Multi-Scale Simulation For Microsurgery Trainer

K. M. Lim[1,2], F. Wang[1], T. Poston[3], L. Zhang[1], C. L. Teo[1], E. Burdet[1,2]

[1]Dept of Mechanical Engineering and [2]Division of Bioengineering, National University of Singapore

[3]Johns Hopkins Singapore

Email: limkm@nus.edu.sg, e.burdet@ieee.org, http://guppy.mpe.nus.edu.sg/~eburdet/

***Abstract*—For use in a Virtual Reality based training system for surgical micromanipulation, we have developed a fast multi-scale FEM algorithm that concentrates detail where needed while still handling global deformations. The resulting 6- to 7-fold speed up is promising for the development of real-time simulation of the mechanical response of a virtual organ or tissue.**

Keywords: microsurgery training; multi-scale FEM; real-time simulation; large deformation;

I. Introduction

In microsurgery, surgeons need to stitch blood vessels or skin under a microscope. There are two key aspects to the extensive training needed.

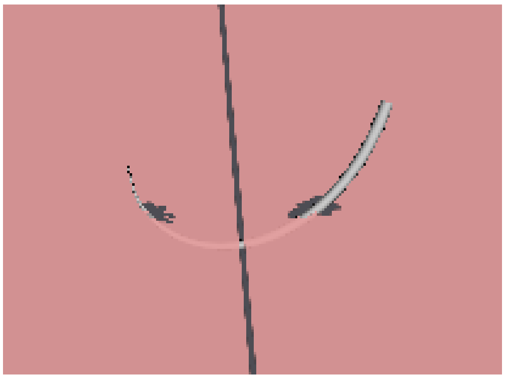

Figure 1. Manipulating a geometric 'suture needle' with small movements of the hand, visually displayed as large, as in a microscope.

The first is to control tremor at the microscopic level, and to master changed hand-eye coordination with angles rotated and distances magnified by the microscope. Natural force feedback is low in microscopic work, so that one option is to work with purely visual control, though haptic training cues may be useful. Fig. 1 shows a virtual needle being guided through geometric skin, with the objective of making the two holes as small and correctly placed as possible.

The surgeon must also learn to control the behavior of the tissues and threads involved. This involves large deformations of viscoelastic materials, with long-distance interactions (for instance, between a point of grasp and an anchor point of a membrane or tissue) so that local deformation is not enough. Simulating the mechanics of this requires fast computation, even for visual feedback (~60 updates/*sec*), and much greater speed for force feedback (~1000 updates/*sec*). While real forces in microsurgery are small, we aim to research the training value of exaggerated forces, for which we must model large global deformations at the higher speed. This paper describes an approach to achieving such speeds, on a testbed that does not yet address tissue modelling in detailed system.

This paper first presents the overall target simulation environment (Section II), then a multi-scale FEM algorithm which achieves fast computation compared to the traditional FEM algorithm with low loss of accuracy (Section III–V).

II. Hand-Eye Coordinated Training

In collaboration with the National University Hospital of Singapore, we are developing a robotic surgical trainer [1], with low cost and risk compared to traditional training methods. Fig. 2 shows our general workstation with a rigidly frame-mounted 6DOF DELTA haptic interface. This provides a 3D workspace usable in many ways, including a trainer for Chinese calligraphy [2]. A semi-transparent panel creates a virtual monitor image corresponding to the haptic workspace.

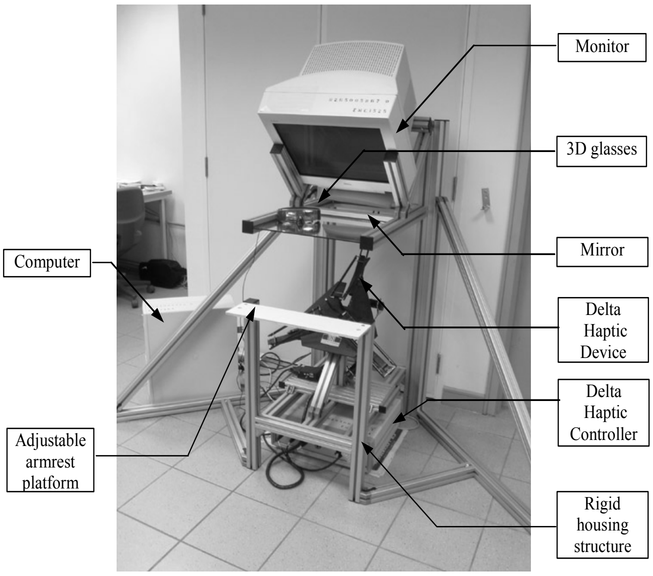

Figure 2. A simulation workstation with 3D view and haptic response provides intuitive manipulation and real-time interaction with virtual objects.

Fig. 3 shows the schema for microsurgery training, using 6 degrees of freedom (DOF) haptic devices and 2DOF haptic forceps [3], and a microscope represented by stereoscopic visual display.

This project was funded by the National Medical Research Council of Singapore (NMRC) and NUS.

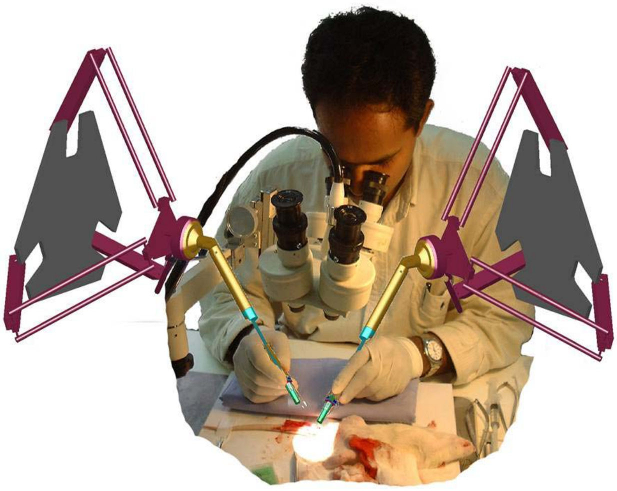

Figure 3. Planned working space

III. Deformation modelling

The modelling of soft tissue mechanics has become an important research topic in robot-assisted surgery and surgical simulation. Much effort has aimed at the development of real-time simulation of the physical behaviour of deformable tissue and the integration of these methods into simulators [4–8]. However, computation speed has always been a major barrier to accurate real-time tissue models for haptic feedback, whether with quasi-static [9, 10] or dynamic [11–13] models. Early developments often simulated deformable bodies using mass-spring systems, for simplicity of implementation and the low computation compared to more realistic stress-strain relations. The body is represented by a set point masses connected by elastic links and the equation of motion is written for each point mass. However, the mass and spring constants to be used in the model cannot be derived from measured properties like the shear modulus, and thus the models do not capture underlying physics properly. Moreover, since the constants are assigned heuristically, varying the level of detail (essential to adequate speed) is hard to systematize.

Realistic elasticity calculations often use the finite element method (FEM). However, it is seldom used in real-time simulation [14–16], due to its complex implementation and expensive computation. It provides more accurate description of deformation mechanics than mass-spring models, but for real-time work it needs modification for adequate accuracy while reducing the computation time.

Several fast algorithms [17–21] have recently been the focus of research in real-time simulation, especially with the need for haptic feedback whose sampling rate is much higher than that needed for visual display. Bro-Nielsen and Cotin [18] used a condensation method to reduce the number of unknowns to only the surface degrees of freedom. Others use an adaptive mesh to reduce the number of unknowns. DiMaio and Salcudean [20] used a real-time mesh refinement technique to generate a multi-grid, with distinct fine and coarse grids, for their needle insertion simulation. Wu et. al. [21] proposed a dynamic progressive mesh which is an extension of the progressive mesh concept. The idea is to generate a hierarchy of meshes off-line by collapsing mesh edges recursively, starting from of a fine mesh, and then using elements from the different meshes in the final computation. They presented simulation results for a two level mesh, but the technique to combine elements from multiple levels in the hierarchy need to be developed.

In this paper, we describe a multi-scale FEM algorithm which uses elements from multiple levels in a hierarchy of mesh similar to the progressive mesh. We show that this algorithm computes fast compared to the traditional FEM algorithm with low loss of accuracy. This algorithm has also been integrated with a visual/haptic feedback workstation as shown in Fig. 2.

IV. Multi-scale FEM for soft tissue simulation

The main idea of our multi-scale FEM is to dynamically use different levels of mesh, at different places (not uniformly progressing from ‘coarse all over’ to ‘fine all over’, as in multi-grid detailed solutions). We create a fine mesh in the domain and then generate coarser elements at various levels by recursively combining adjacent elements. The final mesh used to the computation is formed by choosing elements from different levels, with small elements near an area of interest (for example, where a force is applied or bending is high) and progressively coarser elements as we move away from the region of interest. Fig. 4 shows levels of the mesh for a rectangular domain, starting from fine (top left) to coarse (bottom middle). The multi-scale mesh for an applied line force (bottom right) combines elements from the different levels.

Such a multi-scale mesh especially suits the application of surgical simulation, as the surgical tool usually applies a force on the tissues/organs only in a localized region around it. Hence small elements are placed around the contact region, as haptic feedback needs accurate deformation and force here accurately, and they can capture the large displacement gradient in that region. Progressively larger elements suffice as we move away from the tool, for the accuracy needed for the results for visual display. This multi-scale mesh is formed in real-time as the surgical tool moves over the domain by choosing the correct elements from the various mesh levels. The stiffness matrices of the elements of each level are pre-calculated and stored at the beginning of the simulation. They are assembled in real-time to form the global stiffness matrix for the multi-scale mesh during the simulation.

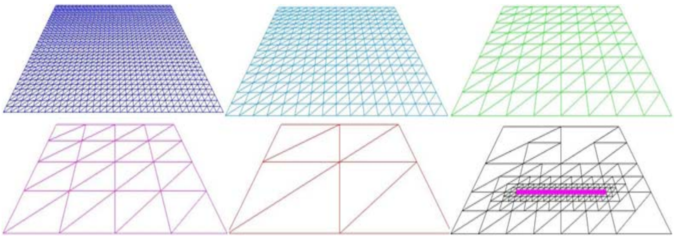

Figure 4. Multi-Scale mesh generation.

Such a mesh greatly decreases the model's DOF compared to the fine mesh, and so shrinks the computational time. In the example of Fig. 5 the fine mesh model has 961DOF which is reduced to 138DOF in the multi-mesh.

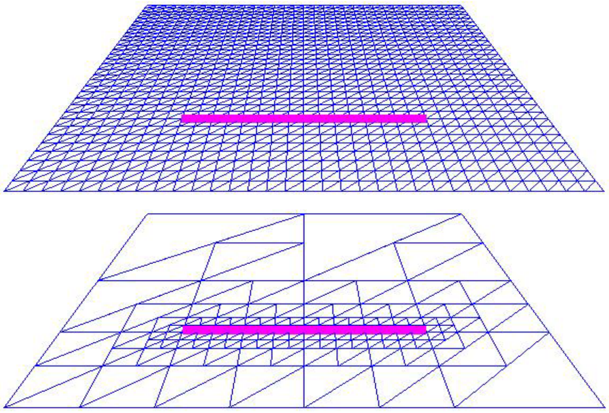

Figure 5. Fine mesh FEM and multi-scale mesh FEM

We implemented the method for the simple case of 2D out-of-plane or anti-plane strain problem. Three-dimensional motion with bending and stretching will be more complex, but can be handled by the same mesh logic.

The governing equation, with applied pressure P, is

$$\nabla\left(G\nabla w\right)+P=0 \qquad (1)$$

where G is shear modulus, and w out-of-plane displacement.

The force applied by the haptic device is modeled by a line load on the 2D domain. The case of a concentrated point force has been studied in [22], and a singularity in the deformation was observed. Such singularity does not arise with a line force. A line force also allows us to derive torque even for the scalar out-of-plane deformation case, around horizontal though not vertical axes, and display it using our 6DOF haptic device. The algorithm is developed to accept any input of shapes in real-time but not only restricted to a point or a line. Here we use the line force so as to compare the multi-scale method's accuracy to that of the *Navier's* serious solution (See Appendix for details).

With a line load applied at $y=y_1$ and between $x=x_1$ to $x=x_2$, it takes the form

$$P(x,y)=F_{\delta}\Pi\left(x,x_1,x_2\right)\delta\left(y-y_1\right) \qquad (2)$$

where $$\Pi=\begin{cases}1 & x_1<x<x_2\\ 0 & otherwise\end{cases} \qquad (3)$$

δ = Dirac delta function

In the FEM formulation, the displacement in an element is approximated by

$$w(x,y)=\sum_{i=1}^{3}N^{(i)}w^{(i)} \qquad (4)$$

where $w^{(i)}$ is the nodal displacement and $N^{(i)}$ is the shape function associated with node i. The weak form of (1) is

$$\iint_{\Omega}N^{(i)}\nabla(G\nabla w)d\Omega+\iint_{\Omega}N^{(i)}Pd\Omega=0 \qquad (5)$$

with the shape functions used as weights in the domain integral. This gives a system of algebraic equations of the form

$$\left[K\right]\left\{U\right\}=\left\{F\right\} \qquad (6)$$

where {U} is the vector of nodal displacements, {F} represents the applied force, and [K] is the stiffness matrix of the system.

Our current implementation treats a square domain fixed so that its edges have zero displacement. We need to determine (1) the reaction force at the haptic device and (2) the displacements elsewhere given the displacement of the haptic device.

V. SIMULATION RESULTS

We implemented the multi-scale FEM algorithm to study its accuracy and speed. Fig. 6 shows four deformation states of a rectangular soft domain (blue mesh) deformed out of plane. The red line and yellow point are analogous to microsurgery tools in contact with soft tissue, which effectively apply a line load and a concentrated force on the soft tissue. The vertical bars at right show the values and directions of the contact forces.

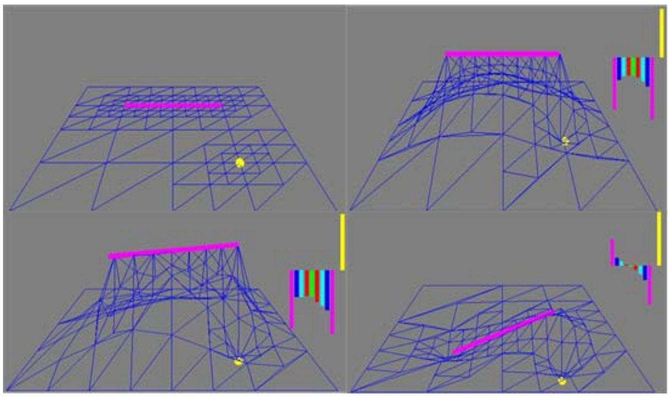

Figure 6. Simulation of multi contact points

A. Analysis of accuracy

We compare our FEM results with Navier's series solution (See Appendix for details) to test their accuracy. The series solution is known to converge rapidly to the exact solution for a rectangular domain. We considered a 20x20*cm* domain pressed down 1*cm* by a 10*cm* bar, as shown in Fig. 7. The shear module G is 1*N/cm*^{2}.

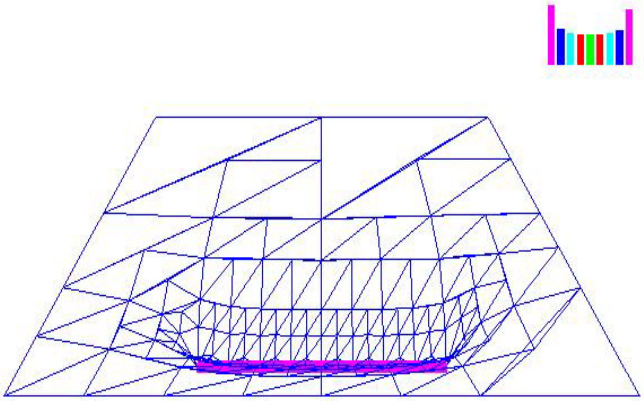

Figure 7. Deformation and forces acting of the 2D domain.

1) *Comparison of the calculated line force.*

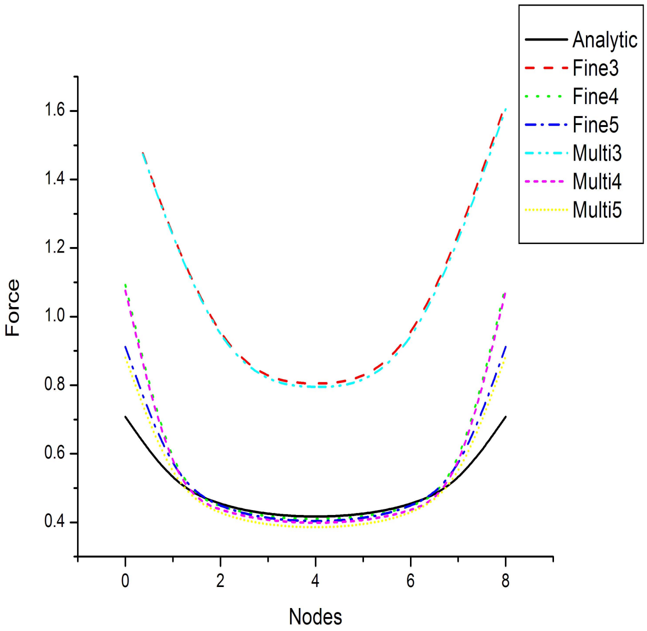

Figure 8. Comparison of calculated line force among analytic method, fine mesh FEM and multi-scale FEM at different levels.

Fig. 8 shows the distributed force acting along the bar obtained from the fine mesh FEM and multi-scale FEM and Navier's series solution. Forty terms are used in Navier's method to give a converged solution, the FEM results also converge to the series solution when enough levels are used. The results for the fine-mesh and multi-scale FEM are indistinguishable. The current model problem needs at least four levels for a good result. With four levels the resultant force found with the fine versus multi-scale meshes differ by only 4%.

2) *Comparison of RMS error.*

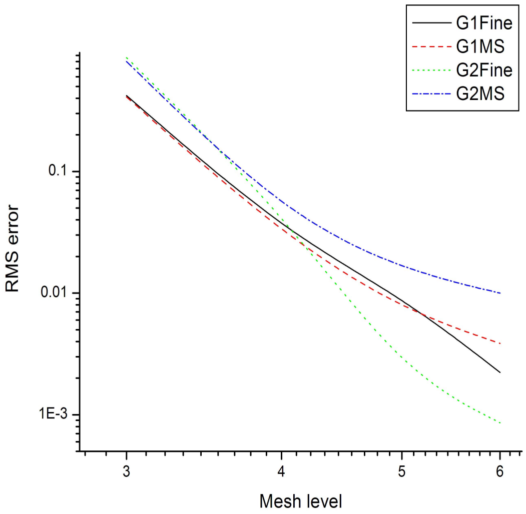

Figure 9. RMS (Root Mean Square) Error , by mesh level count.

Fig. 9 shows the Root Mean Square (RMS) error of the fine and multiscale FEMs compared to the Navier solution for two loading condition: (G1) with the bar at the center of the mesh, and (G2) with the bar near the border. The RMS errors decrease dramatically with more levels and degrees of freedom. For (G1), the multi-scale result is as good as the fine scale for up to 5 levels. For (G2), with the bar near the edge, the multi-scale mesh fails to match the accuracy of the fine mesh beyond level 4, due to the larger elements used further away from the bar. Nevertheless, the accuracy of the multi-scale mesh at these levels is about 1% to 5% of the Navier solution, normally acceptable for virtual-reality simulation where stability and speed outweigh high accuracy.

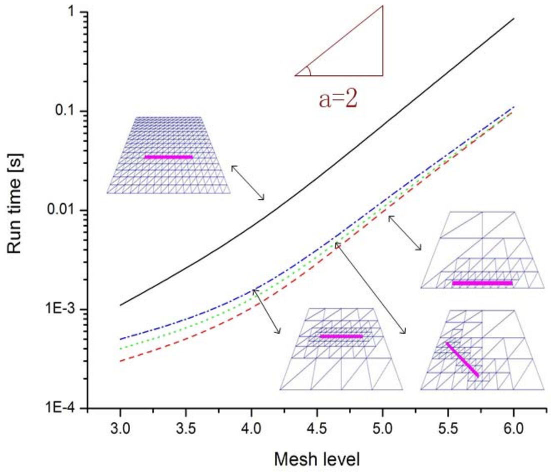

Figure 10. CPU time for multi-scale and fine mesh, with mesh level count.

B. *Analysis of speed*

Fig. 10 compares CPU time using the multi-scale and fine meshes, with the count of mesh levels used. Three different cases of load position and orientation were used for multi-scale FEM method. For each case, the multi-scale FEM uses a different set of elements with a different number of degrees of freedom. This gives slightly different timing in each case. The computation time for the fine mesh FEM (included for comparison) does not change with loading case, since the number of degrees of freedom in the model is fixed.

The results show that CPU time scales roughly as *O(N^2)* for both the fine and multi-scale mesh, where *N* is the number of nodes in each of the mesh. The complexity of the solution process is the same in both cases as the same iterative conjugate gradient solver is used. Here, the major gain in efficiency of the multi-scale mesh is the great reduction in the number of nodal unknowns to be solved while maintaining good accuracy. The number of unknowns N in the multi-scale mesh is about 15% of that in the fine mesh. Even with the extra time taken to assemble the global stiffness matrix in real-time, the multi-scale FEM is still about an order faster than fine-scale FEM.

For better comparison of the three cases, Fig. 11 shows the speed up factor of the multi-scale mesh over fine mesh FEM, defined as the ratio of their computation times. The multi-scale FEM improves the speed by 6 to 7 times at higher mesh levels.

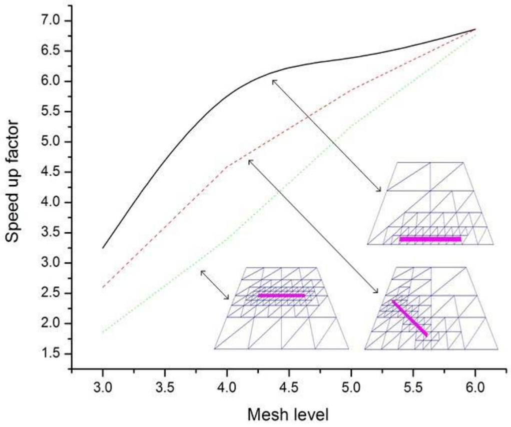

Figure 11. Speed up factor (Fine / MS FEM time) in three cases, with mesh level count.

VI. Discussion

These results show that a major speed up in computation can be achieved using multi-scale FEM, especially at higher mesh levels with more degrees of freedom. These are necessary for more accurate solutions, within say 5% of the 'exact' one. Even with the slight accuracy loss in the multi-scale mesh compared to fine (from 0.1% to 1.5% at level 6), the 'absolute' accuracy of the multi-scale mesh is sufficient for haptic simulation. The 6-to-7-fold speed up is critical for simulation in real-time.

In our implementation of multi-scale FEM on a Xeon 2.66 GHz CPU with 1 GB RAM, a level 4 simulation (with about 8% accuracy) takes 1*ms* to complete. This was just enough to provide the refresh rate of 1 *kHz* required for haptic feedback. The corresponding fine scale FEM (with about 5% accuracy) cannot meet this criterion. The multi-scale FEM provides a good trade-off between a major gain in speed against a slight loss in accuracy, essential for real-time simulation.

In conclusion, spatially varying mesh detail in surface modeling can support large global deformation at the speeds necessary to provide visual and haptic feedback in surgical training, by a high time saving vis-à-vis traditional FEM. It can be extended for non-linear deformation of 3D bodies, and other systems of mesh mechanics. Our ultimate goal in applying it is to model human tissue. It shows great potential to realize a real-time haptic simulation system.

Acknowledgments

We thank (in alphabetical order) Ankur Dhanik, Lim Beng Hai, James Rappel, Teo Cheng Young William and Tee Keng Peng for their collaboration on the micromanipulation learning project.

Appendix

1) Navier's Series Solution Formulation

The governing equation of anti-plane elasticity problem is:

$$\nabla(G\nabla w)+P=0 \qquad \text{(a1)}$$

A series solution for an *L*x*B* rectangular region may be found by Fourier expansion of the load and deflection

$$p(x,y)=\sum_{m=1}^{\infty}\sum_{n=1}^{\infty}P_{mn}\sin(\frac{m\pi x}{L})\sin(\frac{n\pi y}{B}) \qquad \text{(a2)}$$

$$w(x,y)=\sum_{m=1}^{\infty}\sum_{n=1}^{\infty}W_{mn}\sin\frac{m\pi x}{L}\sin\frac{n\pi y}{B} \qquad \text{(a3)}$$

The coefficients of this double Fourier expansion are given by:

$$P_{mn}=\frac{4}{LB}\int_0^L\int_0^B p(x,y)\sin\frac{m\pi x}{L}\sin\frac{n\pi y}{B}\,dydx \qquad \text{(a4)}$$

For a line force,

$$p(x,y)=F\Pi(x,x_1,x_2)\delta(y-y_1) \qquad \text{(a5)}$$

$$\begin{aligned}P_{mn}&=\frac{4}{LB}\int_0^L\int_0^B F\Pi(x,x_1,x_2)\delta(y-y_1)\sin\frac{m\pi x}{L}\sin\frac{n\pi y}{B}\,dydx\\&=\frac{4F}{m\pi B}\sin\frac{n\pi y_1}{B}\left[\cos\frac{m\pi x_1}{L}-\cos\frac{m\pi x_2}{L}\right]\end{aligned} \qquad \text{(a6)}$$

From equation (a3),

$$\nabla(G\nabla w)=G\sum_{m=1}^{\infty}\sum_{n=1}^{\infty}-W_{mn}\left[\left(\frac{m\pi}{L}\right)^2+\left(\frac{n\pi}{B}\right)^2\right]\sin\frac{m\pi x}{L}\sin\frac{n\pi y}{B} \qquad \text{(a7)}$$

Substituting this into (a1) gives:

$$\sum_{m=1}^{\infty}\sum_{n=1}^{\infty}\left\{-W_{mn}\left[\left(\frac{m\pi}{L}\right)^2+\left(\frac{n\pi}{B}\right)^2\right]+\frac{P_{mn}}{G}\right\}\sin\frac{m\pi x}{L}\sin\frac{n\pi y}{B}=0 \quad \text{(a8)}$$

The coefficients of this equation are identically zero,

$$W_{mn}\left[\left(\frac{m\pi}{L}\right)^2+\left(\frac{n\pi}{B}\right)^2\right]-\frac{P_{mn}}{G}=0 \quad \text{(a9)}$$

This gives:

$$W_{mn}=\frac{P_{mn}}{G}\frac{1}{\left(\frac{m\pi}{L}\right)^2+\left(\frac{n\pi}{B}\right)^2} \quad \text{(a10)}$$

Inserting equation (a6) and (a10) into equation (a3),

$$w(x,y)=\sum_{m=1}^{\infty}\sum_{n=1}^{\infty}\frac{4F\sin(n\pi y_1/B)\left[\cos(m\pi x_1/L)-\cos(m\pi x_2/L)\right]\sin\frac{m\pi x}{L}\sin\frac{n\pi y}{L}}{GBm\pi^3\left[\left(\frac{m}{L}\right)^2+\left(\frac{n}{B}\right)^2\right]} \quad \text{(a11)}$$

This above series solution for the anti-plane elasticity problem is used to check the accuracy of our FEM results.

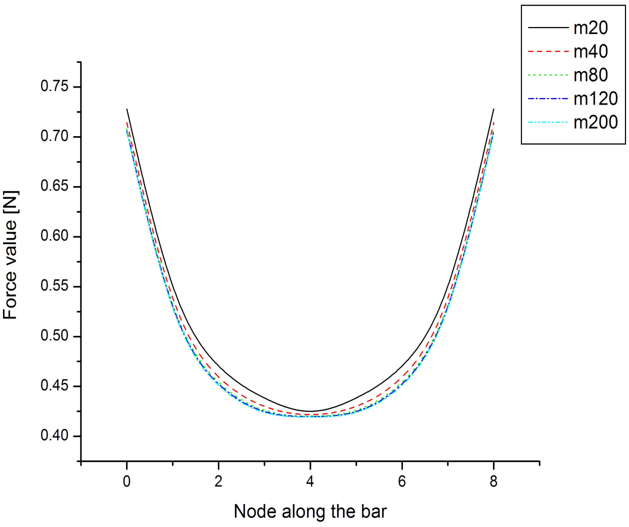

Figure 12. Force value along the bar

Fig. 12 shows the force values along the bar using the Navier's solution with different number of terms *m*, and the results converge rapidly as *m* increases.

REFERENCES

[1] F. Wang, T. Poston, C.L. Teo, K.M. Lim and E. Burdet; "Multisensory learning cues using analytical collision detection between a needle and a tube", Haptic Symposium, IEEE Int. Conference on Virtual Reality; 2004, in press.

[2] E. Burdet, C.L. Teo and H.P. Lim, "A Robotic Teacher for Chinese Ideograms", Haptic Symposium, IEEE Int. Conference on Virtual Reality; 2001.

[3] F. Mani, "Haptic Forceps for a Virtual Reality Microsurgery Trainer", Master Thesis, Swiss Federal Institute of Technology (EPFL), March 2003.

[4] F. Tendick, "A Virtual Environment Testbed for Training Laparoscopic Surgical Skills", Presence; vol. 9, No.3; June 2000; p. 236-255. Trans. Roy. Soc. London, vol. A247, pp. 529–551, April 1955.

[5] Ron Alterovitz, Jean Pouliot, Richard Taschereau, I-Chow Joe Hsu, and Ken Goldberg, Simulating Needle Insertion and Radioactive Seed Implantation for Prostate Brachytherapy. Medicine Meets Virtual Reality 11 (MMVR11), J.D. Westwood et al. (Eds.), IOS Press, January 2003, pp. 19-25.

[6] Mahvash, M., Hayward, V. 2003. Haptic Simulation of a Tool In Contact With a Nonlinear Deformable Body. International Symposium on Surgery Simulation and Soft Tissue Modelling, IS4TM, Lecture Notes in Computer Science, Springer Verlag, N. Ayache, H. Delingette (Eds), Springer Verlag, New York. 2003. pp. 311-320.

[7] C. Simone and A. M. Okamura, "Haptic Modeling of Needle Insertion for Robot-Assisted Percutaneous Therapy," *Proceedings of the IEEE International Conference on Robotics and Automation*, 2002, pp. 2085-2091.

[8] Christopher W. Kennedy, Tie Hu, Jaydev P. Desai, Andrew S. Wechsler, and J. Yasha Kresh, "A Novel approach to Robotic Cardiac Surgery using Haptics and Vision",Cardiovascular Engineering: An International Journal, 2002.

[9] Demetri Terzopoulos, John Platt, Alan Barr, and Kurt Fleischer, "Elastically deformable models." Computer Graphics, 21(4):205-214, July 1987. Proceedings of SIGGRAPH'87 (Anacheim, California).

[10] Gourret, JP, Magnenat Thalmann, N. and Thalman, D., "Simulation of Object and Human Skin De-formations in a Grasping Task". Computer Graphics 23, 3,Boston, 31 july - 4 august 1989, pp 21-30.

[11] Baumann R, Glauser D, "Force feedback for virtual reality based minimally invasive surgery simulator". Medicine meets virtual reality, San Diego CA (1996).

[12] Meseure P, Chaillou C, "Deformable body simulation with adaptive subdivision and cuttings". Proceedings of the WSCG'97: 361-370 ,1997.

[13] Kuehnapfel UG, Neisius B, "CAD-based graphical computer simultaion in endoscopic surgery". End. Surg, 1:181-184, 1993.

[14] M. Sagar, D. Bullivant, G. Mallinson, and P. Hunter, "A Virtual Environment and Model of the Eye for Surgical Simulation." In Computer Graphics Proceedings, Annual Conference Series, pages 205-212, 1994.

[15] Keeve, E., Girod, S., Pfeifle, P., Girod, B., "Anatomy-Based Facial Tissue Modeling Using the Finite Element Method", IEEE Visualization 96, pp. 21-28.

[16] S. Peiper, J Rosen, and D. Zeltzer, "Interactive graphics for plastic surgery: A task-level analysis and implementation." In Symposium on Interactive 3D Graphics, 1992.

[17] O. R. Astley, V. Hayward, "Multirate Haptic Simulation Achived by coupling Finite Element Meshes through Norton Equivalents", Internal conference on Robotics & Automation; 1998.

[18] Bro-Nielsen, M., and Cotin, S. "Real-time volumetric deformable models for surgery simulation using finite elements and condensation", Computer Graphics Forum (Eurographics '96), 15(3):57--66, 1996.

[19] S. Cotin, H. Delingette, and N. Ayache, "A Hybrid Elastic Model allowing Real-Time Cutting, Deformations and Force-Feedback for Surgery Training and Simulation.", The Visual Computer, 16(8), pp. 437-452, 2000.

[20] S.P. DiMaio and S.E. Salcudean, "Simulated Interactive Needle Insertion", Haptic Symposium, IEEE Int. Conference on Virtual Reality; 2002.

[21] X. Wu, M.S. Downes, "A Adaptive Nonlinear Finite Elements for Deformable Body Simulation Using Dynamic Progressive Meshes", Eurographics; Vol 20 ; 2001

[22] K.M. Lim, T. Poston, L. Zhang, B.F. Liu, C.L. Teo, E. Burdet, "Multi-Scale Simulation for A Robotic Surgical Trainer", ICBME 2002.