



Meshless Methods

Soheil Mohammadi

Professor of Computational Mechanics

School of Civil Engineering

University of Tehran

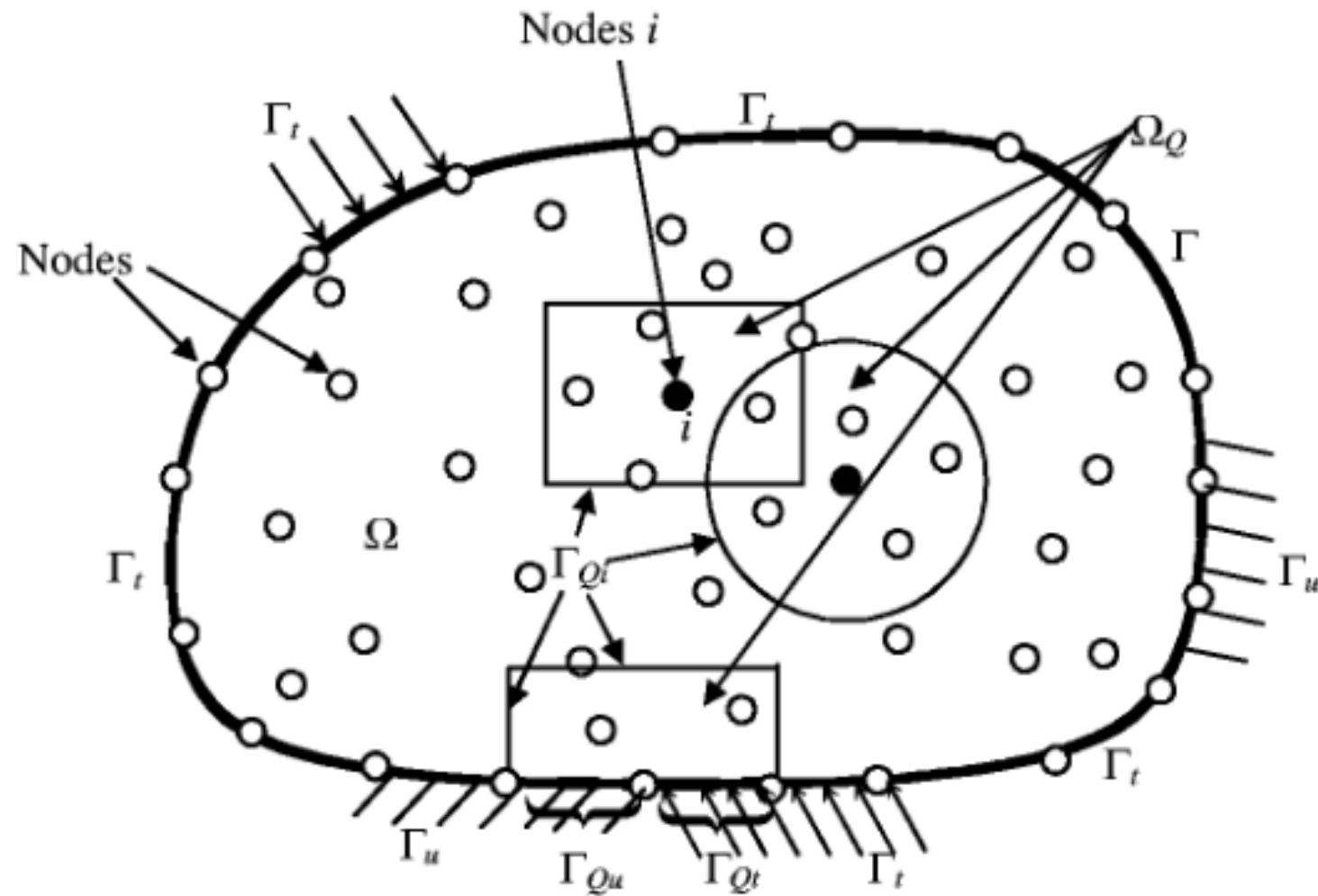
Tel: +98-21-6111 2258, Fax: +98-21-6640 3808, Cell: +98-912-5030 392

Email: smoham@ut.ac.ir, <http://eng.ut.ac.ir/civ/smohammadi>

Meshless Local Petrov-Galerkin (MLPG):

1. Developed by Atluri-Zhu (1998).
2. The aim is to satisfy a required equation point by point using information in a local domain of the points.
3. MLPG is based on a local weak form of the Petrov-Galerkin residual formulation.
4. In the Petrov-Galerkin formulation, the weight(test) and trial functions are selected independently, whereas they are the same in Galerkin formulation.
5. MLPG uses MLS approximation for construction of shape functions.
6. MLPG is similar to collocation methods, but it is more stable due to the use of locally weighted residual integration.

Meshless Local Petrov-Galerkin (MLPG):



Meshless Local Petrov-Galerkin (MLPG):

- The strong form of equilibrium equation can be defined as:

$$\sigma_{ij,j} + b_i = 0 \quad , i, j = x, y$$

with

$$u_i = \bar{u}_i \quad \text{on } \Gamma_u$$

$$\sigma_{ij} n_j = \bar{t}_i \quad \text{on } \Gamma_t$$

- The local weighted residual form for a node I can be expressed as:

$$\int_{\Omega_Q} (\sigma_{ij,j} + b_i) \hat{W}_I d\Omega - \alpha \int_{\Gamma_{\Omega_u}} (u_i - \bar{u}_i) \hat{W}_I d\Gamma = 0$$

where a penalty method is used to enforce the boundary condition.

Meshless Local Petrov-Galerkin (MLPG):

- Using the divergence theorem:

$$\int_{\Gamma_Q} \sigma_{ij} n_j \hat{W}_I d\Gamma + \int_{\Omega_Q} \sigma_{ij} \hat{W}_{I,j} d\Omega + \int_{\Omega_Q} b_i \hat{W}_I d\Omega - \alpha \int_{\Gamma_{\Omega_u}} (u_i - \bar{u}_i) \hat{W}_I d\Gamma = 0$$

- $\hat{W}$ is purposely selected in such a way that it vanishes on Γ_{Q_i} .
- Cubic and quartic splines and other weight functions can be chosen to satisfy the above condition along the boundaries of the internal quadrature domains.
- If $\hat{W} = \delta$, MLPG becomes a collocation method.
- The differentiation on σ_{ij} is now transferred to the weight function. This is highly important for consistency requirements.

Meshless Local Petrov-Galerkin (MLPG):

- Using a W that vanishes on Γ_{Q_i} :

$$\begin{aligned} & \int_{\Omega_Q} \sigma_{ij} \hat{W}_{I,j} d\Omega + \alpha \int_{\Gamma_{\Omega_u}} u_i \hat{W}_I d\Gamma - \int_{\Gamma_{Q_u}} \sigma_{ij} n_j \hat{W}_I d\Gamma \\ &= \int_{\Gamma_{\Omega_t}} \bar{t}_i \hat{W}_I d\Gamma + \alpha \int_{\Gamma_{\Omega_u}} \bar{u}_i \hat{W}_I d\Gamma + \int_{\Omega_Q} b_i \hat{W}_I d\Omega \end{aligned}$$

- and if Ω_Q is located entirely in the domain:

$$\int_{\Omega_Q} \sigma_{ij} \hat{W}_{I,j} d\Omega = \int_{\Omega_Q} b_i \hat{W}_I d\Omega$$

- Therefore, the strong form $\sigma_{ij,j} + b_i = 0$ has been replaced by a relaxed weak form with integration over a small local quadrature domain.

Meshless Local Petrov-Galerkin (MLPG):

- MLPG guarantees satisfaction of the equilibrium equation at a node in an integral sense over a quadrature domain.
- The size of the quadrature domain determines the relaxing extent of the strong form of the differential equation.
 - For accurate and stable solution, sufficient dimension of the quadrature domain is required.
 - Too large support domain does not necessarily provide better results.
- In MLPG, the equilibrium equation and the boundary conditions are satisfied node by node in a weak sense of the local weighted residual.
- This node by node local formulation provides freedom in solving the system equation numerically.

Meshless Local Petrov-Galerkin (MLPG):

- The local weak form can be re-written as:

$$\begin{aligned} & \int_{\Omega_Q} \hat{W}_{I,j}^T \sigma d\Omega + \alpha \int_{\Gamma_{\Omega_u}} \hat{W}_I u d\Gamma - \int_{\Gamma_{Q_u}} \hat{W}_I t d\Gamma \\ &= \int_{\Gamma_{\Omega_t}} \hat{W}_I \bar{t} d\Gamma + \alpha \int_{\Gamma_{\Omega_u}} \hat{W}_I \bar{u} d\Gamma + \int_{\Omega_Q} \hat{W}_I b d\Omega \end{aligned}$$

or

$$\begin{aligned} & \int_{\Omega_Q} \hat{V}_I^T \sigma d\Omega + \alpha \int_{\Gamma_{\Omega_u}} \hat{W}_I u d\Gamma - \int_{\Gamma_{Q_u}} \hat{W}_I t d\Gamma \\ &= \int_{\Gamma_{\Omega_t}} \hat{W}_I \bar{t} d\Gamma + \alpha \int_{\Gamma_{\Omega_u}} \hat{W}_I \bar{u} d\Gamma + \int_{\Omega_Q} \hat{W}_I b d\Omega \end{aligned}$$

with

$$\hat{V}_I = \begin{bmatrix} \hat{W}_{I,x} & 0 \\ 0 & \hat{W}_{I,y} \\ \hat{W}_{I,y} & \hat{W}_{I,x} \end{bmatrix}$$

Meshless Local Petrov-Galerkin (MLPG):

- Using MLS approximation to construct the shape functions for each node using nodes in the support domain Ω_s of a point (and not necessarily a node): ($\Omega_s \neq \Omega_Q$)

$$u^h = \sum_{i=1}^n \Phi_i u_i$$

- Then the weak form is modified into:

$$\begin{aligned} \int_{\Omega_Q} \hat{V}_I^T \left(C \sum_{j=1}^n B_j u_j \right) d\Omega + \alpha \int_{\Gamma_{\Omega_u}} \hat{W}_I \left(\sum_{j=1}^n \Phi_j u_j \right) d\Gamma \\ - \int_{\Gamma_{Q_u}} \hat{W}_I \left(C \sum_{j=1}^n B_j u_j \right) d\Gamma = \int_{\Gamma_{\Omega_t}} \hat{W}_I \bar{t} d\Gamma + \alpha \int_{\Gamma_{\Omega_u}} \hat{W}_I \bar{u} d\Gamma + \int_{\Omega_Q} \hat{W}_I b d\Omega \end{aligned}$$

Meshless Local Petrov-Galerkin (MLPG):

where:

$$B_j = \begin{bmatrix} \phi_{j,x} & 0 \\ 0 & \phi_{j,y} \\ \phi_{j,y} & \phi_{j,x} \end{bmatrix} \quad \hat{W}_I = \begin{bmatrix} \hat{W}_I & 0 \\ 0 & \hat{W}_I \end{bmatrix}$$

$$\sigma = C\varepsilon = C \begin{bmatrix} \partial/\partial x & 0 \\ 0 & \partial/\partial y \\ \partial/\partial y & \partial/\partial x \end{bmatrix} \sum_{j=1}^n \Phi_j u_j = C \sum_{j=1}^n B_j u_j$$

$$n = \begin{bmatrix} n_x & 0 & n_y \\ 0 & n_y & n_x \end{bmatrix}$$

$$t = n\sigma = nC \sum_{j=1}^n B_j u_j$$

Meshless Local Petrov-Galerkin (MLPG):

- The final matrix form of the equation for node I can be written as:

$$\sum_{j=1}^n K_{Ij} u_j = f_I$$

with

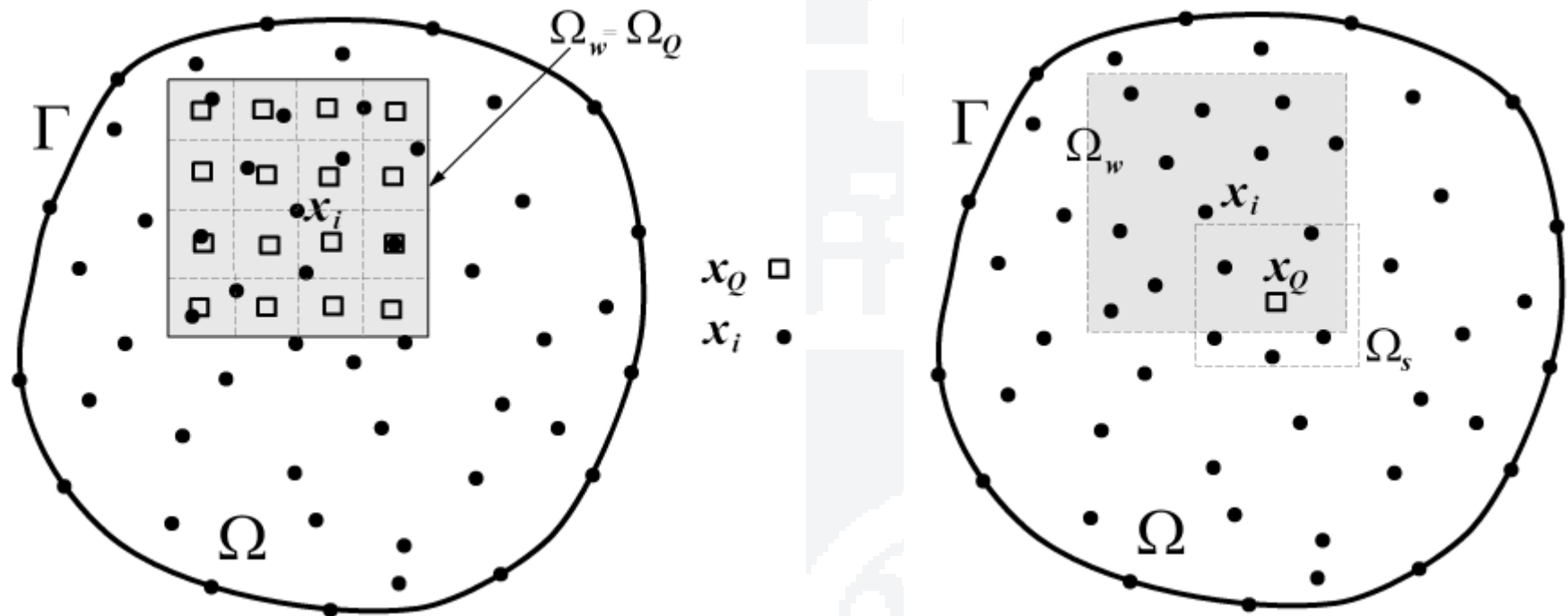
$$K_{Ij} = \int_{\Omega_Q} \hat{V}_I^T C B_j d\Omega + \alpha \int_{\Gamma_{\Omega_u}} \hat{W}_I \Phi_j d\Gamma - \int_{\Gamma_{Q_u}} \hat{W}_I n C B_j d\Gamma$$

$$f_I = \int_{\Omega_Q} \hat{W}_I b d\Omega + \int_{\Gamma_{\Omega_t}} \hat{W}_I \bar{t} d\Gamma + \alpha \int_{\Gamma_{\Omega_u}} \hat{W}_I \bar{u} d\Gamma$$

- Assembling for all n_t nodes:

$$K_{2n_t \times 2n_t} U_{2n_t \times 1} = F_{2n_t \times 1}$$

Meshless Local Petrov-Galerkin (MLPG):



Meshless Local Petrov-Galerkin (MLPG):

- The procedure of solving a problem by MLPG can be summarized as:
- For any node I , determine the weight and integration domain $\Omega_w = \Omega_Q$
 - Find the quadrature points x_Q within the integration domain.
 - Define the support domain Ω_s of the quadrature point x_Q
 - Compute corresponding MLS shape functions
 - Evaluate the integrands numerically.
 - Assemble in global K and F
- Solve the global equation $KU = F$

Meshless Local Petrov-Galerkin (MLPG):

- The global stiffness matrix is banded as long as the support domain is compact.
- K is usually not symmetric. This is a major drawback of the MLPG.
- MLPG is more stable than strong form solutions even for irregular nodal arrangements.
- MLS shape functions have a complex feature, with different forms in each small integration domain.
- Overlapping of interpolation domain makes the integrand in that region very complicated.
- To improve the accuracy of the numerical integration, the quadrature domain Ω_Q should be divided into small regular partitions, with more Gauss points in each partition.

Meshless Local Petrov-Galerkin (MLPG):

- As a simple rule of thumb:
 - the dimension of the rectangular quadrature domain can be taken as $1.5-2.5$ times the average nodal spacing in the vicinity of node I .
 - the quadrature domain is divided evenly into 2×2 cells.
 - each cell contains 4×4 Gauss points.
- MLPG is reproductive.
- MLPG has difficulties near boundaries where the local quadrature domains may intersect with the global boundary, creating complex geometry quadrature domains.
- MLPG requires expensive MLS computations.

Meshless Local Petrov-Galerkin (MLPG):

- MLPG stiffness matrix is un-symmetric.
- MLPG with penalty formulation, only approximately satisfies the boundary conditions. (the accuracy is acceptable for most engineering applications).
- The nodal nature of MLPG allows for direct imposition of essential boundary conditions. Equations associated to the boundary nodes, are simply replaced by the boundary conditions.
 - The essential boundary conditions are satisfied exactly.
 - This method destroys the symmetry of the stiffness matrix, which is, of course, not a problem for non-symmetric MLPG.

Dynamic MLPG:

- The strong form of dynamic equilibrium equation can be defined as:

$$\sigma_{ij,j} + b_i = \rho \ddot{u}_i + \eta_c \dot{u}_i \quad , i, j = x, y$$

with

$$\sigma_{ij} n_j = \bar{t}_i \quad \text{on } \Gamma_t$$

$$u_i = \bar{u}_i \quad \text{on } \Gamma_u$$

$$u_i(x, t_0) = \bar{u}_i(x) \quad x \in \Omega$$

$$\dot{u}_i(x, t_0) = \bar{\dot{u}}_i(x) \quad x \in \Omega$$

Dynamic MLPG:

- The local weighted residual form for a node I can be expressed as

$$\int_{\Omega_Q} (\sigma_{ij,j} + b_i - \rho \ddot{u}_i - \eta_c \dot{u}_i) \hat{W} - d\Omega - \alpha \int_{\Gamma_{\Omega_u}} (u_i - \bar{u}_i) \hat{W} - d\Gamma = 0$$

- Following the same procedure as the static MLPG, the discretized dynamic MLPG is obtained:

$$M\ddot{U}(t) + C\dot{U}(t) + KU(t) = F(t)$$

where,

Dynamic MLPG:

$$K_{Ij} = \int_{\Omega_Q} \hat{V}_I^T C B_j d\Omega - \int_{\Gamma_{Qu}} \hat{W}_I n C B_j d\Gamma + \alpha \int_{\Gamma_{\Omega_u}} \hat{W}_I \Phi_j d\Gamma$$

$$M_{Ij} = \int_{\Omega_Q} \rho \hat{W}_I \Phi_j d\Omega$$

$$C_{Ij} = \int_{\Omega_Q} \eta_c \hat{W}_I \Phi_j d\Omega$$

$$f_I = \int_{\Omega_Q} \hat{W}_I b d\Omega + \int_{\Gamma_{\Omega_t}} \hat{W}_I \bar{t} d\Gamma + \alpha \int_{\Gamma_{\Omega_u}} \hat{W}_I \bar{u} d\Gamma$$

