(\vec{\boldsymbol{x}})^T\,b_1\boldsymbol{H}^b(\vec{\boldsymbol{x}})\,]\,d\Gamma, \quad \boldsymbol{C}^e = \int_{\Gamma^b}\{\,\boldsymbol{H}^b(\vec{\boldsymbol{x}})^T b_2\}\,d\Gamma$$

$$I^e(\boldsymbol{\delta}^e) = \boldsymbol{\delta}^{e^T}\boldsymbol{S}^e\,\boldsymbol{\delta}^e + \boldsymbol{\delta}^{e^T}\boldsymbol{C}^e, \quad I^b(\boldsymbol{\delta}^b) = \boldsymbol{\delta}^{b^T}\boldsymbol{S}^b\,\boldsymbol{\delta}^b + \boldsymbol{\delta}^{b^T}\boldsymbol{C}^b$$

Looping over all elements and scattering to the system degrees of freedom, $\boldsymbol{\delta}$, gives the governing large system matrix form:

Galerkin $\qquad\qquad\qquad\qquad I(\boldsymbol{\delta}) = \sum \boldsymbol{\delta}^T\boldsymbol{S}\,\boldsymbol{\delta} + \sum \boldsymbol{\delta}^T\boldsymbol{C} = \boldsymbol{0}$

Least Squares, Variational $\qquad I(\boldsymbol{\delta}) = \frac{1}{2}\sum \boldsymbol{\delta}^T\boldsymbol{S}\,\boldsymbol{\delta} + \sum \boldsymbol{\delta}^T\boldsymbol{C} \rightarrow minimum$

Setting the Galerkin form to zero or rendering the Least Square form stationary gives the final matrix system (before essential BC):

$$\boldsymbol{S\,\delta} = \boldsymbol{C},\ \boldsymbol{\delta} \neq \boldsymbol{S}^{-1}\,\boldsymbol{C}$$

Here, all the terms in $\boldsymbol{S}$ and $\boldsymbol{C}$ are known from the element integrals.
However, $\boldsymbol{S}$ is usually singular and cannot be inverted until the essential boundary conditions are enforced (next).

Finite Element Analysis Flow Chart, 3

If EBC are present the equations need to be re-arranged in partitioned form

$$\begin{bmatrix} S_{uu} & S_{uk} \\ S_{ku} & S_{kk} \end{bmatrix} \begin{Bmatrix} \delta_u \\ \delta_k \end{Bmatrix} = \begin{Bmatrix} C_u \\ C_k + R_k \end{Bmatrix}$$

where δ_u = unknown DOF, and δ_k = essential boundary values of the DOF.

R_k = reactions associated with essential boundary conditions.

The only unknowns in this matrix system are the vectors δ_u and R_k.
The upper partition gives the unknown DOFs:

$$S_{uu}\delta_u + S_{uk}\delta_k = C_u, \quad \delta_u = S_{uu}^{-1}(C_u - S_{uk}\delta_k).$$

The lower partition gives the reactions at essential boundary conditions:

$$R_k = S_{ku}\delta_u + S_{kk}\delta_k - C_k$$

Now that all DOF are known, the post-process phase starts.
Loop over each element and gather its DOF, $\delta^e \subset_e \delta$.
Recover the solution gradient at points in the element:

$$\vec{\nabla}\, u(\vec{x}) = \frac{\partial H(\vec{x})\, \delta^e}{\partial \vec{x}} \equiv B^e(\vec{x})\, \delta^e$$

Often, they are used to define other physical terms.
Local 'flux' values are found using the PDE coefficients:

$$\sigma(\vec{x}) = c_1 B^e(\vec{x})\, \delta^e$$

and are output for plotting at point $\vec{x}$.

Occasionally, the solution integral is required:

$$\int_\Omega u(\vec{x})\, d\Omega = \sum_e \int_{\Omega^e} H(\vec{x})\, d\Omega\; \delta^e.$$

------- end ------
--- Notes ---
* If curved elements are required, it is necessary to use
non-dimensional coordinates, $\vec{r}$, to interpolate the physical ones:

$$x(\vec{r}) = H(\vec{r})\, x^e, \qquad \frac{\partial}{\partial x} = \frac{\partial}{\partial r}\frac{\partial r}{\partial x}$$

If an error analysis and adaption is required, again loop over all elements,
form a patch of elements, least square fit the discontinuous gradients
to form a continuous gradient in the patch. Integrate the square of the
difference between continuous gradient and the element gradient.
Process that for the error. Compute new element sizes.
Repeat the analysis until an acceptable error is obtained.