汽车车身先进设计制造国家重点实验室（湖南大学）
State Key Laboratory of Advanced Design and Manufacturing for Vehicle Body(Hunan University)

The fast HdBNM--a way toward automatic simulation

Jianming Zhang, Zhenhan Yao

13-17 Oct 2008

Motivation

Two main difficulties in numerical analysis of engineering structures

- Mesh generation
- Large computational scale

➢ To overcome the first difficulty, we proposed the

Hybrid Boundary Node Method (HdBNM)

➢ To overcome the second difficulty, we implemented the

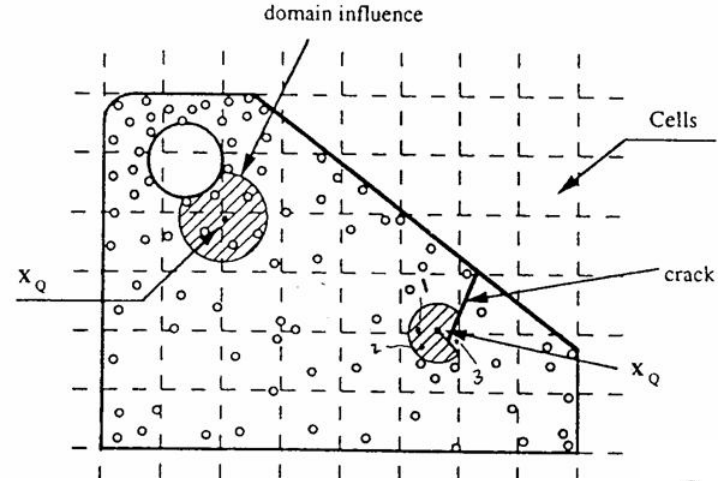
Fast Multipole Techniques (FMM)

Integration schemes

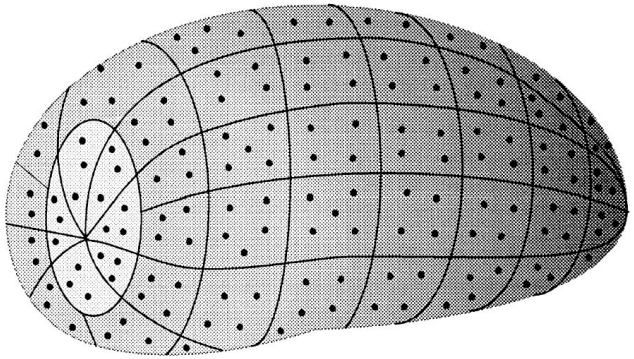
Pseudo meshless or truly meshless ?

Introduction to HdBNM

- Element free Galerkin method (EFG)



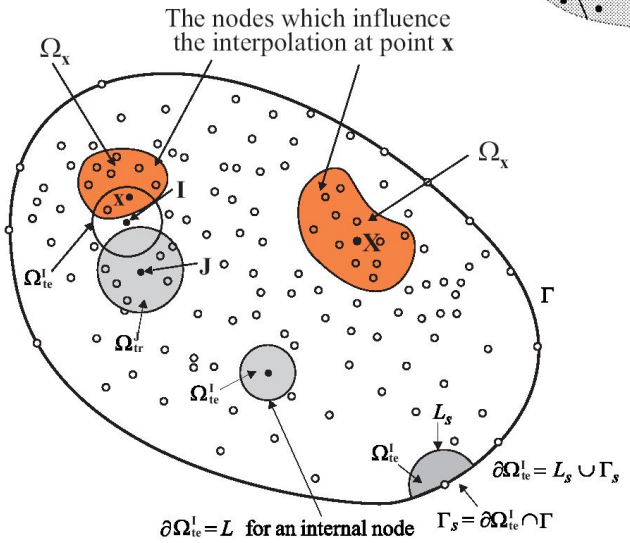
- Boundary Node Method (BNM)



- Meshless Local Boundary Integral Equation (MLBIE)

and

- The Meshless Local Petrov-Galerkin (MLPG) Method



Introduction to HdBNM (2)

	Domain type	Boundary type
Pseudo meshless	Element free Galerkin method	Boundary node method
Truly meshless	Meshless Local Boundary Integral Equation	?

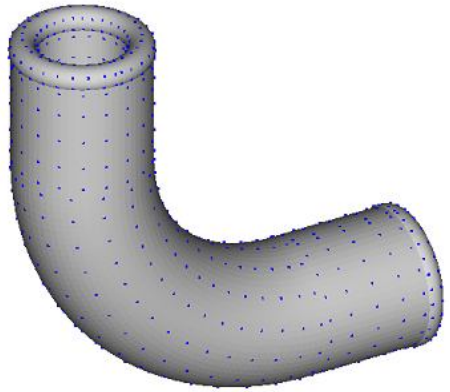
The answer is positive:

Hybrid Boundary Node Method

Formulations of HdBNM

Main features:

- Combines modified functional with the *Moving Least Squares* (MLS) approximation
- Boundary-only truly meshless method



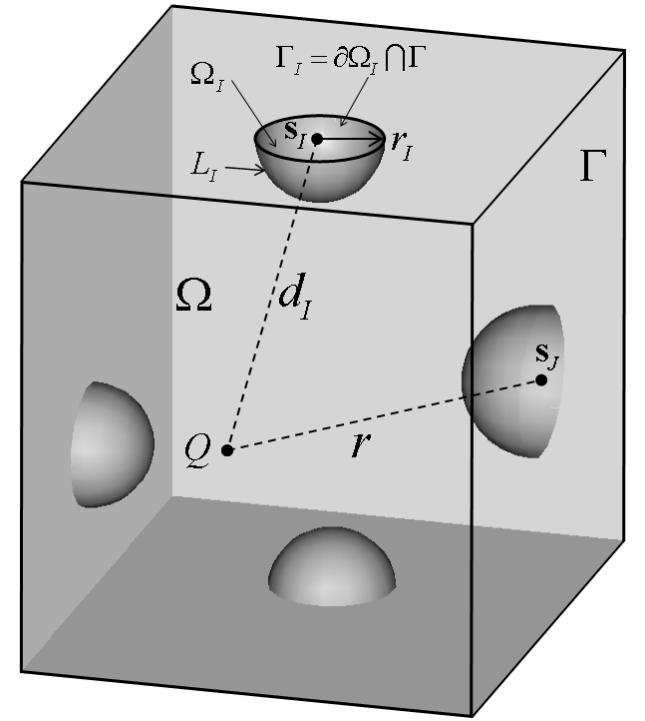
Example of meshless discretization

- Three independent variables
  - internal temperature $u = \sum_{J=1}^{N} u_J^s x_J \quad u_J^s = \frac{1}{\kappa} \frac{1}{4\pi r(Q, \mathbf{s}_J)}$
  - boundary temperature $\tilde{u}(\mathbf{s}) = \sum_{J=1}^{N} \Phi_J(\mathbf{s}) \hat{u}_J$
  - boundary normal flux $\tilde{q}(\mathbf{s}) = \sum_{J=1}^{N} \Phi_J(\mathbf{s}) \hat{q}_J$

Formulations of HdBNM (2)

➢Local weak form

$$\int_\Gamma (q-\tilde{q})\delta u d\Gamma - \int_\Omega u_{,ii}\delta u d\Omega + \int_{\Gamma_q} (\tilde{q}-\bar{q})\delta\tilde{u} d\Gamma - \int_\Gamma (u-\tilde{u})\delta\tilde{q} d\Gamma = 0$$



$$\sum_{J=1}^{N}\int_{\Gamma_I} u_J^s v_I(Q)x_J d\Gamma = \sum_{J=1}^{n}\int_{\Gamma_I} \Phi_J(\mathbf{s})v_I(Q)\hat{u}_J d\Gamma$$

$$\sum_{J=1}^{N}\int_{\Gamma_I} \frac{\partial u_J^s}{\partial n} v_I(Q)x_J d\Gamma = \sum_{J=1}^{n}\int_{\Gamma_I} \Phi_J(\mathbf{s})v_I(Q)\hat{q}_J d\Gamma$$

Formulations of HdBNM (3)

➢System of equations – final form

$$\mathbf{Ux} = \mathbf{H\hat{u}}$$

$$\mathbf{Qx} = \mathbf{H\hat{q}}$$

where

$$U_{IJ} = \int_{\Gamma_I} u_J^s v_I(Q) d\Gamma$$

$$Q_{IJ} = \int_{\Gamma_I} \frac{\partial u_J^s}{\partial n} v_I(Q) d\Gamma$$

$$H_{IJ} = \int_{\Gamma_I} \Phi_J(\mathbf{s}) v_I(Q) d\Gamma$$

➢Solution procedure

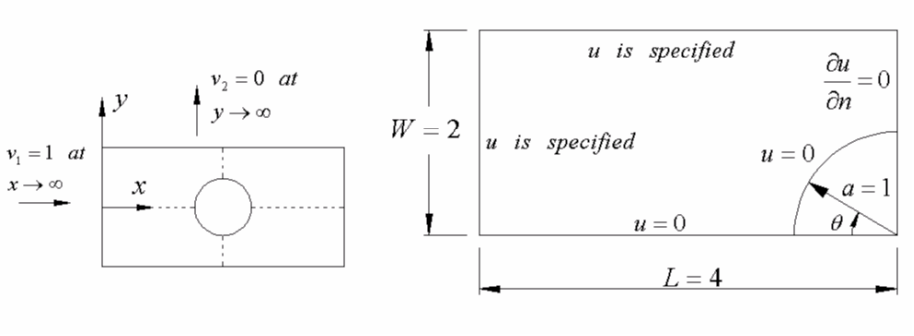
$$\left.\begin{array}{l}\mathbf{Ux} = \mathbf{H\hat{u}} \\ \mathbf{Qx} = \mathbf{H\hat{q}}\end{array}\right\} \rightarrow \mathbf{Ax} = \mathbf{d} \rightarrow \mathbf{x} = \mathbf{A}^{-1}\mathbf{d}$$

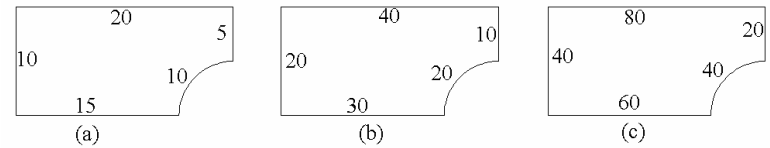
$$\mathbf{Ux} = \mathbf{H\hat{u}} \rightarrow \hat{\mathbf{u}} = \mathbf{H}^{-1}\mathbf{Ux}$$

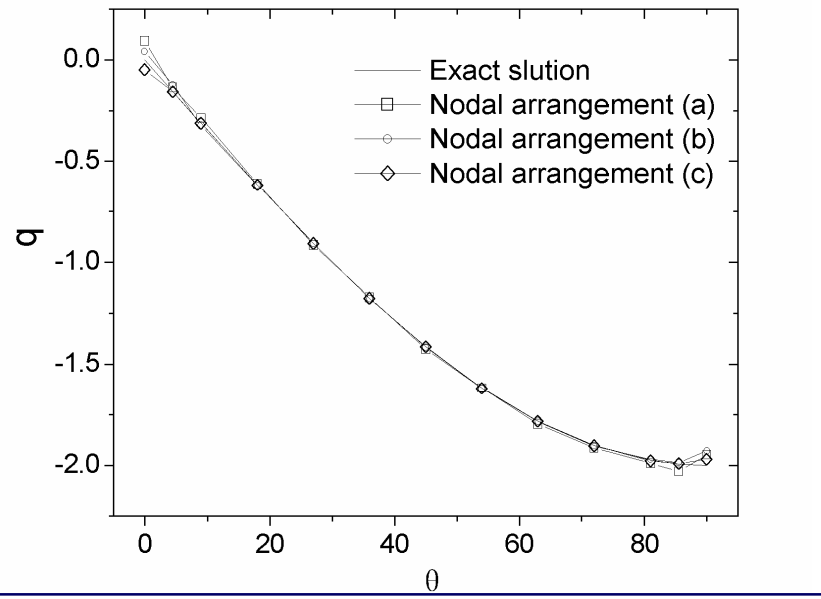
$$\mathbf{Qx} = \mathbf{H\hat{q}} \rightarrow \hat{\mathbf{q}} = \mathbf{H}^{-1}\mathbf{Qx}$$

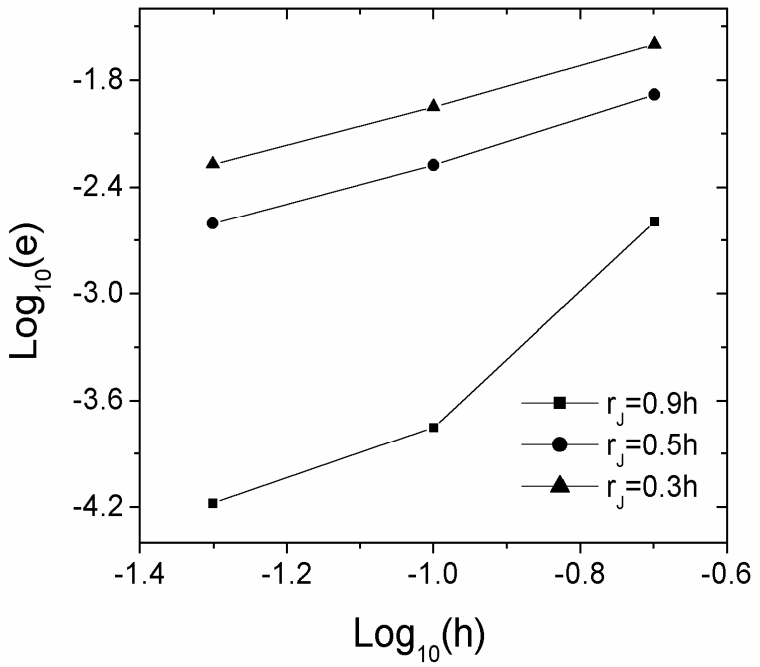
Numerical results from HdBNM (2D potential problem)

- Potential flow



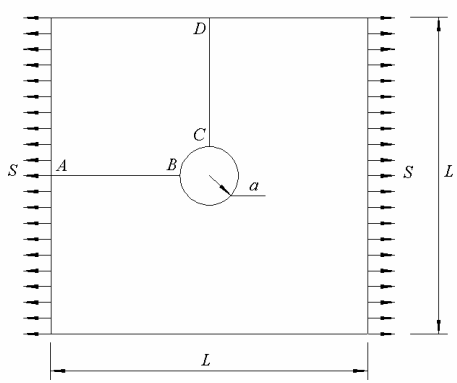


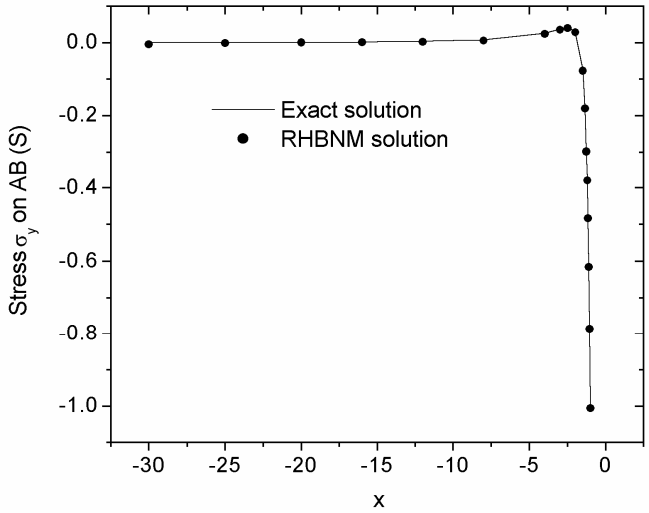




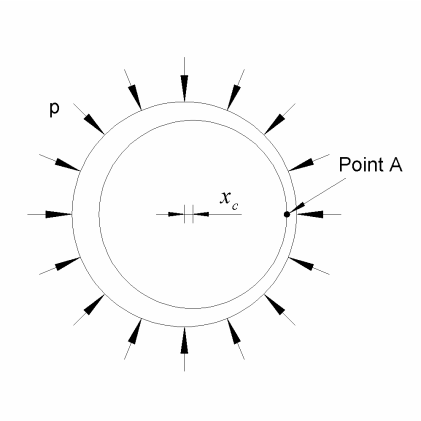
Numerical results from HdBNM (2D elasticity problem)

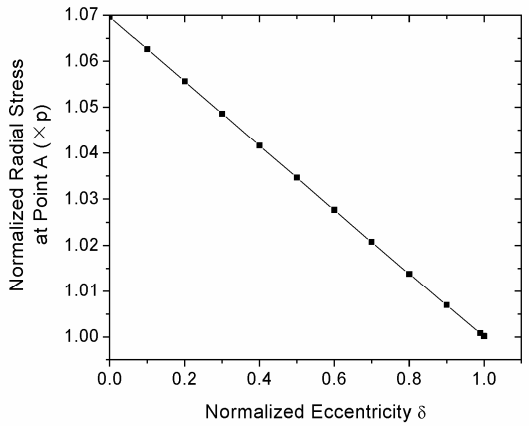
- 2D Lame problem

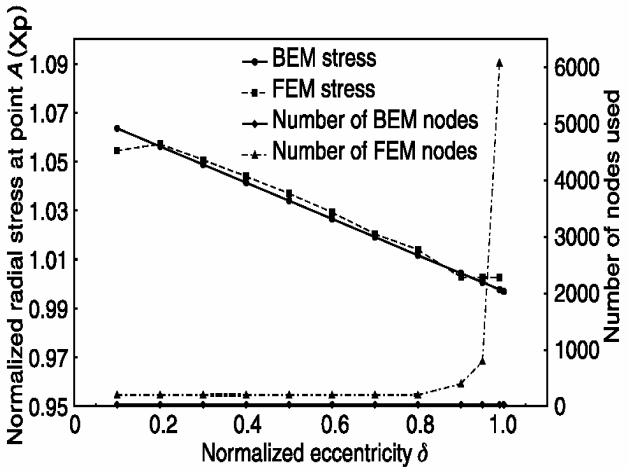




- Eccentric coating







Numerical results from HdBNM (3D potential problem)

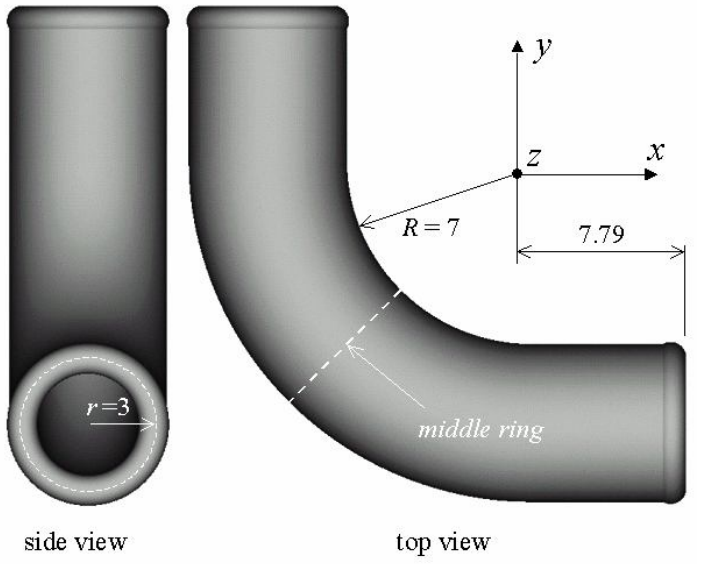
- Dirichlet problem in an elbow pipe

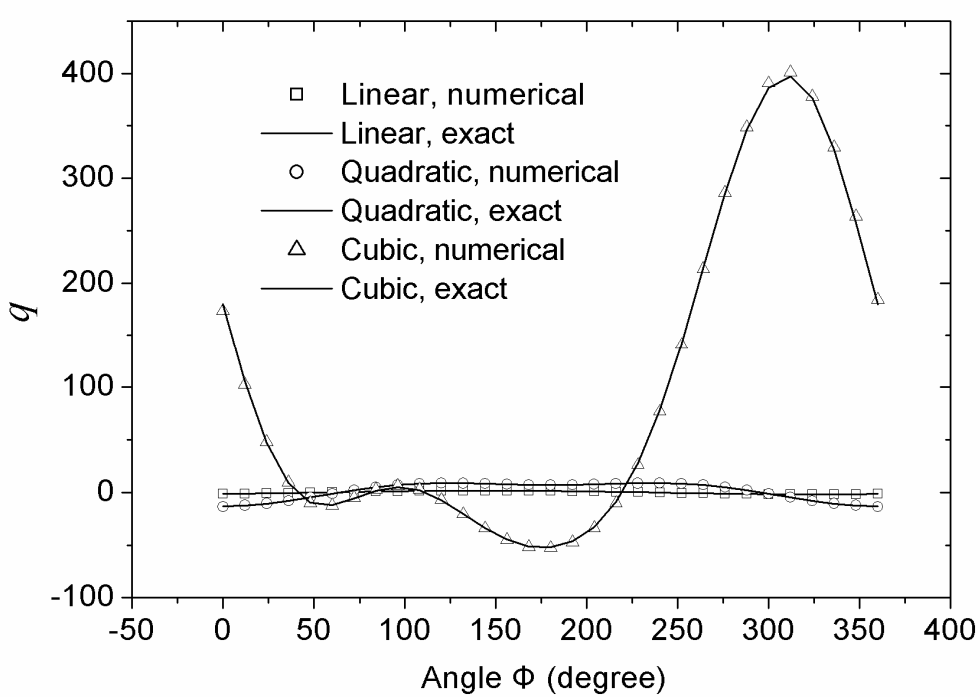
  Analytical solution: $(1)\ u = x + y + z$

  $(2)\ u = -2x^2 + y^2 + z^2$

  $(3)\ u = x^3 + y^3 + z^3 - 3yx^2 - 3xz^2 - 3zy^2$

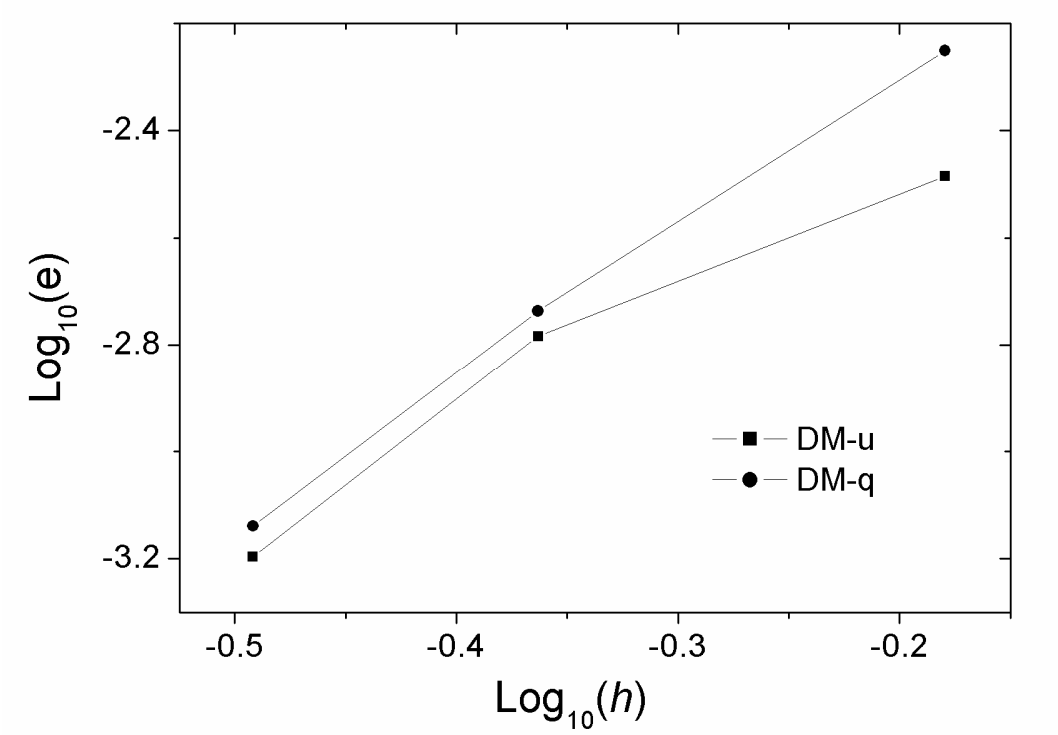
Input file contains 240 data, only!

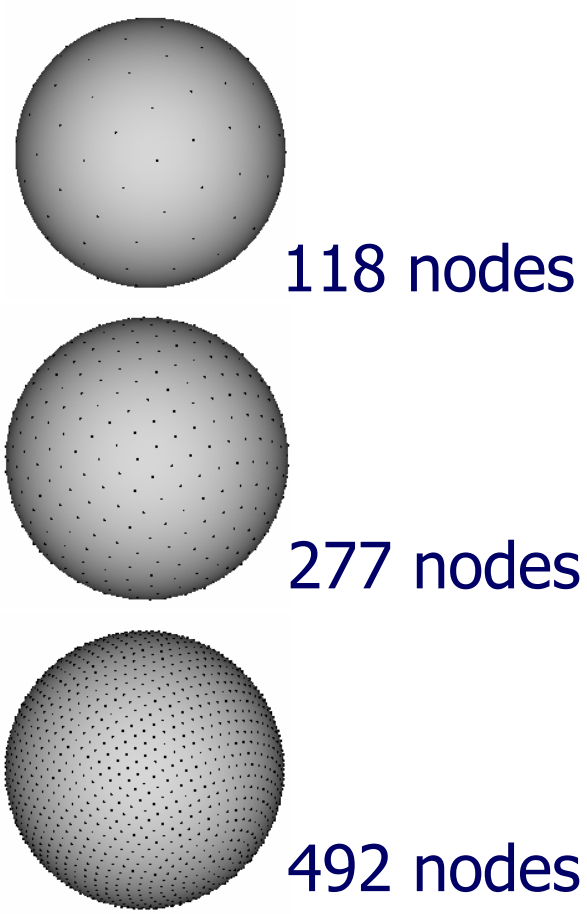




Numerical results from HdBNM (3D potential problem 2)

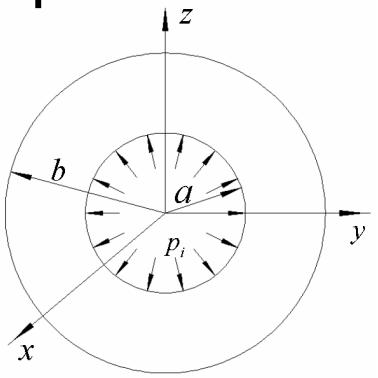
- Convergence study

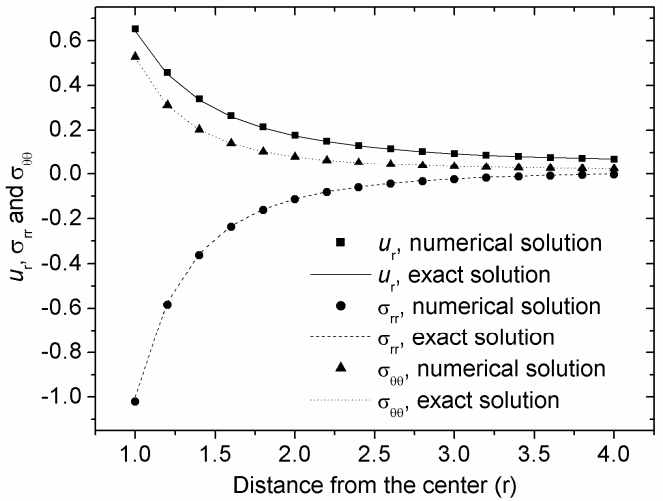




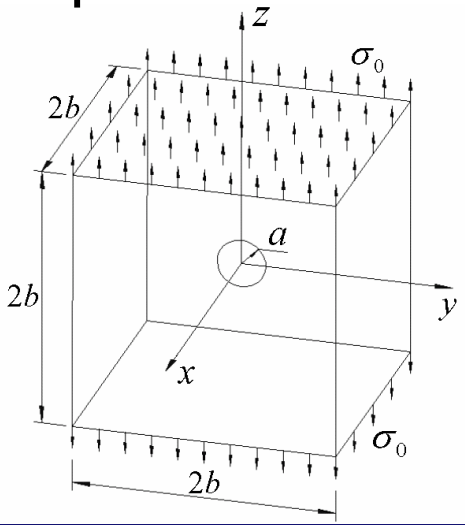
Numerical results from HdBNM (3D elasticity problem)

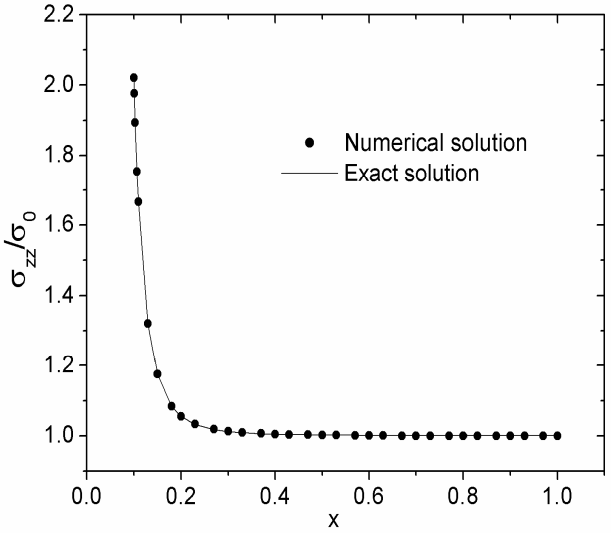
- 3D Lame problem





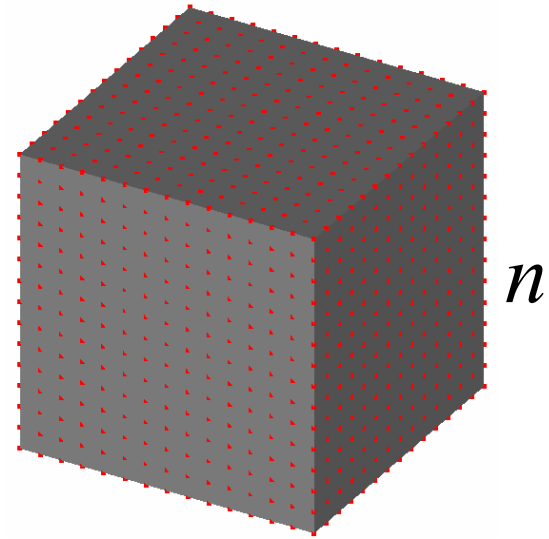
- 3D Kirsch problem





Advantages and disadvantages of HdBNM

- Advantages
  - Reduction of the dimensionality as BEM
  - No need for calculating derivates of MLS shape functions
  - Truly meshless
  - Ability to make direct use of Brep data of CAD geometry
- Disadvantage
  - Dense and unsymmetrical coefficient matrices



	Domain type methods (FEM, EFG, MLPG)	Boundary type methods (BEM, HdBNM)	Boundary type with linear complexity
Total degrees of freedom	$O(n^3)$	$O(n^2)$	$O(n^2)$
Memory requirement	$O(n^3)$	$O(n^4)$	$O(n^2)$
Time complexity	$O(n^3)$	$O(n^4)$	$O(n^2)$

Accelerating the HdBNM with Fast Multipole Method (FMM)

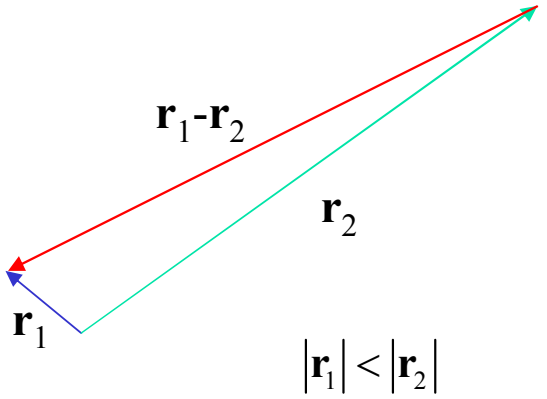
Reducing computational complexity to *O*(*N*)

Fast multipole method

➢First addition theorem

Let $\mathbf{r}_1$ and $\mathbf{r}_2$ be two vectors with spherical coordinates (r_1, α_1, β_1) and (r_2, α_2, β_2), respectively. It follows

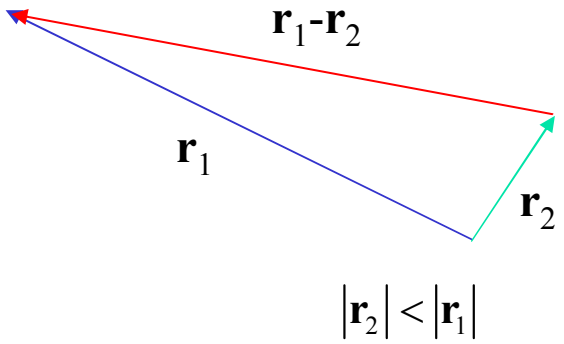
$$\frac{1}{|\mathbf{r}_1 - \mathbf{r}_2|} = \begin{cases} \sum_{n=0}^{\infty} \sum_{m=-n}^{n} R_n^m(\mathbf{r}_1) \overline{S_n^m(\mathbf{r}_2)}, & |\mathbf{r}_1| < |\mathbf{r}_2| \\ \sum_{n=0}^{\infty} \sum_{m=-n}^{n} R_n^m(\mathbf{r}_2) \overline{S_n^m(\mathbf{r}_1)}, & |\mathbf{r}_1| > |\mathbf{r}_2| \end{cases}$$



where

$$R_n^m(\mathbf{r}) = \frac{1}{(n+m)!} P_n^m(\cos\alpha) e^{im\beta} r^n$$

$$S_n^m(\mathbf{r}) = (n-m)! P_n^m(\cos\alpha) e^{im\beta} \frac{1}{r^{n+1}}$$

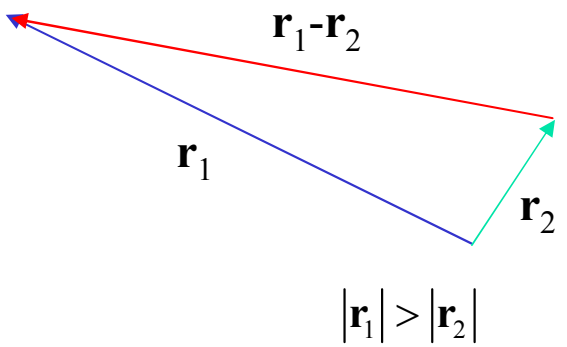


Fast multipole method (2)

➢Second addition theorem

Let $\mathbf{r}_1$ and $\mathbf{r}_2$ be two vectors such that $|\mathbf{r}_1| > |\mathbf{r}_2|$, then

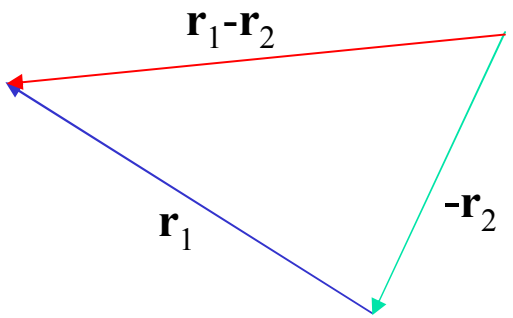
$$S_n^m(\mathbf{r}_1 - \mathbf{r}_2) = \sum_{n'=0}^{\infty} \sum_{m'=-n'}^{n'} \overline{R_{n'}^{m'}(\mathbf{r}_2)} S_{n+n'}^{m+m'}(\mathbf{r}_1)$$



➢Third addition theorem

Let $\mathbf{r}_1$ and $\mathbf{r}_2$ be two arbitrary vectors, then

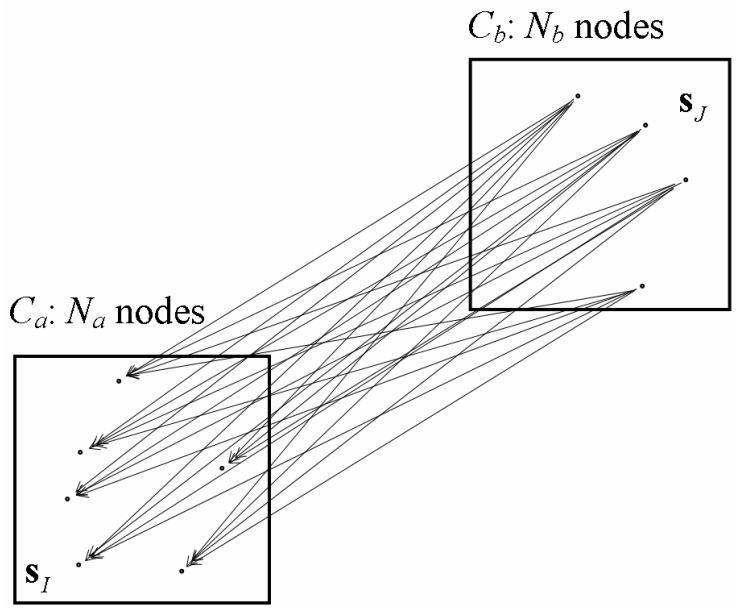
$$R_n^m(\mathbf{r}_1 - \mathbf{r}_2) = \sum_{n'=0}^{n} \sum_{m'=-n'}^{n'} R_{n'}^{m'}(\mathbf{-r}_2) R_{n-n'}^{m-m'}(\mathbf{r}_1)$$



Fast multipole method (3)

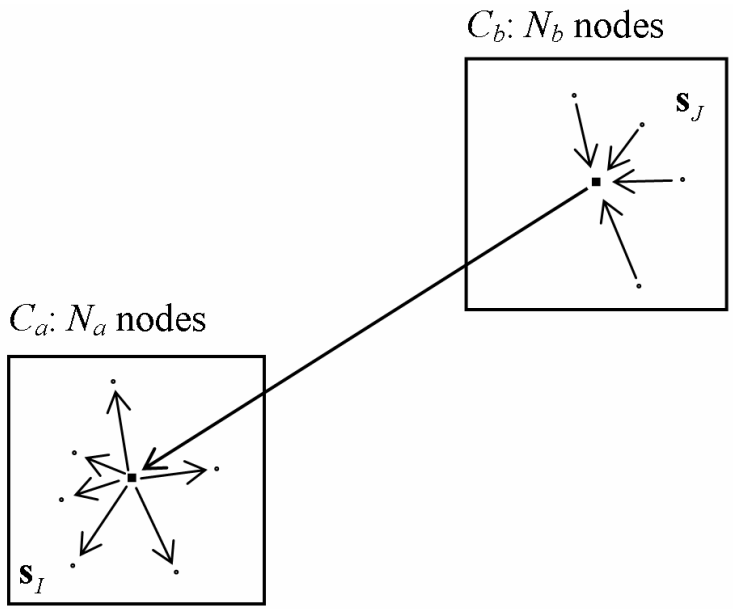
- **Ideas of FMM**

Node-node interactions



Complexity $O(N_a N_b)$

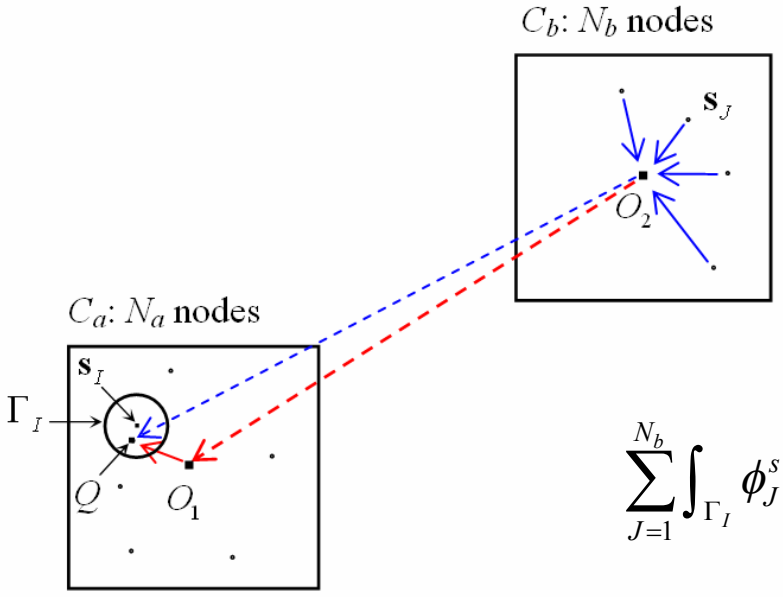
Cell-cell interactions



Complexity $O(N_a+N_b)$

Fast multipole method (4)

- **Multipole expansion**



$$\phi_J^s = \frac{1}{4\pi\kappa}\frac{1}{r(Q,\mathbf{s}_J)} = \frac{1}{4\pi\kappa}\sum_{n=0}^{\infty}\sum_{m=-n}^{n}\overline{S_n^m}(\overrightarrow{O_2Q})\,R_n^m(\overrightarrow{O_2\mathbf{s}_J})$$

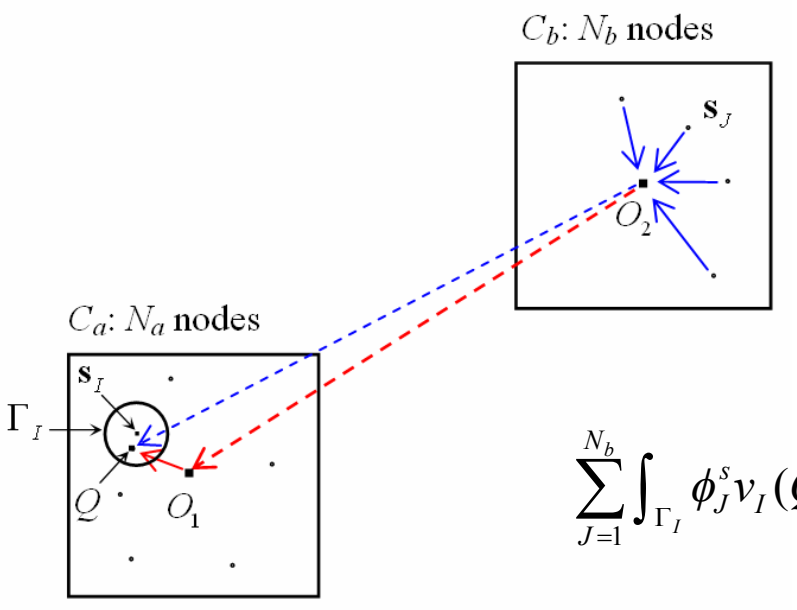
for $\quad \left|\overrightarrow{O_2Q}\right| > \left|\overrightarrow{O_2\mathbf{s}_J}\right|$

$$\sum_{J=1}^{N_b}\int_{\Gamma_I}\phi_J^s v_I(Q)x_J' d\Gamma = \sum_{n=0}^{\infty}\sum_{m=-n}^{n}\int_{\Gamma_I}\frac{1}{4\pi\kappa}\overline{S_n^m}(\overrightarrow{O_2Q})\,v_I(Q)d\Gamma\; M_n^m(O_2)$$

where $\quad M_n^m(O_2) = \sum_{J=1}^{N_b} R_n^m(\overrightarrow{O_2\mathbf{s}_J})x_J'$

Fast multipole method (5)

- **Local expansion**



$$\overline{S_n^m}(\overrightarrow{O_2Q}) = \sum_{n'=0}^{\infty}\sum_{m'=-n'}^{n'} (-1)^{n'} R_{n'}^{m'}(\overrightarrow{O_1Q})\,\overline{S_{n+n'}^{m+m'}}(\overrightarrow{O_1O_2})$$

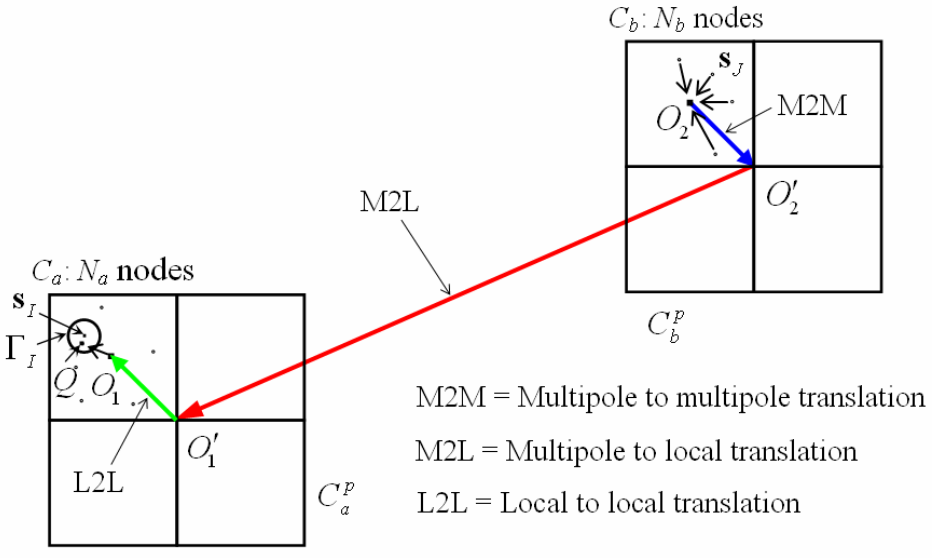
for $\left|\overrightarrow{O_1O_2}\right| > \left|\overrightarrow{O_1Q}\right|$

$$\sum_{J=1}^{N_b}\int_{\Gamma_I}\phi_J^s v_I(Q)x_J' d\Gamma = \sum_{n'=0}^{\infty}\sum_{m'=-n'}^{n'}\int_{\Gamma_I}\frac{1}{4\pi\kappa}R_{n'}^{m'}(\overrightarrow{O_1Q})\,v_I(Q)d\Gamma\; L_{n'}^{m'}(O_1)$$

where $$L_{n'}^{m'}(O_1) = \sum_{n=0}^{\infty}\sum_{m=-n}^{n}(-1)^{n'}\overline{S_{n+n'}^{m+m'}}(\overrightarrow{O_1O_2})M_n^m(Q_2)$$

Fast multipole method (6)

➢ Translation operators



$$M_{n'}^{m'}(Q_2') = \sum_{n=0}^{\infty}\sum_{m=-n}^{n} R_n^m(\overrightarrow{O_2'O_2})M_{n-n'}^{m-m'}(Q_2)$$

Multipole to multipole translation

$$L_n^m(O_1') = \sum_{n'=0}^{\infty}\sum_{m'=-n'}^{n'} (-1)^n \overline{S_{n+n'}^{m+m'}}(\overrightarrow{O_1'O_2'})M_{n'}^{m'}(Q_2')$$

Multipole to local translation

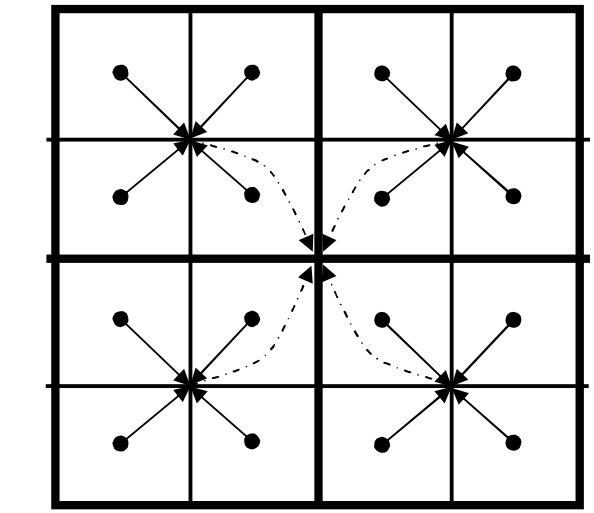
$$L_{n'}^{m'}(O_1) = \sum_{n=0}^{\infty}\sum_{m=-n}^{n} R_{n-n'}^{m-m'}(\overrightarrow{O_1'O_1})L_n^m(Q_1')$$

Local to local translation

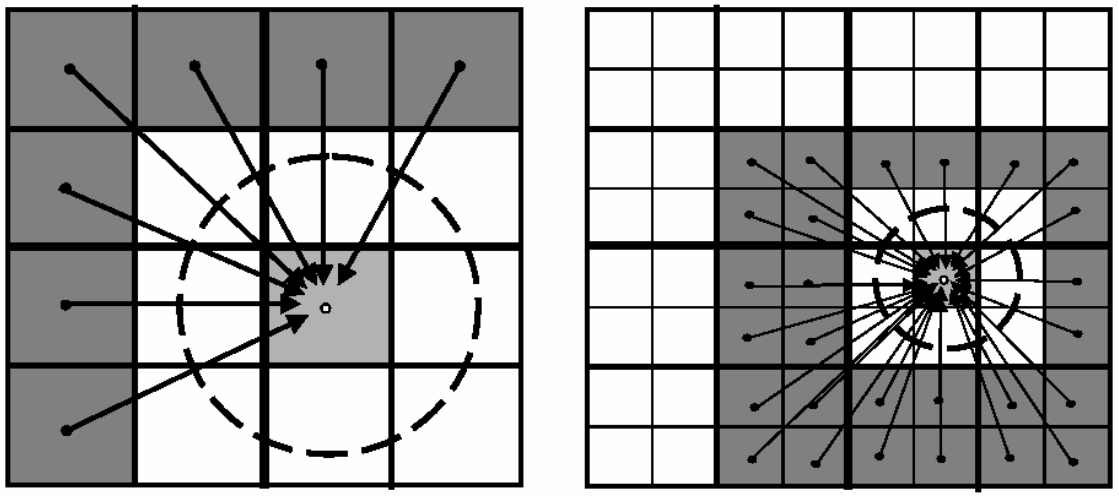
Fast multipole method (7)

- **Recursive algorithm**

Upward pass



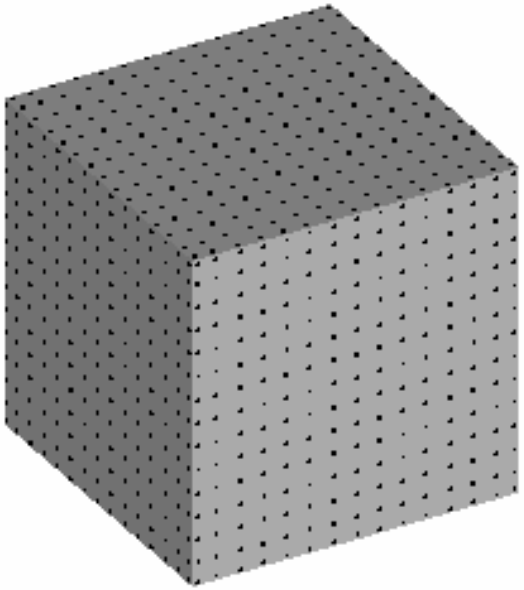
Downward pass



Multipole moments are accumulated from leaves to the root (***Upward pass***); and local moments are distributed from the root to the leaves (***Downward pass***). This is accomplished at a linear complexity.

Results from FM-HdBNM

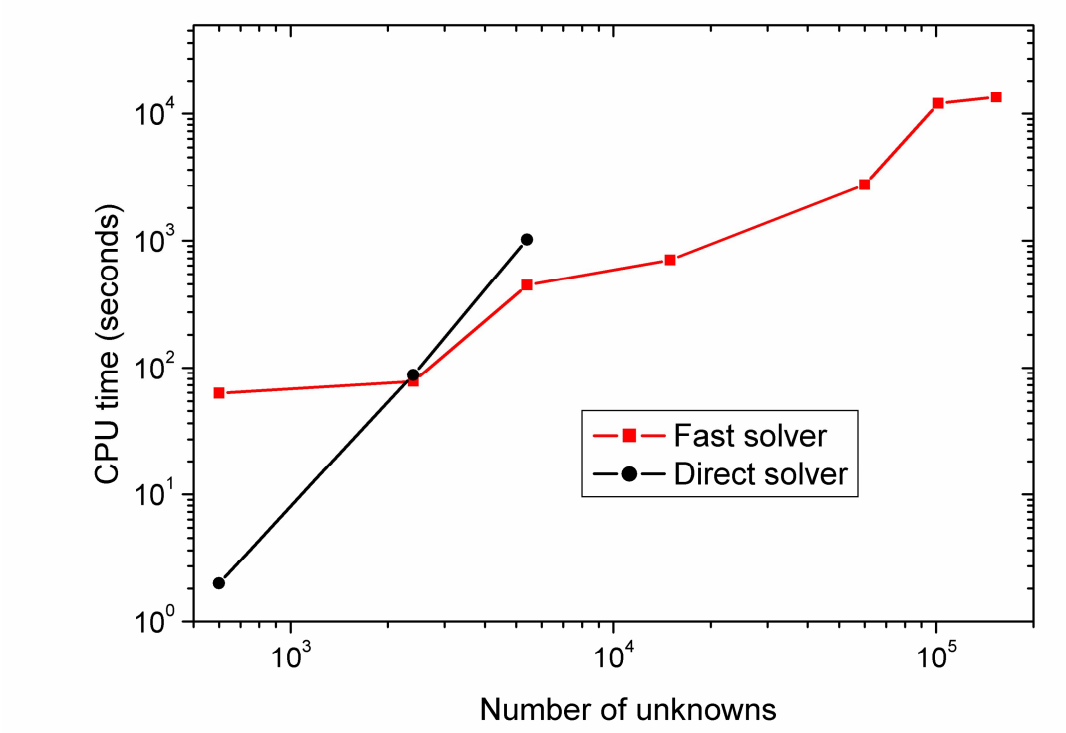
- **Results for a cube**



Max number of unknowns

Direct computation: 5400

FMM computation: 153600

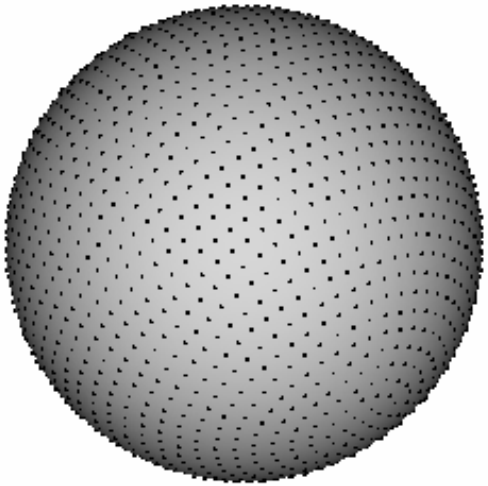


CPU time vs number of unknowns

Performed on a desktop computer with an Intel(R) Pentium(R) 4 CPU (1.99GHz)

Results from FM-HdBNM (2)

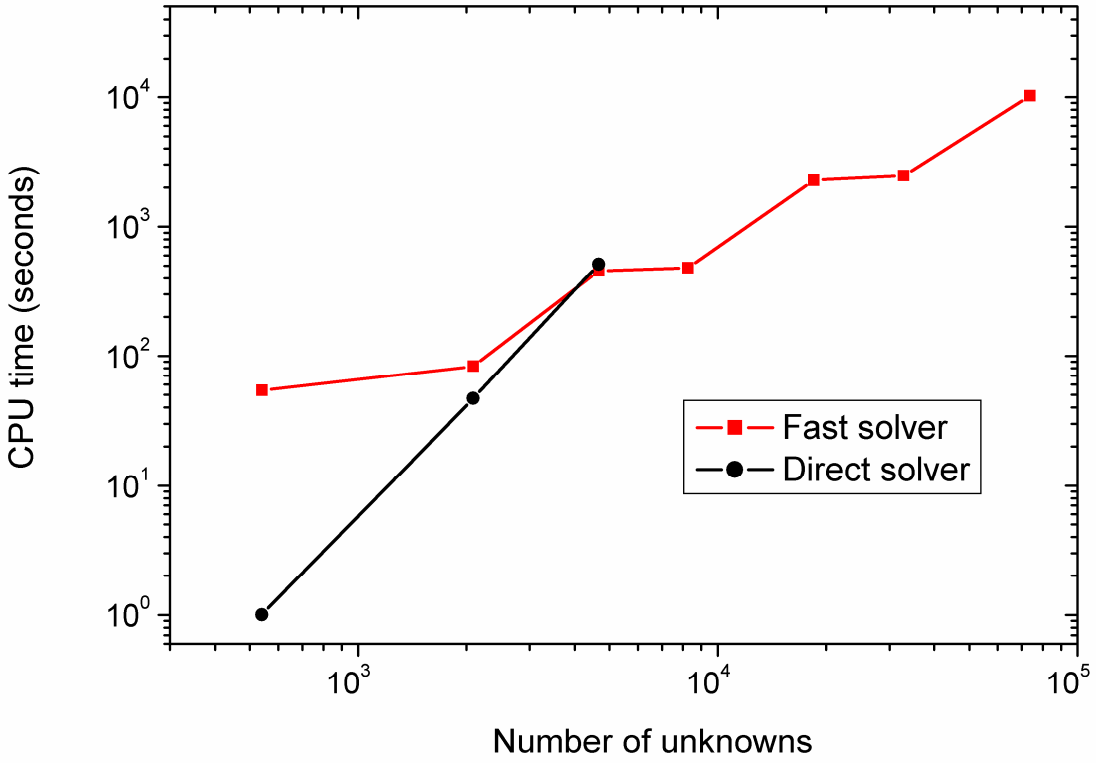
- **Results for a sphere**



Max number of unknowns

Direct computation: 4664

FMM computation: 73664

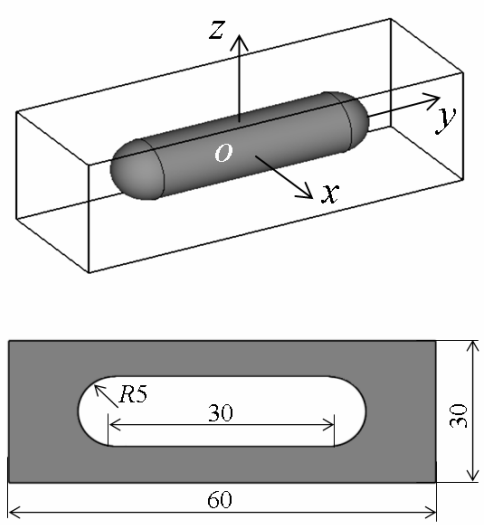


CPU time vs number of unknowns

Performed on a desktop computer with an Intel(R) Pentium(R) 4 CPU (1.99GHz)

Results from FM-HdBNM (3)

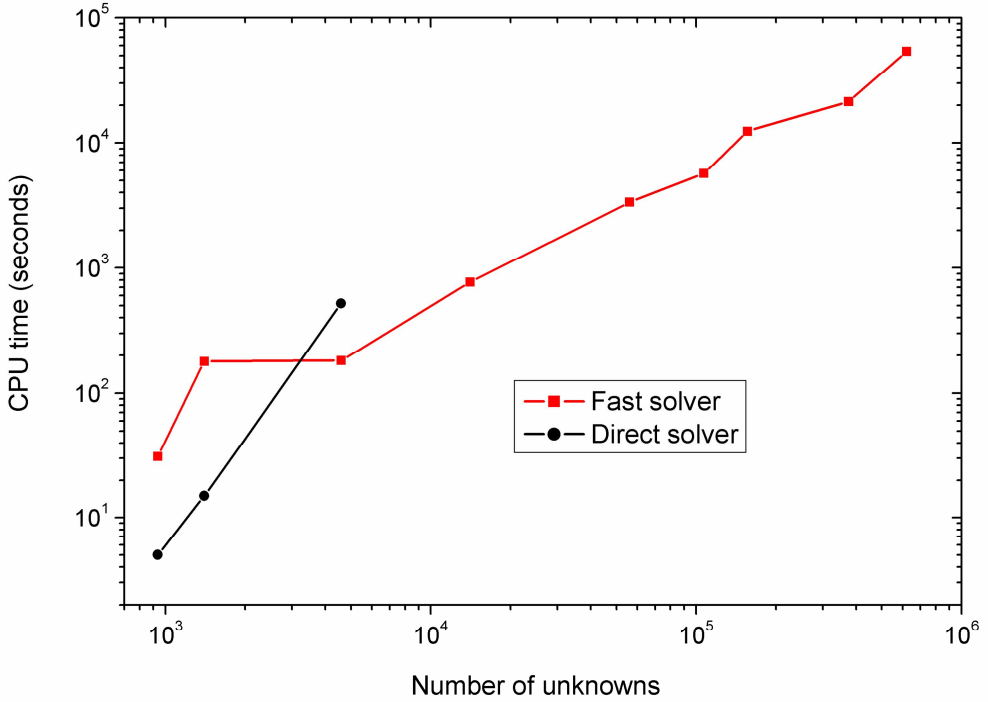
- **A cuboid including a cavity**



Max number of unknowns

Direct computation: 4598

FMM computation: 620708



CPU time vs number of unknowns

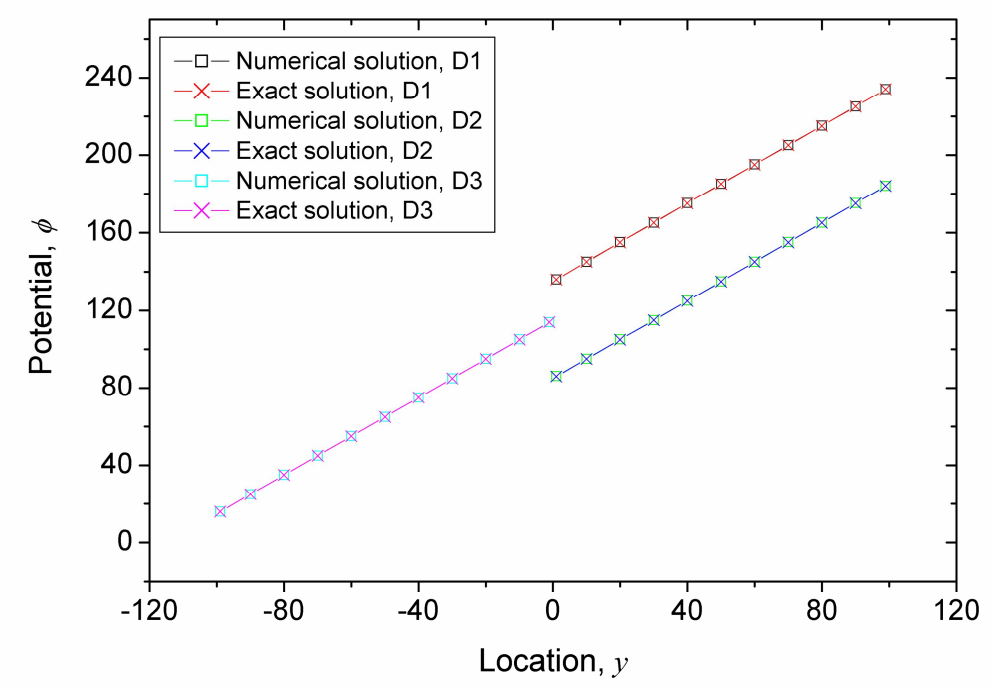
Performed on a desktop computer with an Intel(R) Pentium(R) CPU (1.99GHz)

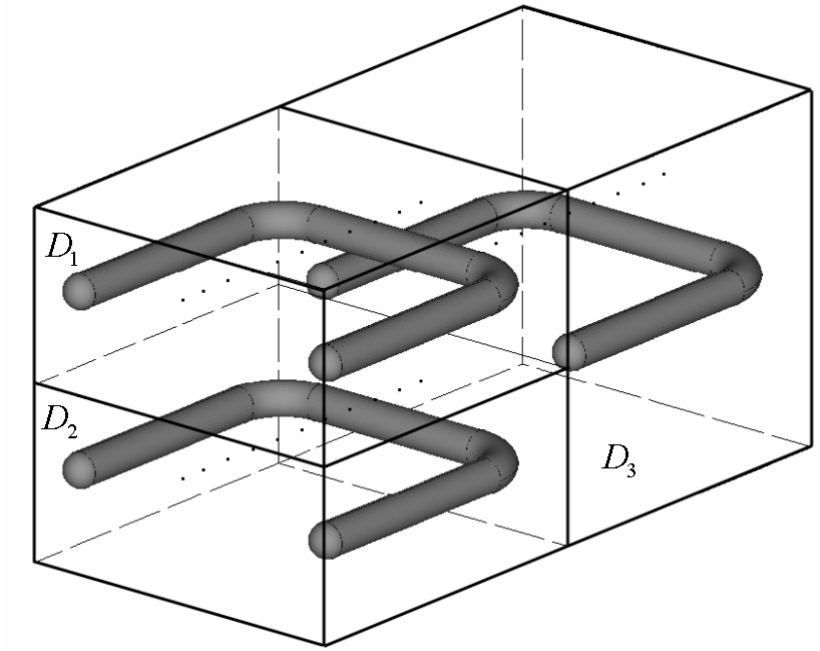
Results from FM-HdBNM (4)

- **Three boxes with inclusions**

Analytical solution: $\phi = x + y + z$

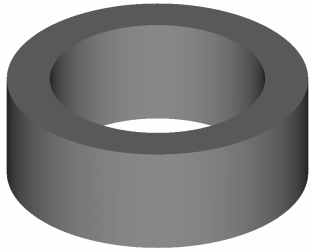
Total number of unknowns: 57506





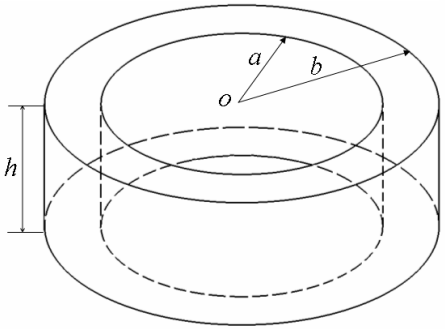
Results from FM-HdBNM (5)

- **A cylinder**

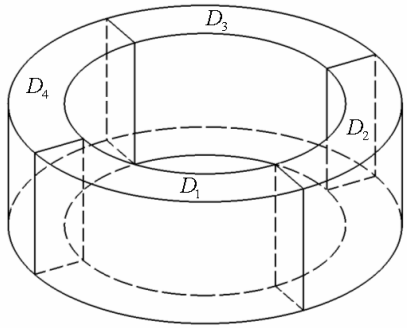


Analytical solution:

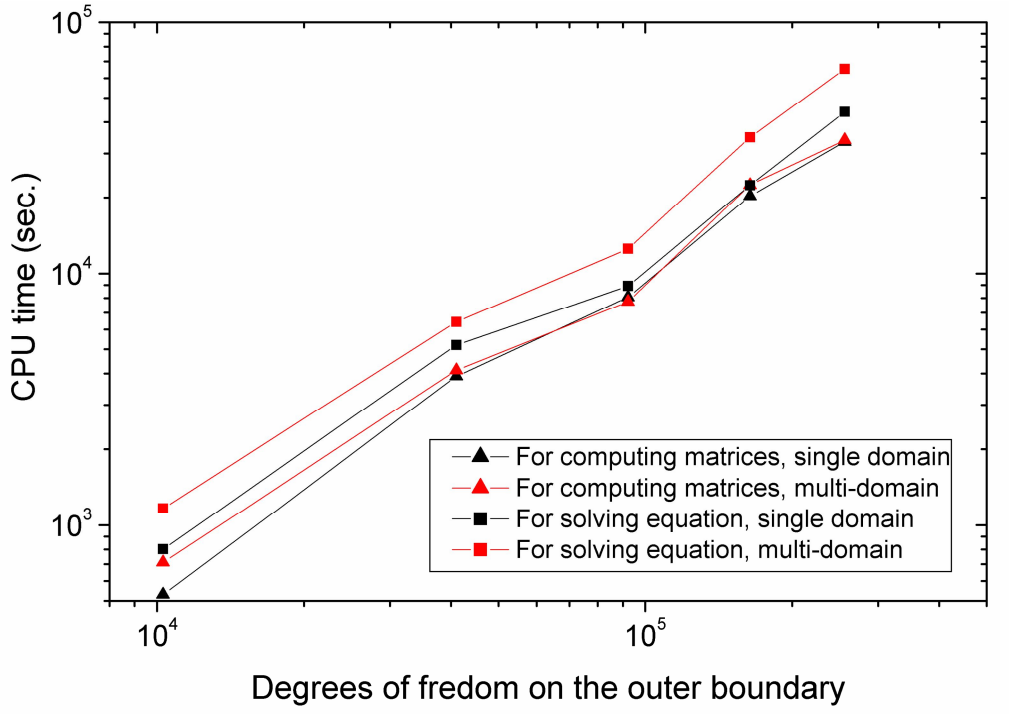
$$\phi = x^3 + y^3 + z^3 - 3yx^2 - 3xz^2 - 3zy^2$$



Single domain model



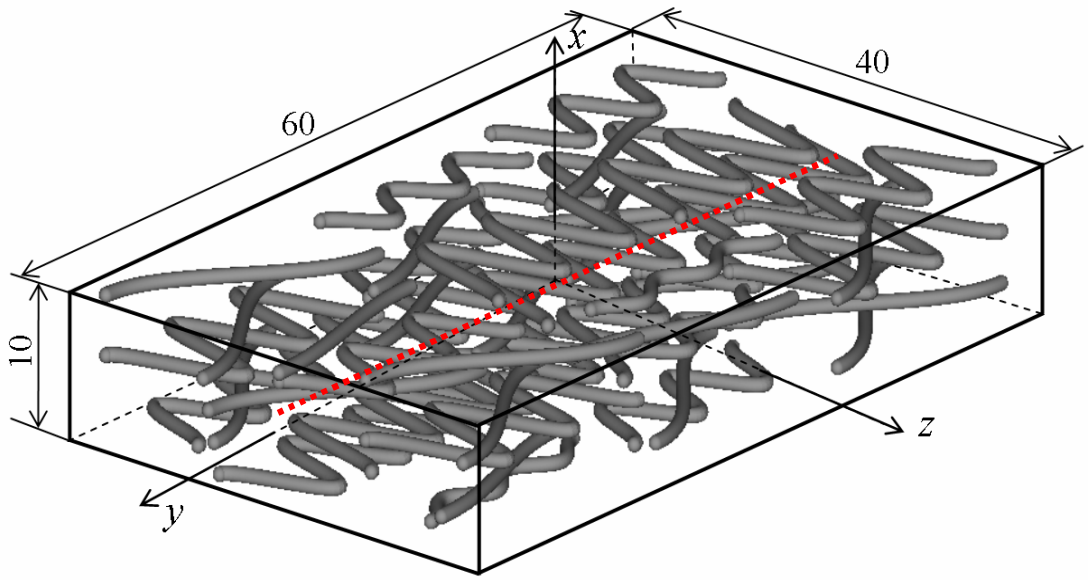
Four sub-domains model

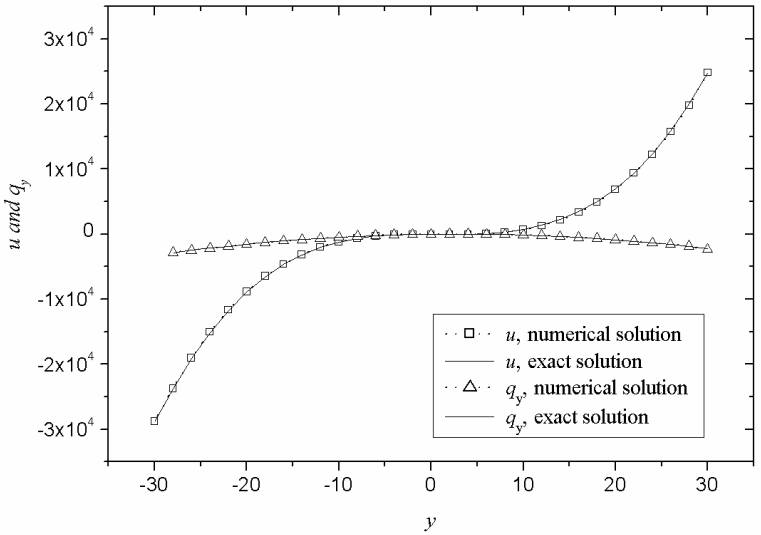


Timing results for the thick cylinder problem

Results from FM-HdBNM (6)

- **Results for a cuboid including many cavities in curved tube shape**





Number of nodes	108960	137370	159680	192500	255180
Relative error of nodal values	6.42%	3.81%	3.35%	2.33%	1.91%
CPU time (seconds)	25386	37840	43058	55670	67725

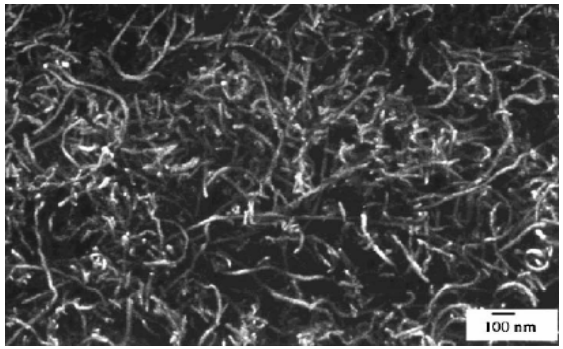
$$u = x^3 + y^3 + z^3 - 3yx^2 - 3xz^2 - 3zy^2$$

$$e = \frac{1}{|q|_{\max}} \sqrt{\frac{1}{N} \sum_{i=1}^{N} (q_i^{(e)} - q_i^{(n)})^2}$$

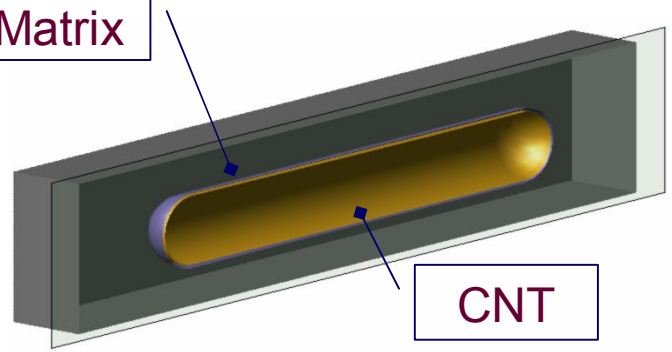
Application of fast HdBNM to thermal analysis of CNT composites

Thermal analysis of CNT composites

➢CNT based composites

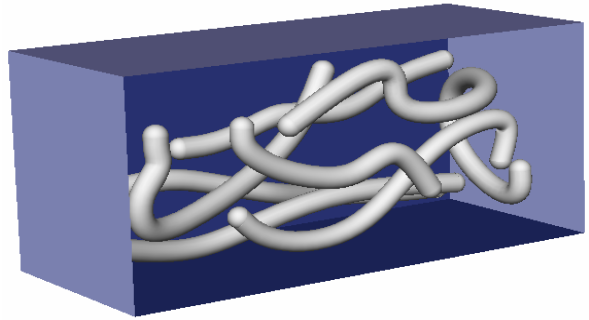


Nanotube-reinforced polymers

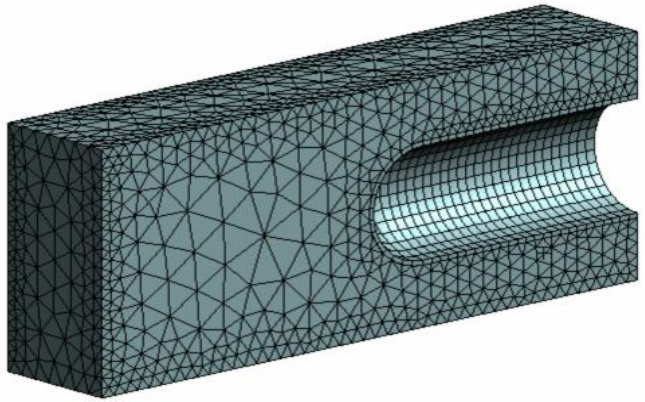


Representative Volume Element (RVE)

➢Numerical simulation model

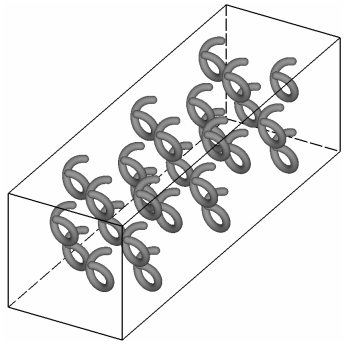


RVE including curved CNTs

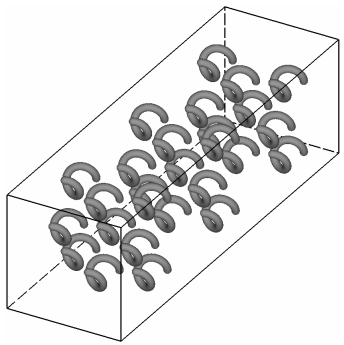


FEM/BEM Mesh for a quarter of an RVE

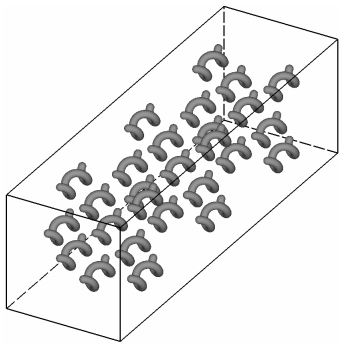
Thermal analysis of CNT composites (2)



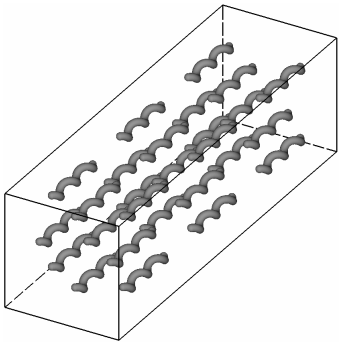
$\kappa = 0.4362$



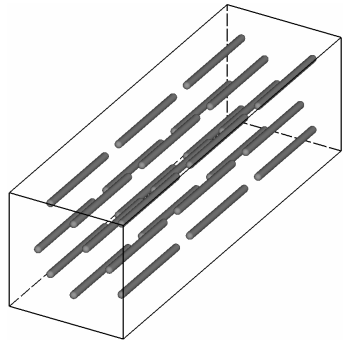
$\kappa = 0.6975$



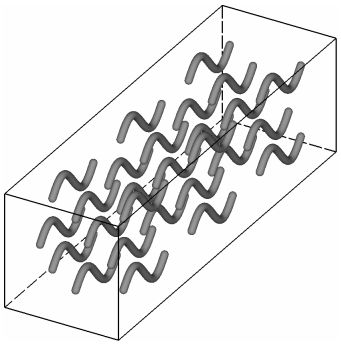
$\kappa = 0.8703$



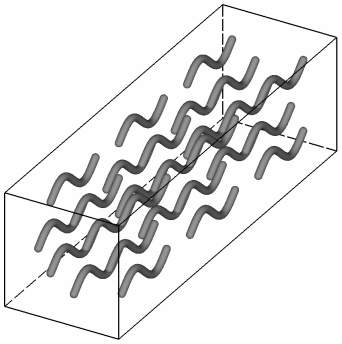
$\kappa = 0.9445$



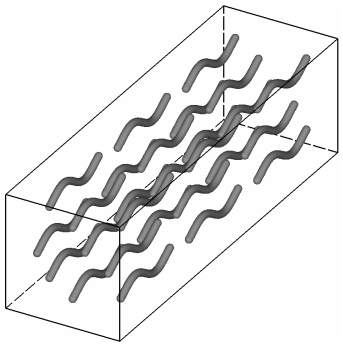
$\kappa = 0.9482$



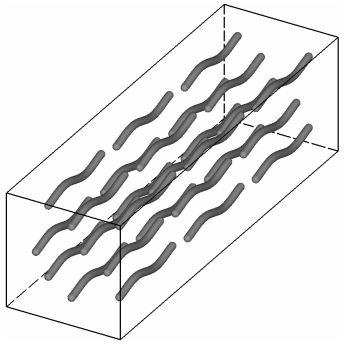
$\kappa = 0.6679$



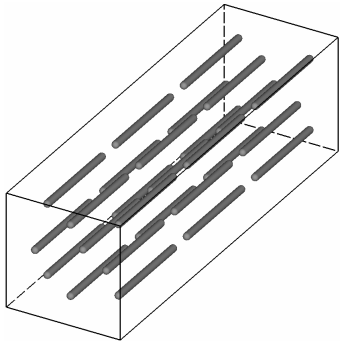
$\kappa = 0.7431$



$\kappa = 0.8257$

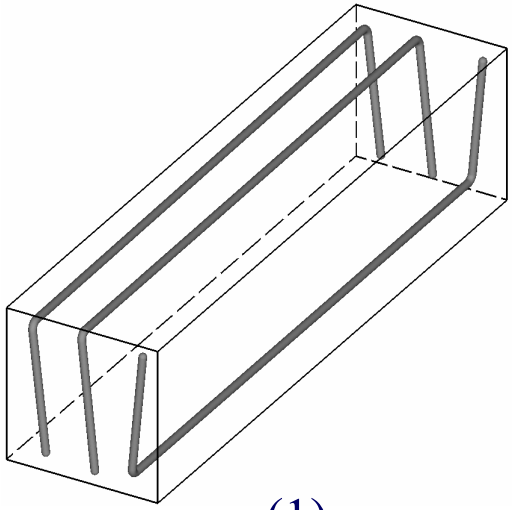


$\kappa = 0.9381$

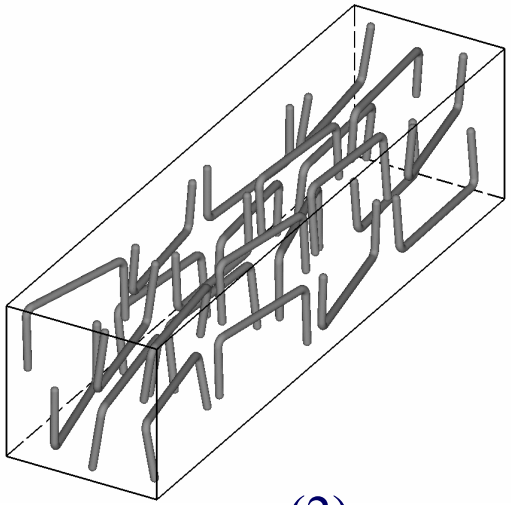


$\kappa = 0.9482$

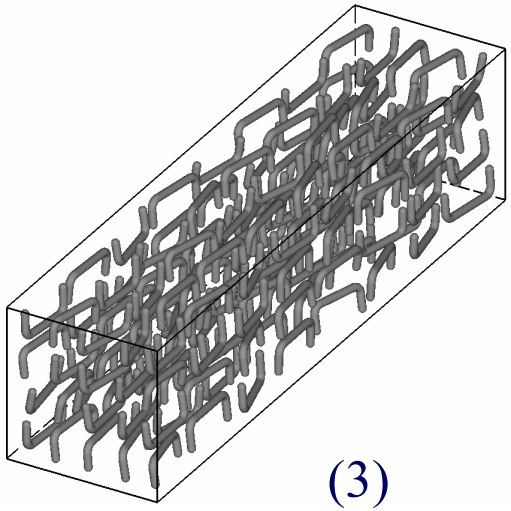
Thermal analysis of CNT composites (3)



(1)



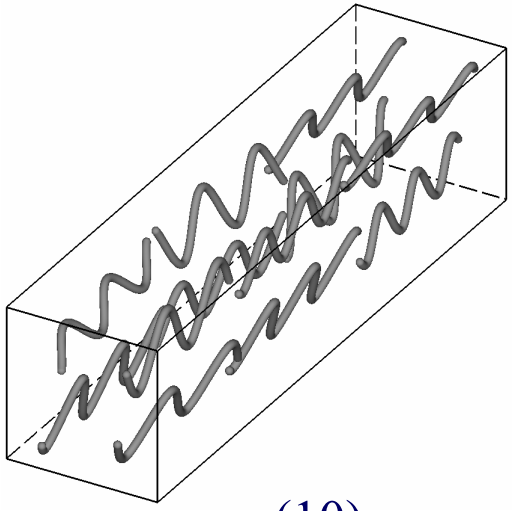
(2)



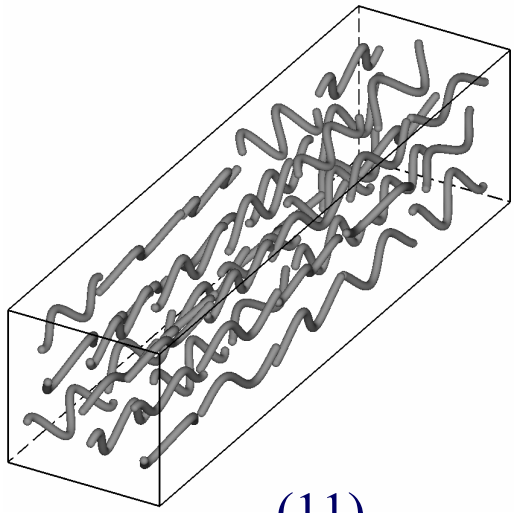
(3)

No.	*AVR Len*	*CNT Num*	*Fraction, v*	*k*	*k/v*
(1)	1117	3	0.59%	8.472	1436
(2)	312.3	24	1.09%	1.432	131.4
(3)	139.8	160	3.78%	1.356	35.87

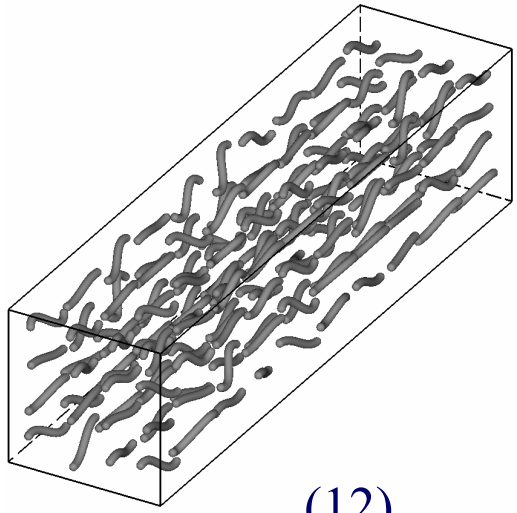
Thermal analysis of CNT composites (4)



(10)



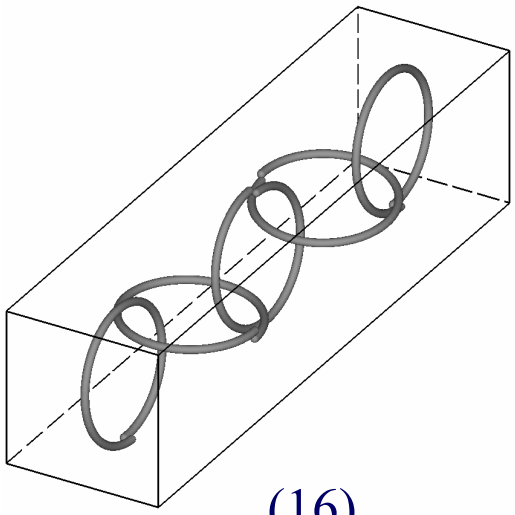
(11)



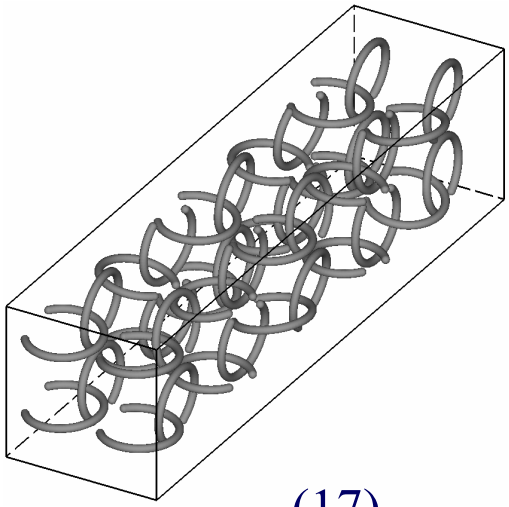
(12)

No.	*AVR Len*	*CNT Num*	*Fraction, v*	*k*	*k/v*
(10)	437.7	12	1.29%	1.186	91.91
(11)	205.1	45	2.23%	1.097	49.18
(12)	80.2	160	3.06%	0.760	24.83

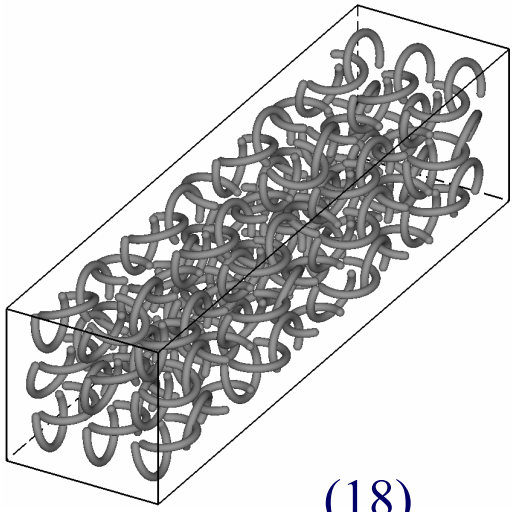
Thermal analysis of CNT composites (5)



(16)



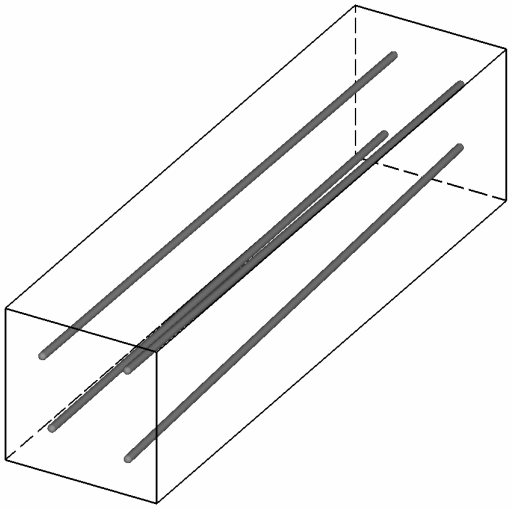
(17)



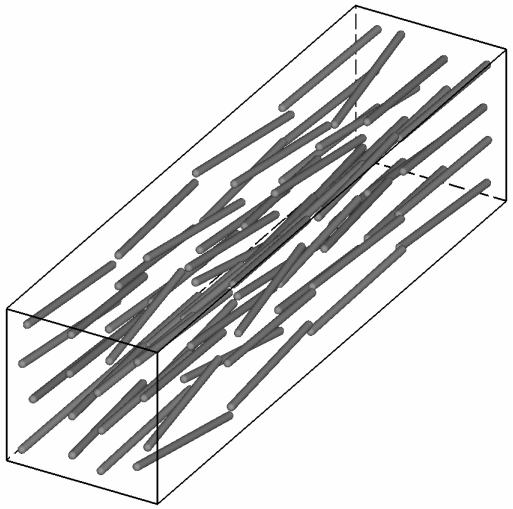
(18)

No.	*AVR Len*	*CNT Num*	*Fraction, v*	*k*	*k/v*
(16)	537.9	5	0.656%	0.796	121.3
(17)	239.2	40	2.31%	1.094	47.34
(18)	149.6	135	4.84%	1.709	35.31

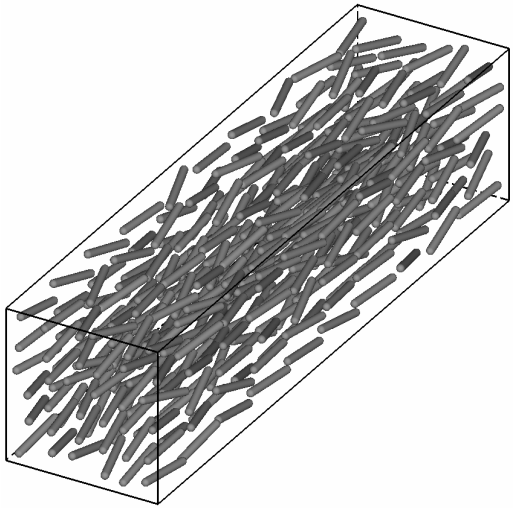
Thermal analysis of CNT composites (5)



(19)



(20)



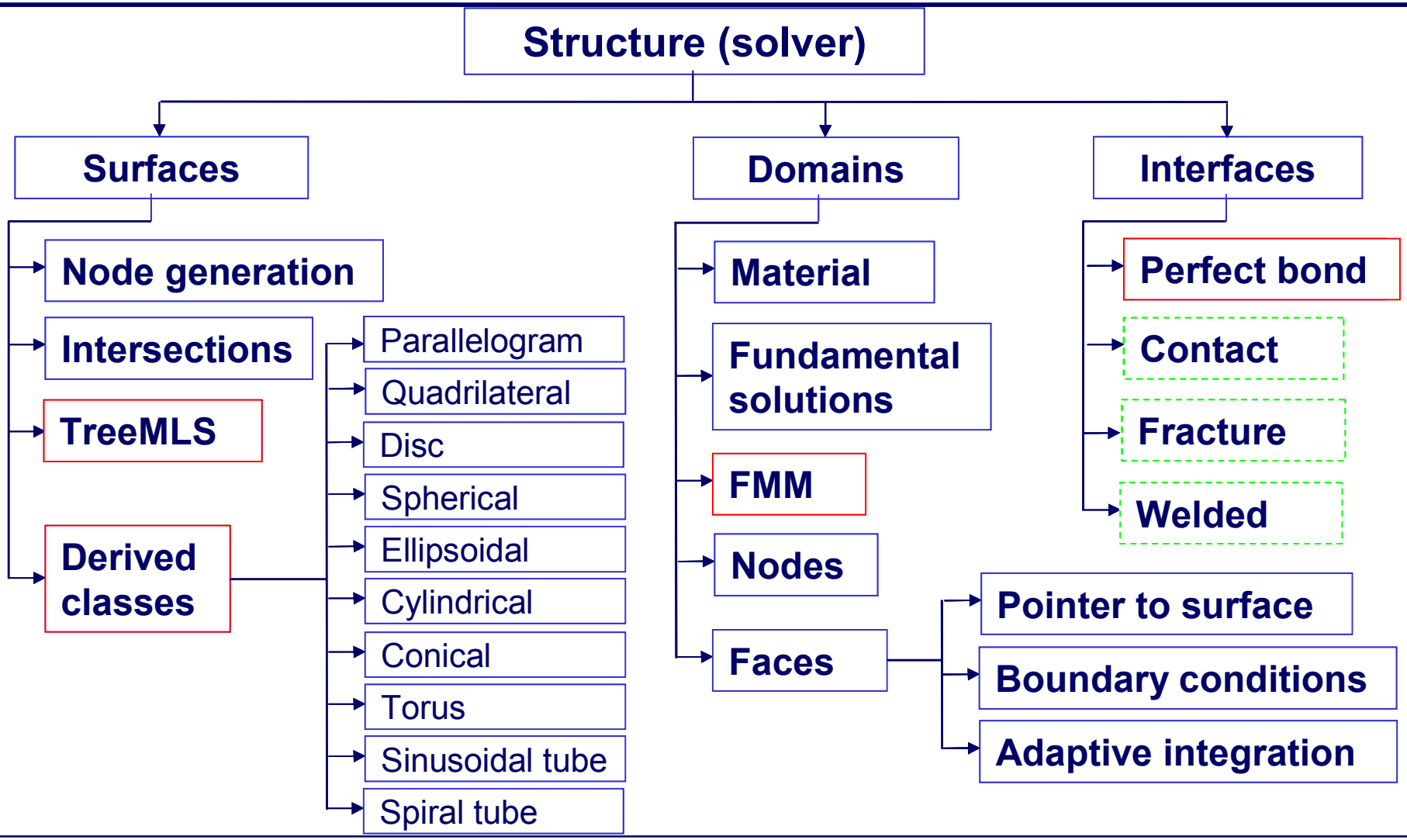
(21)

No.	*AVR Len*	*CNT Num*	*Fraction, v*	*k*	*k/v*
(19)	790.0	4	0.77%	2.990	387.4
(20)	194.7	64	2.99%	1.790	59.85
(21)	77.0	360	4.84%	1.204	19.02

Software development

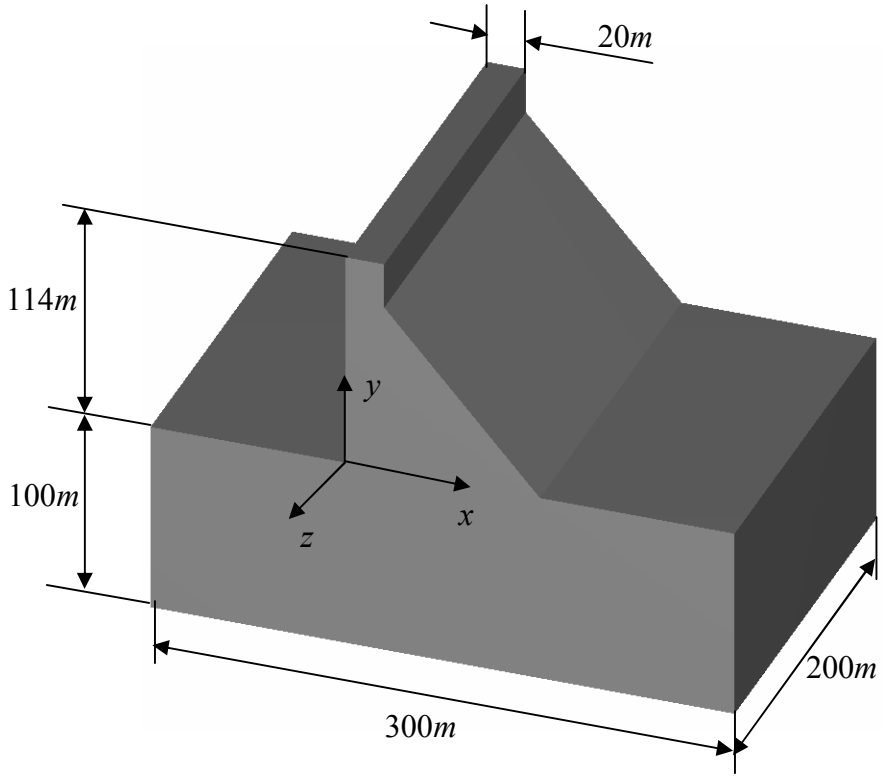
Automatic, accurate and efficient analysis of large-scale complex structures with arbitrary geometries and material composition

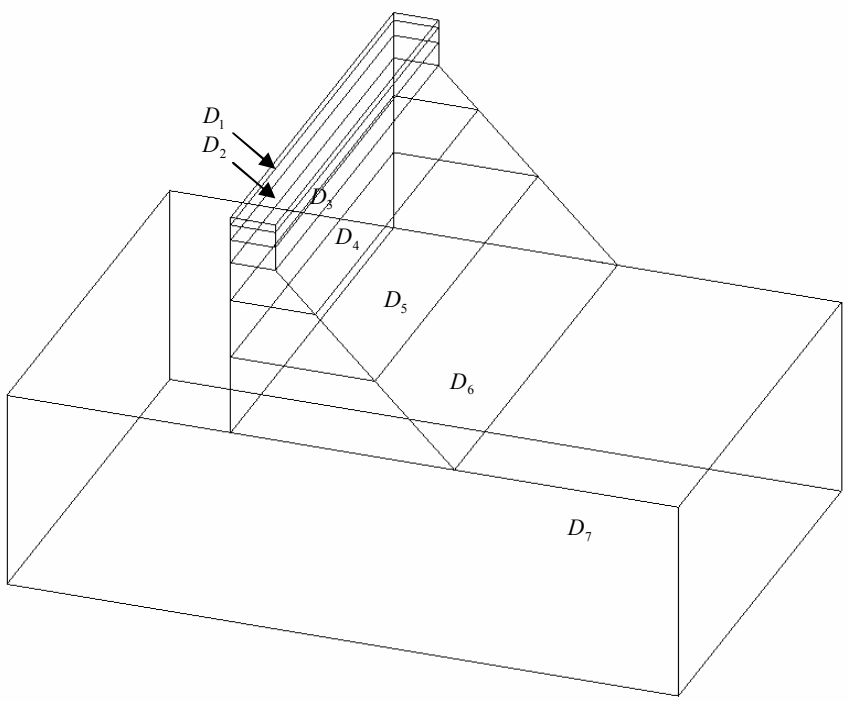
Software (Overall class organization)



Thermal analysis of concrete dam

➢ Geometric model



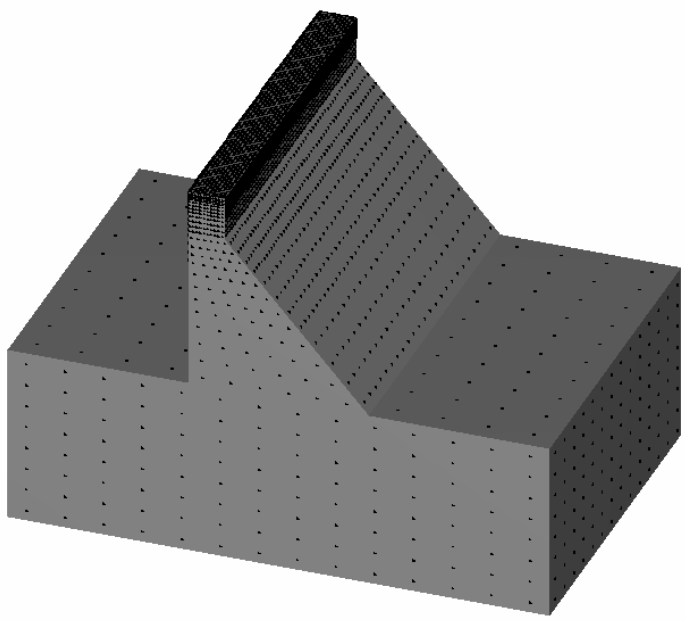


Thermal analysis of concrete dam

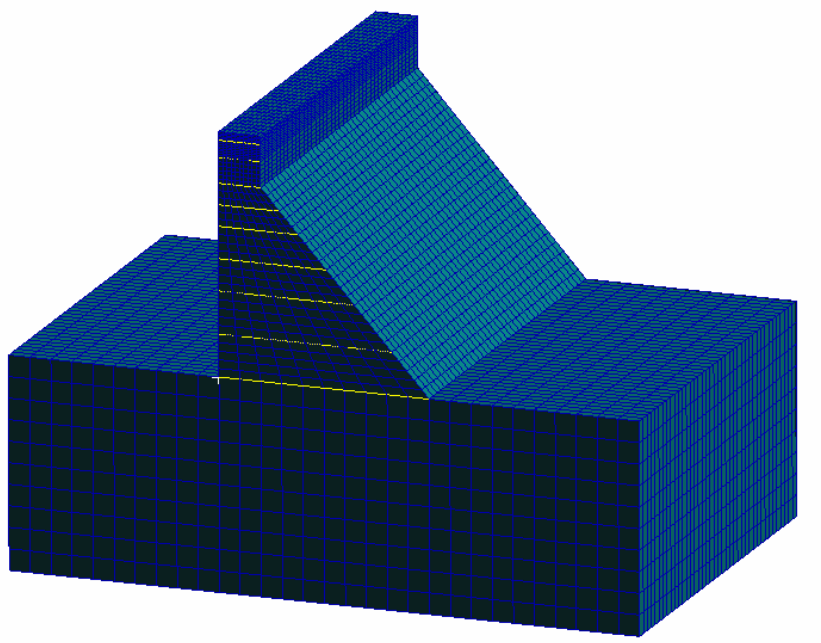
- **Discrete models**

27100, 15381, 8492 boundary Nodes used for HBNM analysis

112000、56000、29201 elements used for MSC/NASTRAN



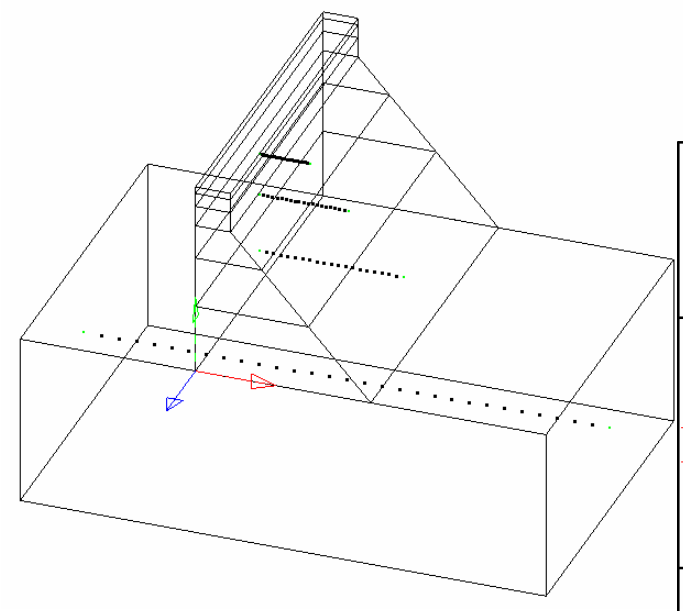
8492 boundary Nodes



29201 elements (FEM)

Thermal analysis of concrete dam

- **Homogeneous Case**



Evaluation Points

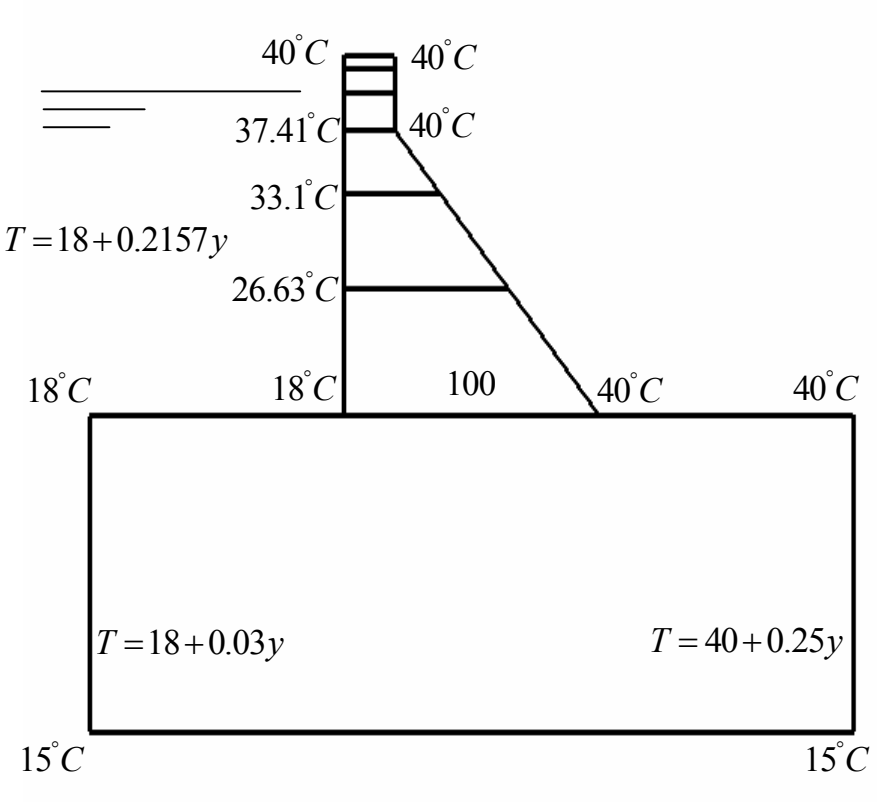
Analytical solution:

$$T = x^3 + y^3 + z^3 - 3yx^2 - 3xz^2 - 3zy^2$$

	Dofs	Time (sec.)	Relative Error（%）			
			Layer 4	Layer 5	Layer 6	Layer 7
HdBNM	27100	557	0.16	0.28	0.41	0.04
	15384	170	0.58	0.67	0.50	0.04
	8492	56	1.02	1.58	1.80	0.07
FEM	121023	265	0.32	0.37	0.45	0.45
	69679	82	0.33	0.38	0.70	0.46
	33386	25	0.65	0.66	1.46	0.90

Thermal analysis of concrete dam

➢ Heterogeneous Case



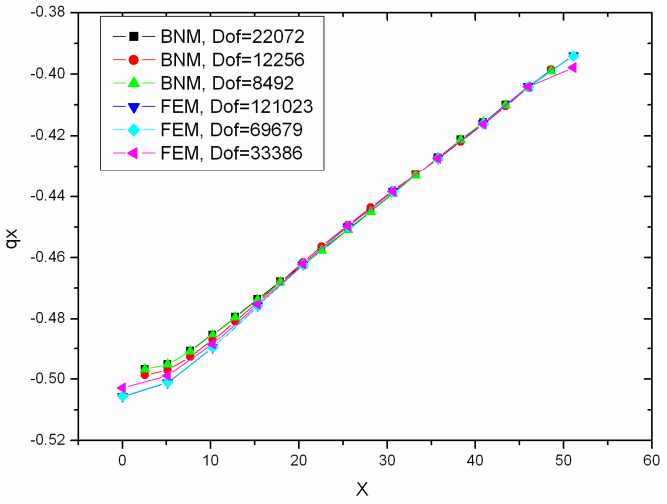
Boundary Condition

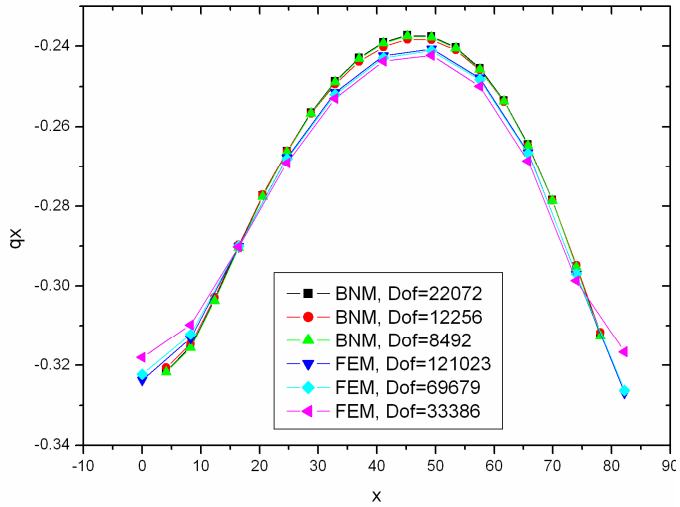
Layer	λ (W/m°C)
1	2.575
2	2.475
3	2.375
4	2.325
5	2.275
6	1.275
7	2.4378

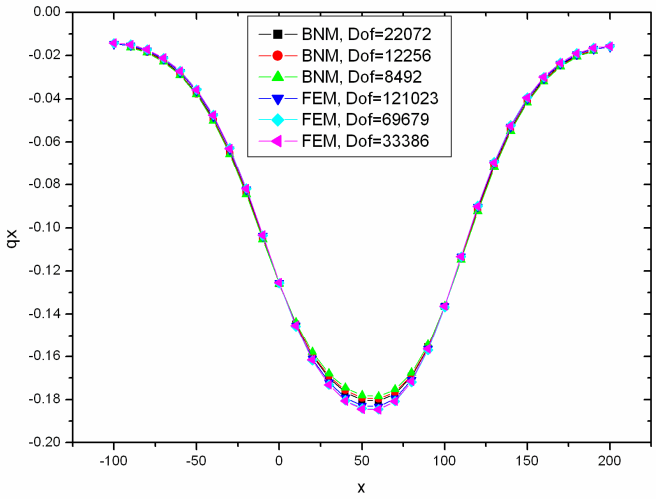
Materials for different layers

Thermal analysis of concrete dam

- **Results of q_x at evaluation points**

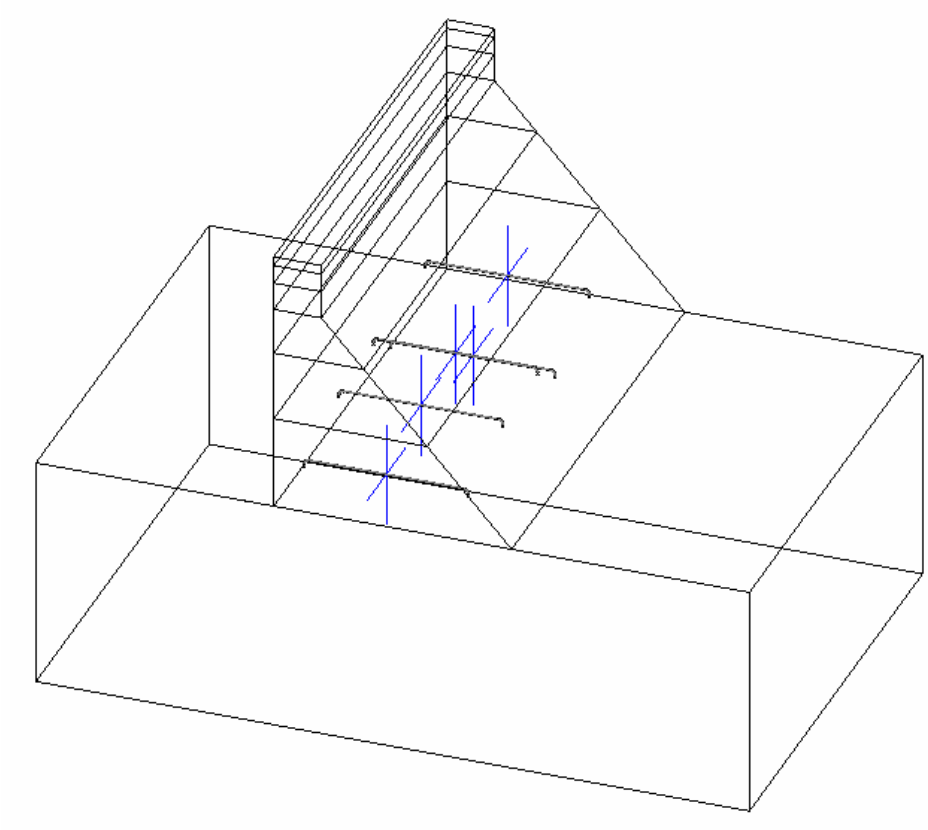




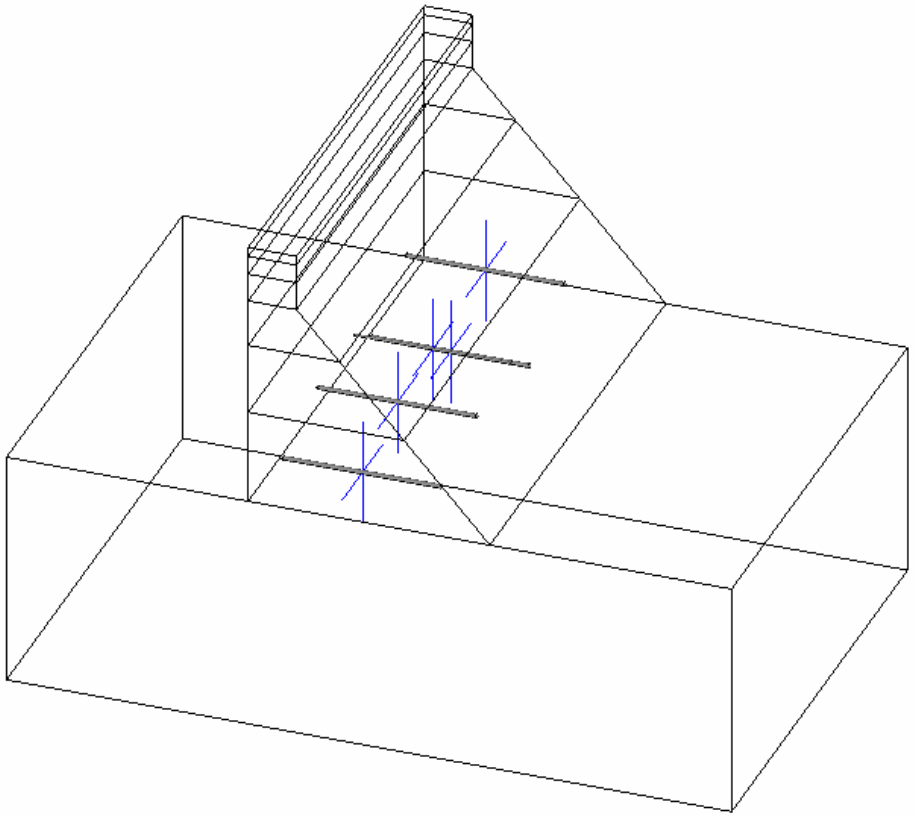


Thermal analysis of concrete dam

- **Cases with rebars and water pipes embeded**

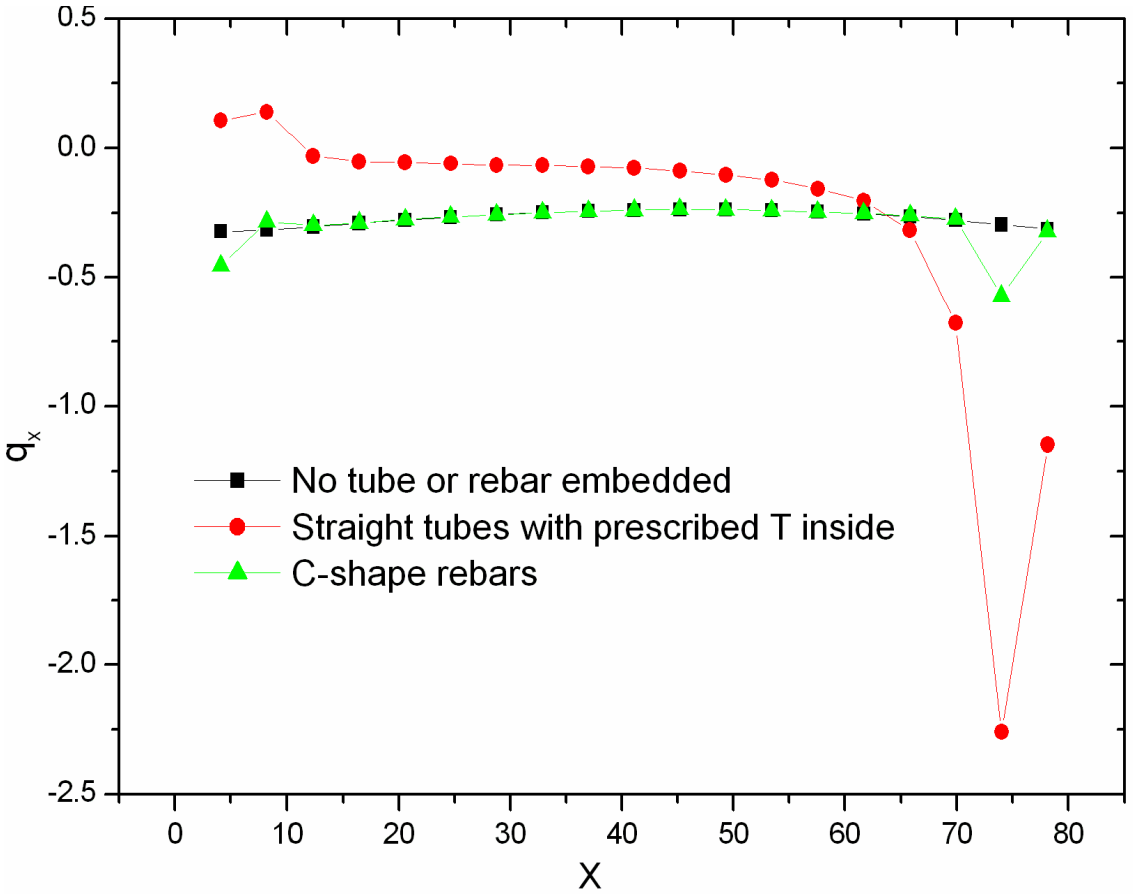


Five embeded rebars



Five embeded water pipes

Thermal analysis of concrete dam



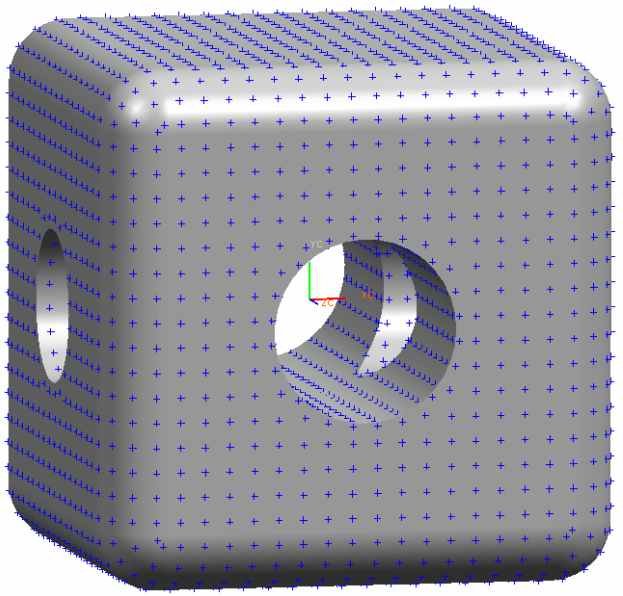
qx along a line segment near the rebar and pipe

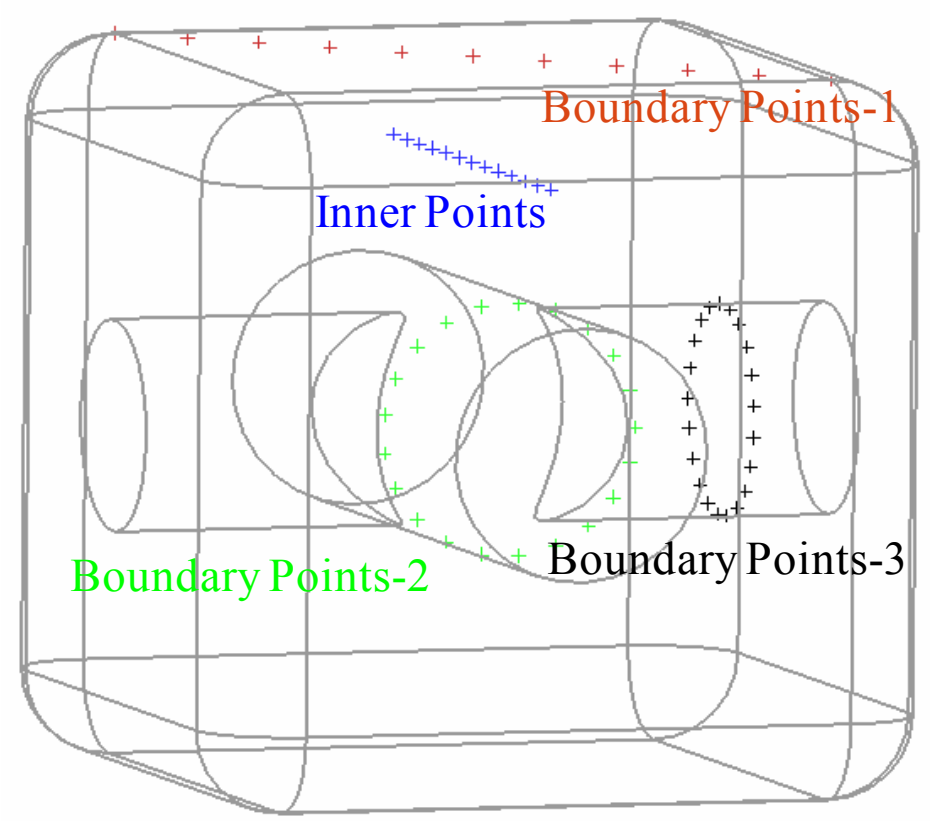
Analysis of body with trimmed surfaces

- **Node distribution and output locations**

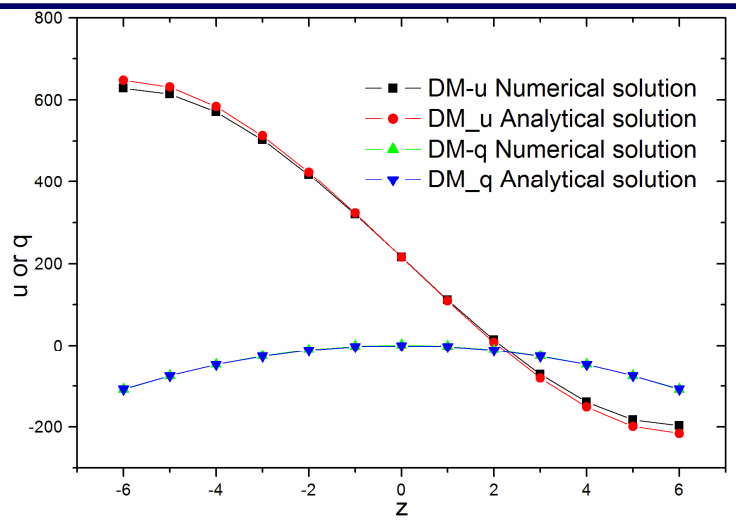
Analytical solution:

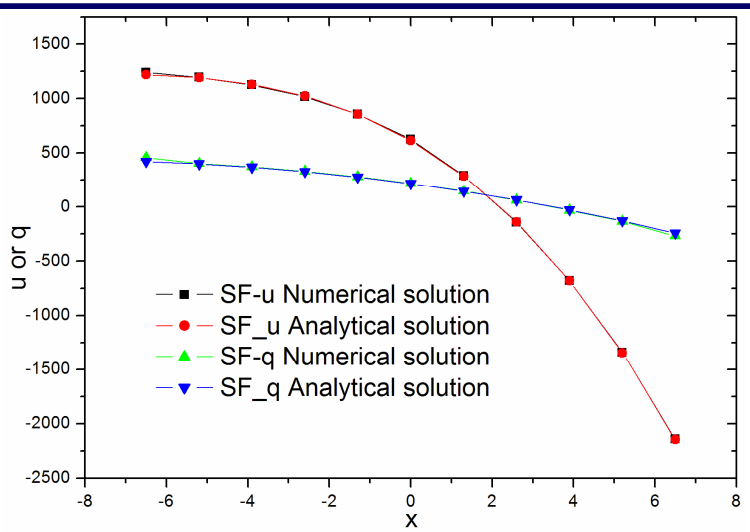
$$\phi = x^3 + y^3 + z^3 - 3yx^2 - 3xz^2 - 3zy^2$$

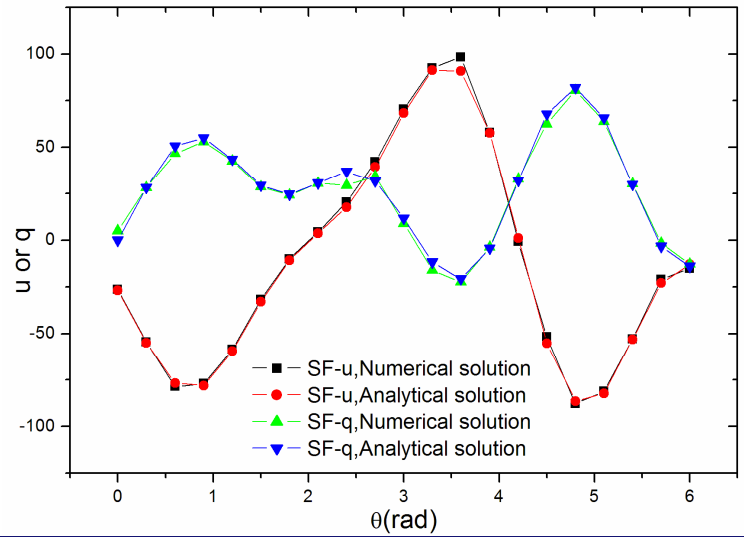


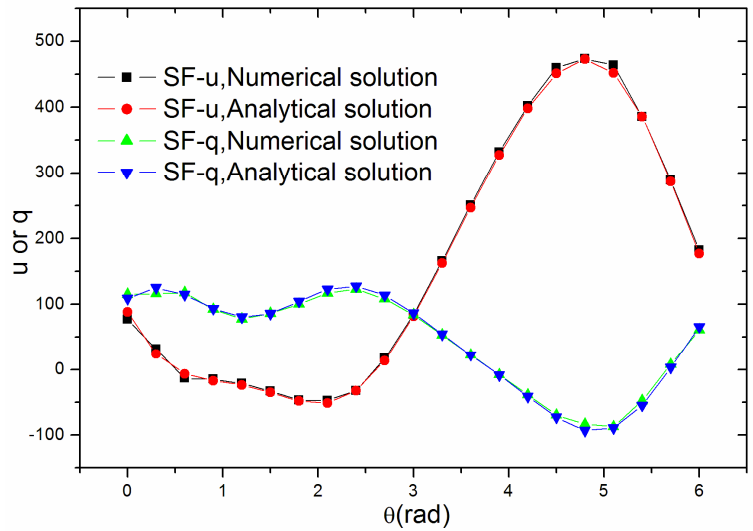


Analysis of body with trimmed surfaces









Conclusions

- Numerical results clearly demonstrate that the fast HdBNM is an efficient and accurate solution method for large scale analysis of structures with intricate geometries.

- The fast HdBNM retains the advantages of both the meshless method and the BEM. Therefore, the method has real potential for seamless interactions with the CAD packages and offering an automatic simulation tool for srtuctural analyses.