Real-time Computer Simulation of Three Dimensional Elastostatics using the Finite Point Method

2011 2nd International Conference on Mechanical, Industrial, and Manufacturing Technologies
(MIMT 2011)
26-28 February 2011, Singapore

Authors:

Kirana Kumara P, Ashitava Ghosal
Centre for Product Design and Manufacturing
Indian Institute of Science, Bangalore

Need for the Present Work

- ❖ Real-time and realistic simulation of biological organs – for surgical simulators
- ❖ Use continuum mechanics based models for better realism
- ❖ Test the procedure on benchmark problems – beam, bar

Present Approach to Achieve the Speed Needed for Real-time Performance

- ✓ Use the simplified material behaviour – linear elastic
- ✓ Use Finite Point Method (FPM)
- ✓ Use a Graphics Processing Unit (GPU)

Governing Equations for Linear Elastostatics

$$(\lambda+\mu)\left\{\frac{\partial^2 u_x}{\partial x^2}+\frac{\partial^2 u_y}{\partial x\partial y}+\frac{\partial^2 u_z}{\partial x\partial z}\right\}+\mu\left\{\frac{\partial^2 u_x}{\partial x^2}+\frac{\partial^2 u_x}{\partial y^2}+\frac{\partial^2 u_x}{\partial z^2}\right\}=0$$

$$(\lambda+\mu)\left\{\frac{\partial^2 u_y}{\partial y^2}+\frac{\partial^2 u_x}{\partial x\partial y}+\frac{\partial^2 u_z}{\partial y\partial z}\right\}+\mu\left\{\frac{\partial^2 u_y}{\partial x^2}+\frac{\partial^2 u_y}{\partial y^2}+\frac{\partial^2 u_y}{\partial z^2}\right\}=0$$

$$(\lambda+\mu)\left\{\frac{\partial^2 u_z}{\partial z^2}+\frac{\partial^2 u_x}{\partial x\partial z}+\frac{\partial^2 u_y}{\partial y\partial z}\right\}+\mu\left\{\frac{\partial^2 u_z}{\partial x^2}+\frac{\partial^2 u_z}{\partial y^2}+\frac{\partial^2 u_z}{\partial z^2}\right\}=0$$

$$\{\sigma\}=[D]\{\varepsilon\}$$

$$[D]=\begin{bmatrix} D_{11} & D_{12} & D_{12} & & & \\ D_{12} & D_{11} & D_{12} & & & \\ D_{12} & D_{12} & D_{11} & & & \\ & & & (D_{11}-D_{12})/2 & & \\ & & & & (D_{11}-D_{12})/2 & \\ & & & & & (D_{11}-D_{12})/2 \end{bmatrix}$$

$$\lambda=\frac{\nu E}{(1+\nu)(1-2\nu)} \qquad \mu=\frac{E}{2(1+\nu)} \qquad D_{11}=\frac{E(1-\nu)}{(1-2\nu)(1+\nu)} \qquad D_{12}=\frac{E\nu}{(1-2\nu)(1+\nu)}$$

Finite Point Method (FPM)

(E. Onate, 2001, Computers and structures)

$$u(x) \cong \hat{u}(x) = \sum_{l=1}^{m} p_l(x)\alpha_l = \mathbf{p}(x)^{\mathrm{T}}\boldsymbol{\alpha} \qquad \text{-----------(1)}$$

where $\boldsymbol{\alpha} = [\alpha_1, \alpha_2, \ldots \alpha_m]^{\mathrm{T}}$

Function $u(x)$ can now be sampled at the n points belonging to Ω_i giving

$$\mathbf{u}^h = \begin{Bmatrix} u_1^h \\ u_2^h \\ \vdots \\ u_n^h \end{Bmatrix} \cong \begin{Bmatrix} \hat{u}_1 \\ \hat{u}_2 \\ \vdots \\ \hat{u}_n \end{Bmatrix} = \begin{Bmatrix} \mathbf{p}_1^{\mathrm{T}} \\ \mathbf{p}_2^{\mathrm{T}} \\ \vdots \\ \mathbf{p}_n^{\mathrm{T}} \end{Bmatrix} \boldsymbol{\alpha} = \mathbf{C}\boldsymbol{\alpha} \qquad \text{----(2)}$$

- In FPM, n>m and hence approximation cannot fit nodal values.
- This problem can be overcome by determining the u(x) values by minimizing the sum of the square distances of the error at each point weighted with a function φ(x) as

$$J = \sum_{j=1}^{n} \varphi(x_j)(u_j^h - \hat{u}(x_j))^2 = \sum_{j=1}^{n} \varphi(x_j)(u_j^h - \mathbf{p}_j^{\mathrm{T}}\boldsymbol{\alpha})^2 \qquad \text{------(3)}$$

with respect to the $\boldsymbol{\alpha}$ parameters. This approximation is termed weighted least square (WLS) interpolation.

Standard minimization of Eq (3) gives

$$\boldsymbol{\alpha} = \bar{\mathbf{C}}^{-1}\mathbf{u}^h, \quad \bar{\mathbf{C}}^{-1} = \mathbf{A}^{-1}\mathbf{B}, \qquad \text{-----(4)}$$

$$\mathbf{A} = \sum_{j=1}^{n} \varphi(x_j)\mathbf{p}(x_j)\mathbf{p}^{\mathrm{T}}(x_j), \qquad \text{-----(5)}$$

$$\mathbf{B} = [\varphi(x_1)\mathbf{p}(x_1), \varphi(x_2)\mathbf{p}(x_2), \ldots, \varphi(x_n)\mathbf{p}(x_n)]$$

Final approximation obtained by substituting **α** from Eq (4) into Eq (1):

$$\hat{u}(x) = \mathbf{p}^{\mathrm{T}}\bar{\mathbf{C}}^{-1}\mathbf{u}^{h} = \mathbf{N}^{\mathrm{T}}\mathbf{u}^{h} = \sum_{j=1}^{n} N_j^i u_j^h, \qquad \text{-----(6)}$$

where the "shape functions" are

$$N_j^i(x) = \sum_{l=1}^{m} p_l(x)\bar{C}_{lj}^{-1} = \mathbf{p}^{\mathrm{T}}(x)\bar{\mathbf{C}}^{-1}. \qquad \text{------(7)}$$

Discretization of governing equations

$$A(u_j) = 0 \quad \text{in } \Omega \qquad \text{-----(8)}$$

with boundary conditions

$$u_j - \bar{u}_j = 0 \quad \text{on } \Gamma_u, \qquad \text{------(9)}$$

$$B(u_j) = 0 \quad \text{on } \Gamma_t. \qquad \text{-----(10)}$$

Substituting Eq (6) in Eq (8), Eq (9) and Eq (10) and collocating the differential equation at each point in the analysis domain,

$$[A(\hat{u}_j)]_p = 0 \quad p = 1, 2 \ldots N_r,$$
$$\left[\hat{u}_j\right]_s - \bar{u}_j = 0 \quad s = 1, 2 \cdots N_u,$$
$$\left[B(\hat{u}_j)\right]_r = 0 \quad r = 1, 2 \ldots N_t.$$

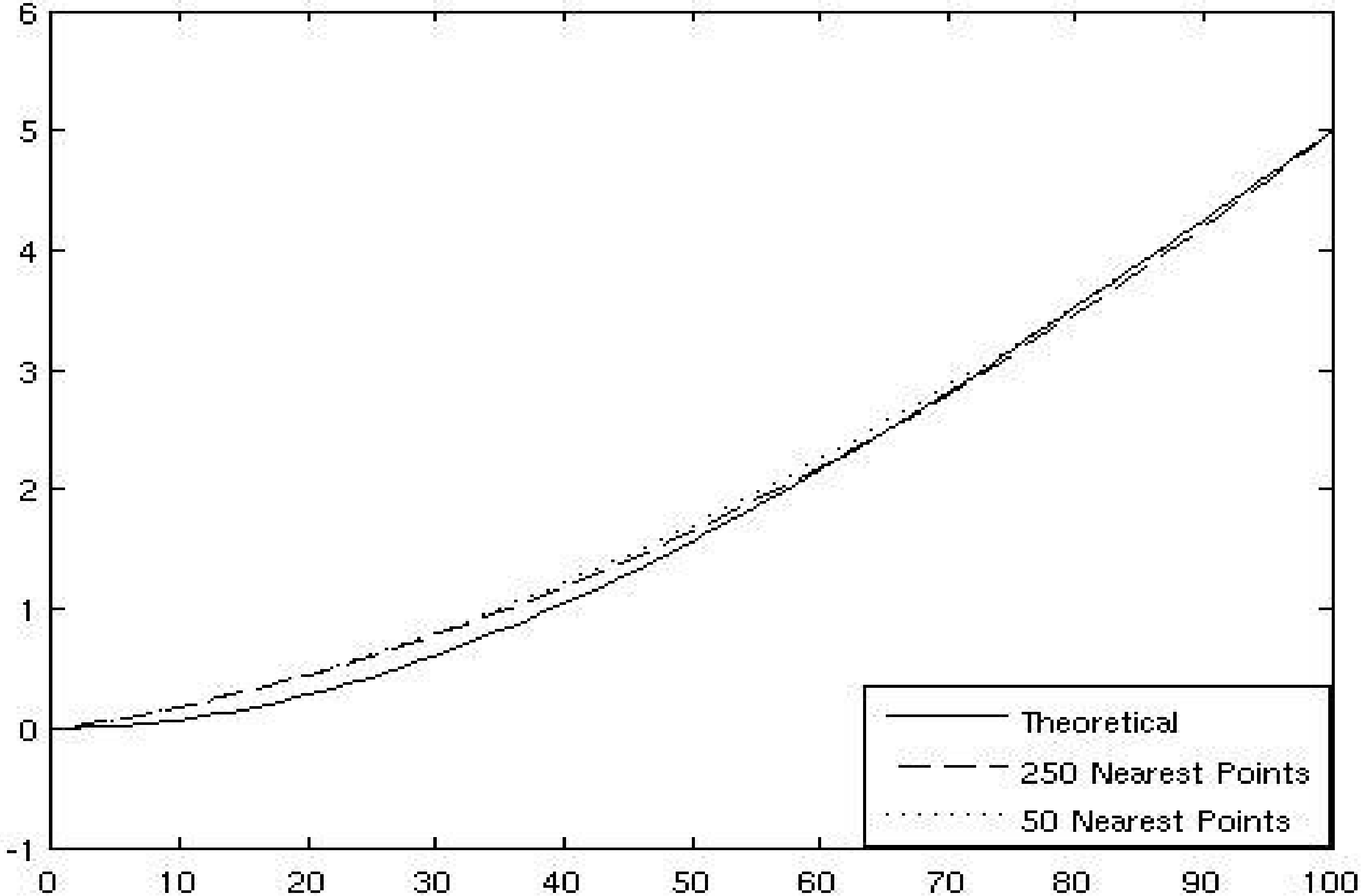
-------(11)

Sample Problem Description

A beam of length=99 mm and (4mmX4mm) cross section is discritized by uniformly spaced nodes located 1mm apart. Linear elastic material behaviour is assumed and strains and displacements are assumed small. E=200000N/mm^2 , v=0.33

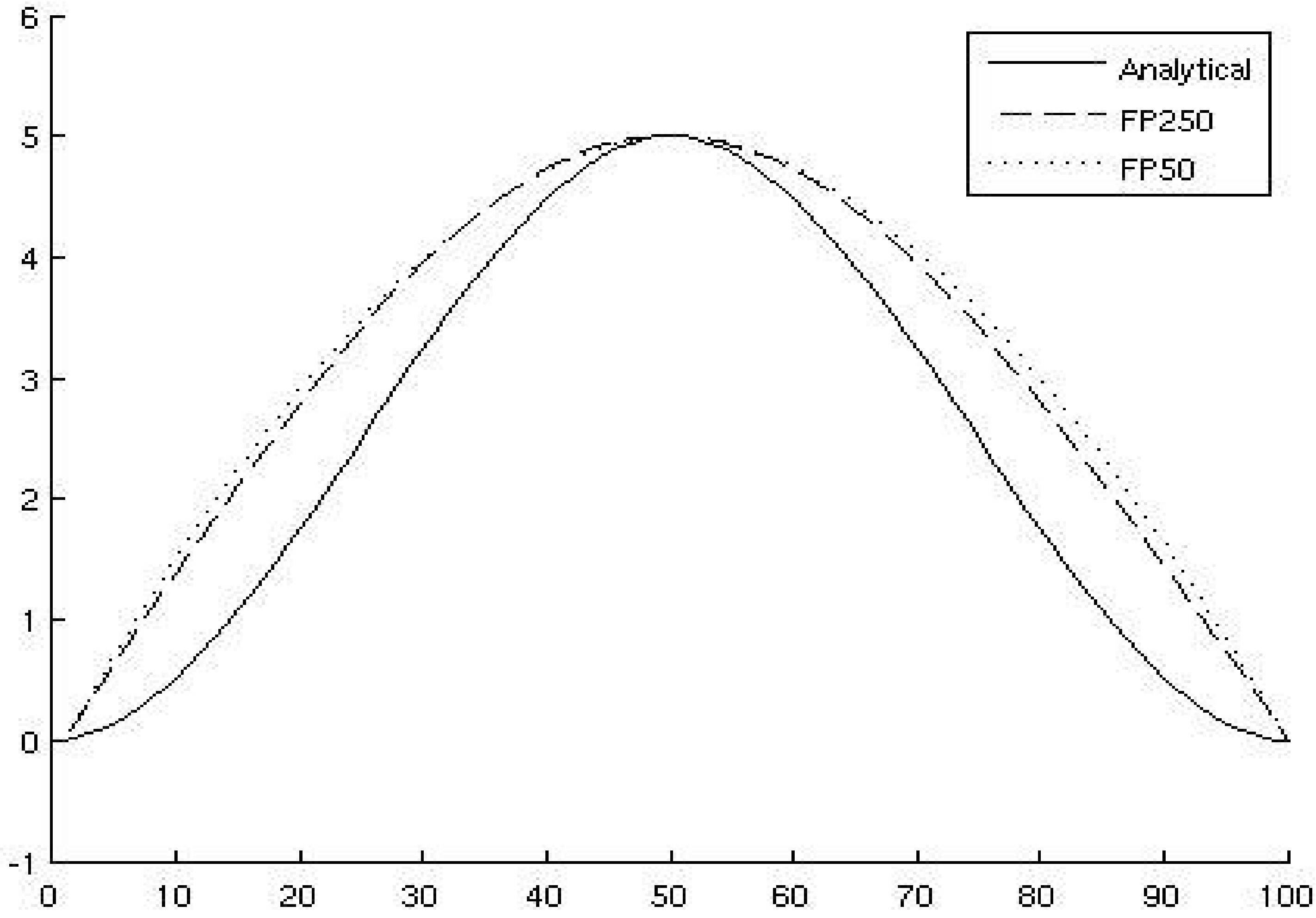
Results from FPM

Cantilever:

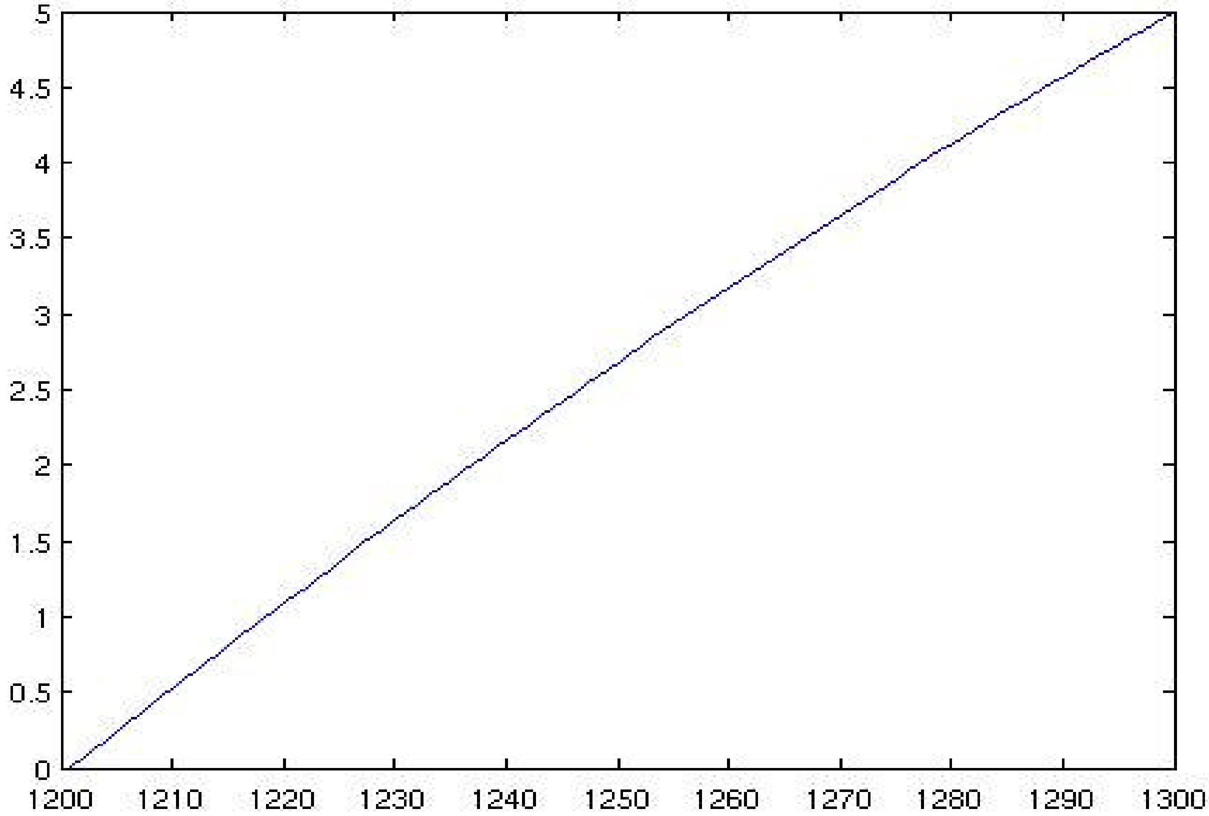


Theoretical formula: $y=(-1/6)(F/EI)(x^3 - 3l^2 x + 2l^3)$, x being measured from tip

Fixed at both ends:



Uniaxial tension:

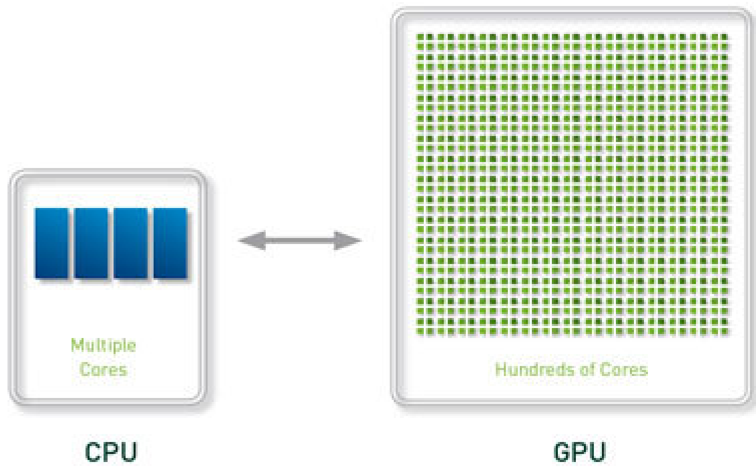


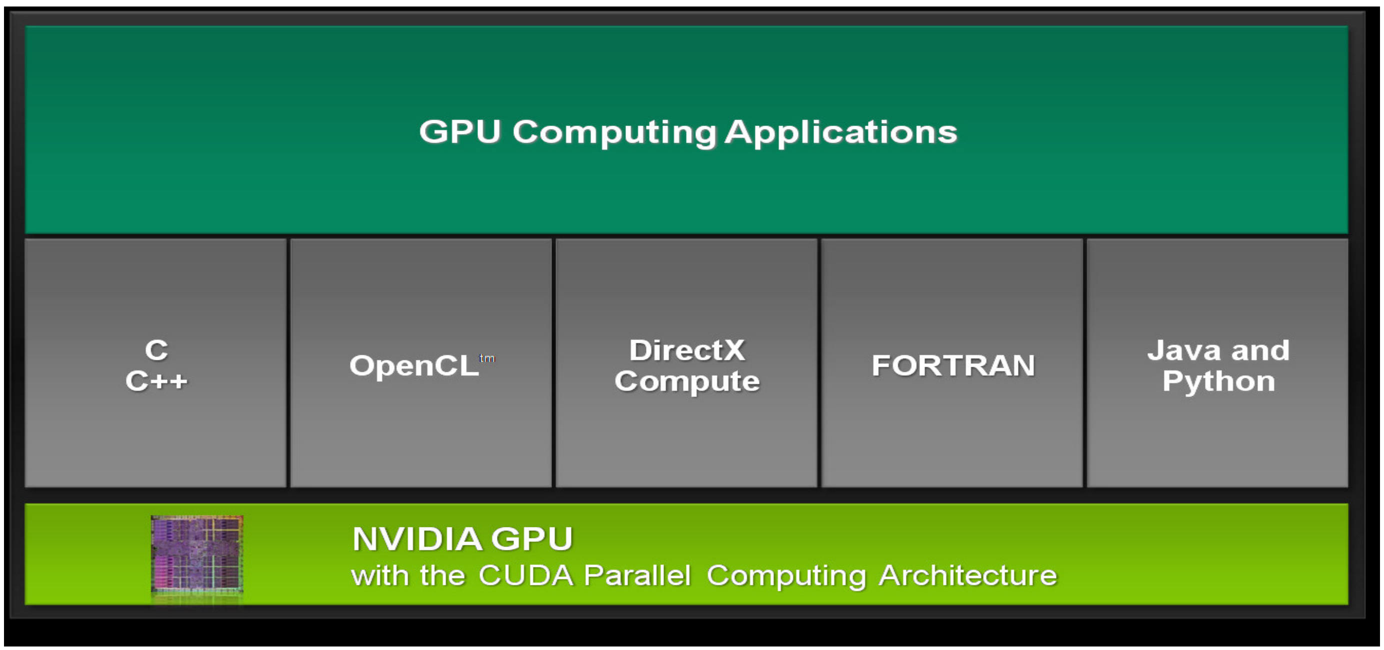
Truncated exponential weight function is used as the weight function:

$$w(r) = \frac{\exp(-r^2/c^2) - \exp(-r_m{}^2/c^2)}{1 - \exp(-r_m{}^2/c^2)}$$

$$c = 0.25\, r_{\max} \qquad r_m = 2\, r_{\max}$$

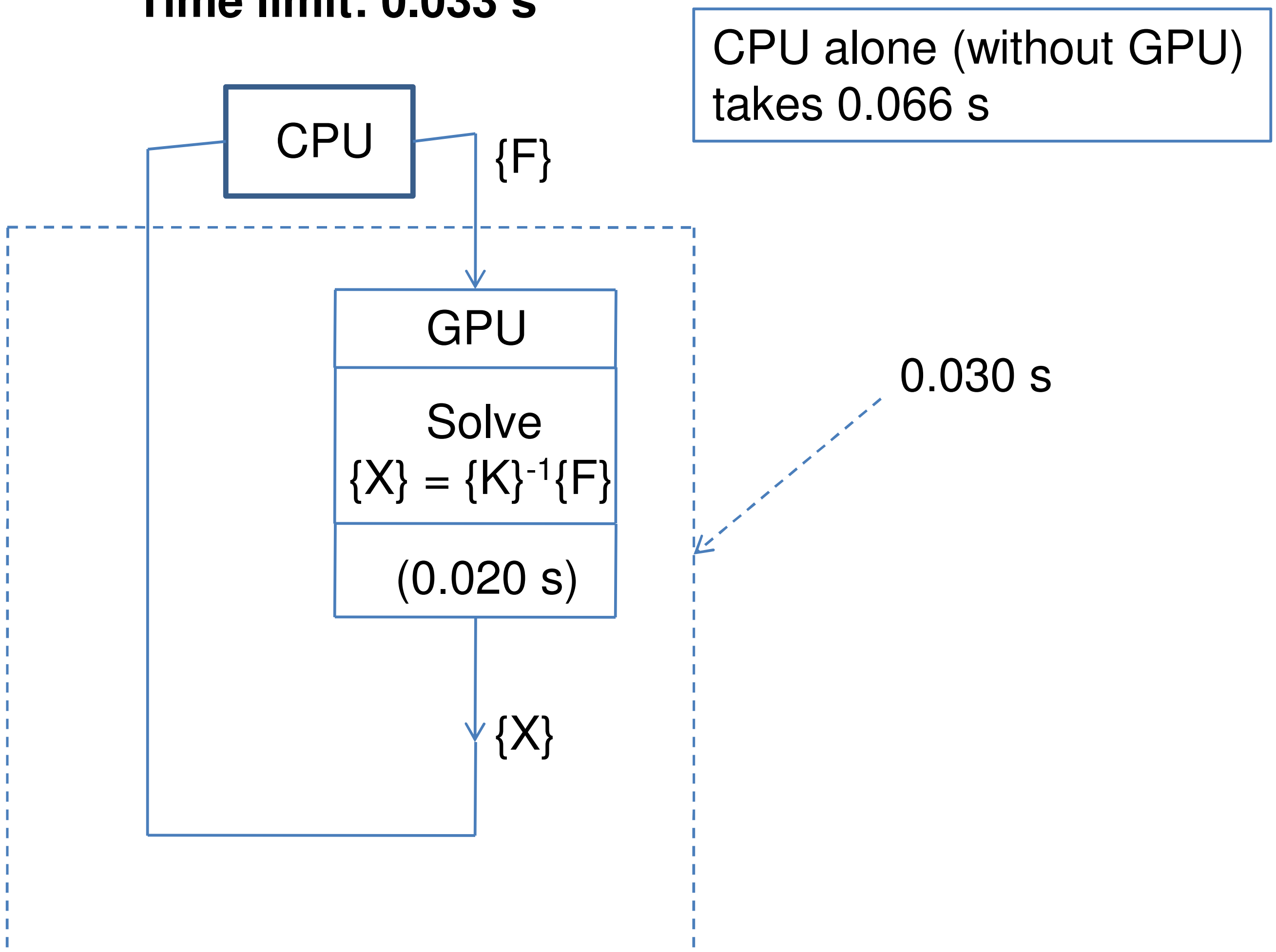
GPU (Images from www.nvidia.com)





Solution Strategy and Solution Times (GPU used: NVIDIA GeForce GTX 460)

Time limit: 0.033 s



(MATLAB and GPUmat are used for obtaining the results)

Conclusions

- CPU alone > Real-time not possible
- Real-time graphical simulation possible with GPU acceleration

References

[1] OPENHAPTICS™ TOOLKIT version 2.0, PROGRAMMER'S GUIDE, SensAble technologies, 2005

[2] U. Meier, O. Lopez, C. Monserrat, M. C. Juan, M. Alcaniz, "Real-time deformable models for surgery simulation: a survey," Computer Methods and Programs in Biomedicine (2005) 77, 183-97

[3] L S Srinath, Advanced Mechanics of Solids, Tata McGraw-Hill, Third Edition, 2009

[4] Yi-Je Lim, Suvranu De, "Real time simulation of nonlinear tissue response in virtual surgery using the point collocation-based method of finite spheres," Comput. Methods Appl. Mech. Engrg. 196 (2007) 3011–3024

[5] (2010) The MATLAB website. [Online]. Available: http://www.mathworks.com/

[6] (2010) GPUmat: GPU toolbox for MATLAB. [Online]. Available: http://gp-you.org/

[7] An Introduction to Meshfree Methods and Their Programming, G R Liu and Y T Gu, Springer,2005

[8] Meshfree and particle methods and their applications, Shaofan Li and Wing Kam Liu, Appl Mech Rev vol 55, no 1, January 2002

[9] A finite point method for elasticity problems, E. Onate, F. Perazzo, J. Miquel, Computers and Structures 79(2001) 2151-2163

[10] Modelling of soft tissue deformation for laparoscopic surgery simulation, G. Szekely, Ch. Brechbuhler, R. Hutter, A. Rhomberg, N. Ironmonger, P. Schmid, Medical Image Analysis 4 (2000) 57–66

Thank You!