Applying GPU Data-Parallelism to an Agent Based Model (ABM) of Tissue Inflammation

Yun (Yvonna) Li[1], Yee Key (Nicole) Li[3], Alireza Najafi Yazdi[2], Luc Mongeau[2], Caroline Shung[2]

[1]Department of Civil Engineering, [2]Department of Mechanical Engineering, McGill University, Montreal, Quebec, Canada; [3]Department of Hearing and Speech Sciences, University of Maryland, College Park, Maryland, USA

Introduction

Scarring of the vocal folds is commonly caused by surgical injury or high impact voice use. Scarring involves a plethora of cells and chemicals whose interactions are dynamic and non-linear (Li et al, 2011). Modelling cellular interactions using computer simulations constitutes the first step towards a better understanding of the wound healing process and development of patient specific treatments of vocal fold injuries.

In this project, the dynamics of vocal fold wound healing were simulated using an Agent Based Model (ABM). As our understanding of the wound healing process becomes more detailed, the models grow in complexity and spatial refinement. This significantly increases the computational costs and performance time of the simulations. It was hypothesized that implementation of agent rules would be better suited for Graphics Processing Unit (GPU) based platforms.

Computational Methods

The ABM model was implemented on a central processing unit (CPU) and graphics processing unit (GPU) to quantify computational speedups gained from the GPUs parallel structure. The CPU-ABM was implemented in C programming language. The GPU-ABM was programmed using NVIDIA's Compute Unified Device Architecture (CUDA) C programming interface on a NIVIDIA Tesla C2070 GPU card.

The target ABM consists of 16 agents essential to vocal fold inflammation and repair: (1) platelets, (2) neutrophils, macrophages and fibroblasts, (3) transforming growth factor [TGF]-β1, basic fibroblast growth factor [bFGF]), (4) collagenase (MMP-8), (5) collagen type I, elastin and HA and (6) a damage associated molecular pattern (DAMP) variable (7) inflammatory mediators (IL-1β, IL-6, IL-8, IL-10, TNF-α and MMP-8). At each time step, the state and behavior of each agent is governed by a set of stochastic computational rules validated against empirical human biomarker profiles.

The simplified model used for benchmarking purposes includes 8 chemicals in diffusion (Table 1) and one cell type (fibroblast), acting in response to changing chemical gradients. Fibroblasts are the most important and abundant cells found in the vocal folds for wound healing. The memory access pattern and float point operations performed to replicate the fibroblast's cellular behaviour is representative that of other cells, but not of overall tissue inflammation.

Substances	Cell Sources	Functions in Wound Healing Used in the ABM
TGF-b1	Platelets, Macrophages, Fibroblasts	Chemotactic to fibroblasts; Inhibit expression of TNF-a in fibroblasts; Activate resting fibroblasts; Mitogenic to fibroblasts (proliferation)
bFGF	Macrophages, Fibroblasts	Mitogenic to fibroblasts (proliferation); Stimulate fibroblast migration
TNF-a	Neutrophils, Macrophages, Fibroblasts	Stimulate expression of TGF-b in fibroblasts; Mitogenic to fibroblasts (proliferation); Stimulate expression of IL-6 in fibroblasts
IL-1b	Platelets, Macrophages	Mitogenic to fibroblasts (proliferation)
IL-6	Macrophages, Fibroblasts	Stimulate collagen synthesis in fibroblasts
IL-8	Macrophages, Fibroblasts	Inhibit collagen synthesis in fibroblasts
IL-10	Macrophages	Inhibit expression of TNF-a in fibroblasts; Inhibit expressions of IL-6 and IL-8 in fibroblasts; Stimulate expression of TGF-b in fibroblasts; I

Table 1 – Chemicals and associated Fibroblast agent rule (Li et al , 2011) modified for benchmarking

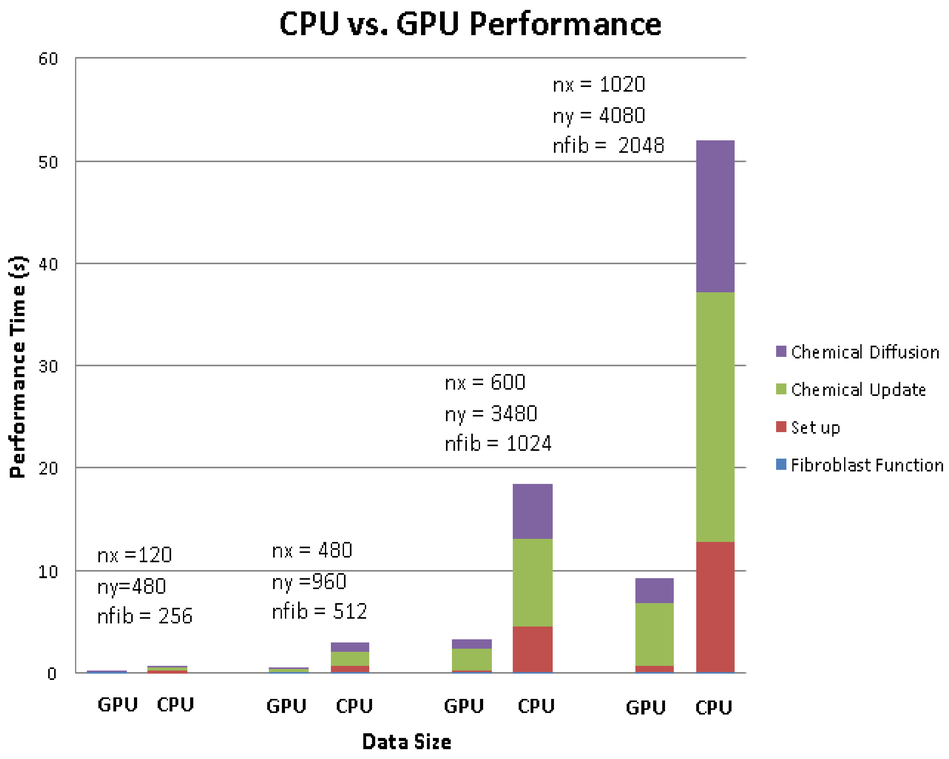

Figure 1. CPU vs. GPU Performance

	nx,*ny, nfib	Fibroblast (s)	Set up (s)	Chemical Update (s)	Chemical Diffusion (s)	Total (s)	Ratio of GPU to CPU Performance
GPU	120 * 480, 256	0.04	0.015	0.091	0.041	0.19	4.05
CPU		0	0.18	0.4	0.19	0.77	
GPU	480* 960, 512	0.041	0.044	0.34	0.15	0.58	5.19
CPU		0.01	0.72	1.37	0.91	3.01	
GPU	600*3480,1024	0.039	0.24	2.13	0.89	3.3	5.62
CPU		0.03	4.43	8.6	5.47	18.53	
GPU	1020*4080, 2048	0.037	0.65	6.1	2.47	9.25	5.61
CPU		0.06	12.76	24.32	14.78	51.92	
							average =5.12

Table 2. CPU vs. GPU Performance

References

Li, N. Y. K., Vodovotz, Y., Kim, K. H., Mi, Q., Hebda, P. A., & Verdolini Abbott, K. (2011). Biosimulation of acute phonotrauma: An extended model. Laryngoscope, 121(11), 2418-2428

Li, N. Y. K., Heris, H. K., & Mongeau, L. Current understanding and future direction of vocal fold mechanobiology, 2013. Cytology and Molecular Biology.

NVIDIA Corporation. CUDA C Best Practices Guide, 2012. http://docs.nvidia.com/cuda/cuda-c-best-practices-guide/index.html, May 25, 2013.

NVIDIA Corporation. CUDA C Programming Guide, 2012. http://docs.nvidia.com/cuda/cuda-c-best-practices-guide/index.html, May 26, 2013

Results of a Simplified Vocal Fold ABM

Model simulations of the simplified CPU-ABM and GPU-ABM were conducted for benchmarking and validation purposes. In each run, chemical update, chemical diffusion and fibroblast cell function were executed consecutively in that order. The performance times were calculated from an average of 50 runs. CPU-ABM and GPU-ABM performance times were compared for different spatial dimensions (nx, ny) and cell populations (nfib). The cell populations chosen represent a realistic cell density of fibroblasts inside vocal folds. The raw output data was fed into ParaView, an open source program which was used to visualize numerical data.

Figure 1 compares the performance times of simplified CPU-ABM and GPU-ABM for different spatial dimensions and cell populations. Fibroblast activity was the least computationally expensive function for all cases. The performance of the simplified GPU-ABM demonstrated significant speedup compared to the CPU implementation, for all test cases. On average, GPU-ABM provided a five-fold speedup on computation time compared to CPU-ABM.

Discussion

From the results, the GPU-ABM was consistently faster than the CPU implementation for different spatial resolutions and cell populations. The relatively short time required by both CPU-ABM and GPU-ABM to model cellular behaviour of fibroblasts is likely due the low cell density. The relative computation costs of simulating cellular behavior may be different in a complete model of the vocal focal folds healing process will include millions of cells and other cell types.

While the CPU implementation of fibroblast cell function showed positive linear relationship between the number of cells and performance time, the GPU implementation of the fibroblasts showed no significant change (and actually slight decrease) in performance time with respect to increase in the number of cells. One plausible explanation for this is that the GPU's low memory bandwidth can only be covered by highly parallelized data, and in the range of 256 to 2048 cells, the data parallelization is insufficient to take full advantage of the GPU. This suggests that as the number of cells continues to increase, as in the case where we expand our model to include other cell types, the GPU will continue to out-perform the CPU. A more complex and sophisticated model with millions of cells of many different types will benefit more from the GPU implementation.

Conclusion

Computational cost of ABM simulations will grow as models increases in complexity and spatial refinement. The work done validates that task parallelism on a GPU platform is a critical step in significantly reducing computation costs for this technology to be relevant in a clinical setting.

Future work includes optimization of GPU implementation on high performance clusters and expansion of the model in light of advancing achievements in empirical testing.

Acknowledgements

Work supported by NIH(NIDCD) grant R03 DC012112