GPGPU Accelerated Cardiac Electrophysiology in the Human Heart

Guillermo Vigueras[1], Ishani Roy[1], David Nordsletten[1], Andrew Cookson[1], Jack Lee[1],Pablo Lamata[2], Nicolas Smith[1]

[1] *King's College of London,* [2] *University of Oxford*

Correspondence: [1] Department of Biomedical Engineering, The Rayne Institute, 4th Floor
Lambeth Wing, St. Thomas' Hospital, Lambeth Palace Rd., SE1 7EH London, UK

The modelling of electrical activity in the heart is of significant medical and scientific interest, providing an improved understanding of the related biophysical phenomena important for diagnosing and treatmenting cardiac diseases. Cardiac activation is typically simulated by solving partial differential equations coupled to a system of ordinary differential equations describing the electrical behavior of the cell membrane. However, the numerical solution is computationally demanding due to the fine spatiotemporal resolution required by the governing equations. This demand is further increased when dealing with human size geometries, making simulation within clinical timescales challenging. In recent years, Graphics Processing Units (GPUs) have been adapted to be used in a more general context arising the term General Purpose GPU (GP-GPU) computing. These devices are endowed with hundreds of computing cores as well as fast on-chip memories resulting in a platform with a low price/computational power ratio. These capabilities make the GPU a promising platform for simulating the electrical activity in the human heart.

Different works have proposed the use of GPUs for accelerating the simulation of the electrical activity in the heart. Some of them propose the implementation of the ODE solver on the GPU and evaluate the approach using 2D tissues [2]. Other work proposes the acceleration of the ODE solver through the application of GPUs for simulating the electrical activity, most recently demonstrated in small mammalian hearts [4]. Still a further extension solves the system of PDEs and ODEs present in the Monodomain model using the GPU [1]. However, the authors use a simple cell model which allows the use of single precision without significantly affecting accuracy and evaluate the approach using small 2D tissues (~400K nodes). It is in this context that we believe that the benefits provided by GPUs for simulating the electrical activity in the human heart have not been yet exploited.

Excitation in cardiac tissue can be modelled by the Monodomain model which consists of a diffusion equation coupled with a model of cellular activation. This equation is integrated to find the electrical wave propagation during the cardiac cycle. The Monodomain model can be described in the following form:

$$\frac{\partial V_m}{\partial t} = \nabla \cdot (D \nabla V_m) - \frac{1}{C_m}(I_{ion}(V_m, v) + I_{ext}) \qquad (1)$$

$$\frac{dv}{dt} = f(t, V_m, v) \qquad (2)$$

where *Vm* is the membrane potential, $D = \sigma / \chi Cm$ is the diffusion coefficient related to the cardiac structure and conductivity and distribution of gap junctions between cells. *Cm* is the membrane capacitance and σ is the conductivity. χ is cell's surface area to volume ratio. v is the vector of variables defining the cell model, in our case the tenTusscher and Panfilov 2006 model [6]. *Iion* is the total ionic current which is a function of the voltage *Vm*, the gating variables and the ion concentrations.

The Monodomain model was solved using the finite element method (FEM) within CHeart [3], an efficient parallel software platform for multi-physics cardiac applications. Based on Fortran 2003/2008, CHeart enables the efficient simulation of Monodomain by discretizing equation (1) and (2) into an algebraic matrix system requiring no further FEM element-level evaluations. This enables the core ODE and linear system to be efficiently computed and ported to the GPU.

In this paper we propose the acceleration of Monodomain simulations in the human heart by implementing some parts of the CHeart software onto the GPU. As other works do, we migrate the ODE solver to the GPU. However, complex cell models are required for providing realistic results, which forces the use of double precision computations [2]. In addition, we propose the solution of the system of PDEs through the GPU. Due to the spatiotemporal resolution required, the number of nodes in the human mesh can increase up to tens of millions, for that reason we propose the use of multiple GPUs for accelerating the Monodomain simulations. However, due to space limitations we only show in this abstract performance results for solving the system of ODEs.

Figure 1 shows the propagation of the membrane potential in a left ventricular mesh at different stages when using the GPU. Figure 2 shows the performance running different monodomain simulations with different meshes. In order to check the performance when the mesh resolution decreases, we have used the same benchmark mesh as in a previous study [5]. In this case, we have used resolutions 0.2 mm. (~58K nodes), 0.1 mm. (~443K nodes) and 0.05 mm. (~3500K nodes). In addition, we have obtained results using a more realistic mesh. For this purpose, we have used a human left ventricular (LV) mesh containing ~2500K nodes. We have run the simulations for 1000 ms. setting the time step to 0.01 ms. shown previously to be sufficient for numerical convergence [5].

The different implementations have been compared using different platform configurations. For the CPU simulations we have used a machine with an 8 core AMD Opteron @ 2.0 GHz and 128 GB of RAM. The theoretical performance of this platform per core is 13.75 GFlops. For our GPU simulations we have used a Tesla C1060 processor with 240 SPs and 4GB of device memory for solving the system of ODEs. The theoretical performance of the GPU is 77.76 GFlops. In order to do a fair comparison between the CPU and GPU versions we have used up to 6 cores (82.5 GFlops of theoretical performance) and compared performance provided by the single core, multi-core and GPU implementations.

The graph on the left in Figure 2 shows the total run time (in seconds) required for solving the system of ODEs for the different problem sizes considered. It can be seen that the GPU version is always faster than the single core CPU one. However, when comparing the muti-core (6 CPU) and GPU versions, the former is faster than the latter for the smallest problem size. Nevertheless, as the number or nodes increases the GPU implementation becomes significantly faster than the multi-core implementation. Given the fact that the computational power of the 6 CPU configuration is slightly higher, these results show that the GPU architecture and the way the memory hierarchy is interconnected to the array of SMs is better suited than the CPU model for executing the cell model. In this way, the GPU provides a better scalability with the problem size.

We have also measured the impact in the total run time of migrating to the GPU the cell model. For this purpose we have compared the single core and GPU implementations. The graph on the right of Figure 2 shows the average run time of one simulation step. It can be seen that differences in run time between single core and GPU increases with the problem size for the benchmark problem. However, for the LV Mesh the simulation step when using the GPU takes longer than for the benchmark with .05 refinement. This is due to the fact that solving the PDE for the LV Mesh problem takes longer since the geometry is more complex than the one in the benchmark problem. In addition, these results show that the workload alleviation provided by the GPU makes the solution of the PDE the dominant part in the simulation step.

To conclude, we have proposed the acceleration through GPUs of the simulation of the electrical activity in the human heart. To do so, we propose the migration to the GPU of some parts of a parallel software for multi-physics cardiac applications. Performance results show that the GPU architecture is well suited for cardiac simulations, providing significant speed up with respect to a parallel CPU version.

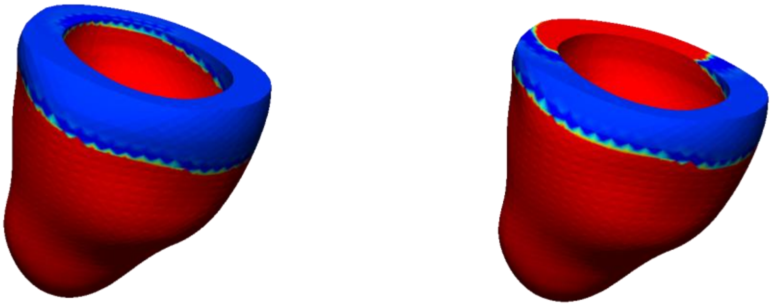

Figure 1. Visualization of membrane potential propagation for the LV Mesh at different simulation stages.

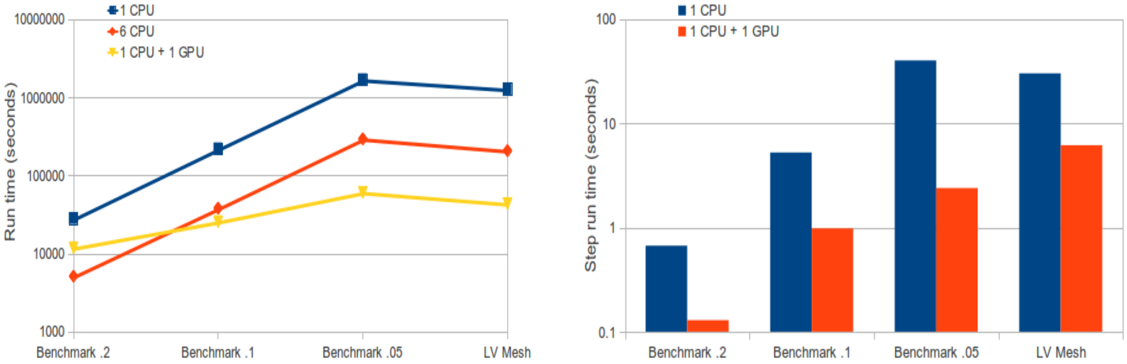

Figure 2. Left: run time (in seconds) for solving the system of ODEs for the single core, many-core and GPU implementations. Right: average run time of a simulation step for the single core and GPU versions**.**

References

1. Rocha, BM., Campos, FO., Amorim, RM., Plank, G., dos Santos, RW., Liebmann, M., Haase, G., Accelerating cardiac excitation spread simulations using graphics processing units. Concurrency and Computation: Practice and Experience. 23. 1532-0634.

2. Ezio Bartocci, Elizabeth M. Cherry, James Glimm, Radu Grosu, Scott A. Smolka, Flavio H. Fenton. Toward Realtime Simulation of Cardiac Dynamics. The 9th International Conference on Computational Methods in Systems Biology. 2011.

3. Nordsletten, D., Kay, D., Smith, N. A non-conforming monolithic finite element method for problems of coupled mechanics. Journal of Computational Physics. 229, 7571-7593.

4. Vigmond, EJ., Boyle, PM., Leon LJ., Plank G. Near-real-time simulations of bioelectric activity in small mammalian hearts using graphical processing units. 31st International Conference of the IEEE EMBS. 2009.

5. Niederer, SA. et al. Verification of cardiac tissue electrophysiology simulators using an N-version benchmark. *Phil. Trans. R. Soc. A* 13 November 2011 vol. 369 no. 1954 4331-4351.

6. Ten Tusscher, K.H., Panfilov, A.V., 2006. Cell model for efficient simulation of wave propagation in human ventricular tissue under normal and pathological conditions. Phys. Med. Biol. 51, 6141-6156.