Paweł KAPUSTA, Michał MAJCHROWICZ, Dominik SANKOWSKI, Lidia JACKOWSKA-STRUMIŁŁO, Robert BANASIAK
Politechnika Łódzka, Instytut Informatyki Stosowanej

Distributed multi-node, multi-GPU, heterogeneous system for 3D image reconstruction in Electrical Capacitance Tomography – network performance and application analysis

***Abstract**. 3D ECT provides a lot of challenging computational issues as image reconstruction requires execution of many basic operations of linear algebra, especially when the solutions are based on Finite Element Method. In order to reach real-time reconstruction a 3D ECT computational subsystem has to be able to transform capacitance data into image in fractions of seconds. By performing computations in parallel and in a distributed, heterogeneous, multi-GPU environment a significant speed-up can be achieved. Nevertheless performed tests clearly illustrate the need for developing a highly optimized distributed platform, which would mitigate existing hardware and software limitations.*

***Streszczenie.** 3D ECT zapewnia wiele złożonych problemów obliczeniowych, jako, że rekonstrukcja obrazu wymaga wykonania wielu podstawowych operacji algebry liniowej, zwłaszcza, gdy rozwiązania oparte są na Metodzie Elementów Skończonych. W celu osiągnięcia rekonstrukcji w czasie rzeczywistym system obliczeniowy musi być zdolny do przekształcania danych pomiarowych na obraz w ułamkach sekund. Poprzez wykonywanie obliczeń w sposób równoległy, z wykorzystaniem rozproszonego środowiska heterogenicznego multi-GPU można uzyskać znaczne ich przyspieszenie. Niemniej przeprowadzone badania wyraźnie pokazują potrzebę opracowania wysoce zoptymalizowanej, rozproszonej platformy, która pozwoliłaby na ominięcie istniejących ograniczeń sprzętowych i programowych. **(Rozproszony, wielowęzłowy, heterogeniczny system multi-GPU do celów rekonstrukcji obrazów 3D w elektrycznej tomografii pojemnościowej – analiza wydajności sieciowej oraz zastosowania)**.*

Keywords: parallel computations, distributed computations, Electrical Capacitance Tomography, Finite Element Method.
Słowa kluczowe: obliczenia równoległe, obliczenia rozproszone, elektryczna tomografia pojemnościowa, Metoda Elementów Skończonych.

Introduction

Electrical Capacitance Tomography is a relatively mature imaging method in industrial process tomography [4]. The ECT is performing a task of imaging of materials with a contrast in dielectric permittivity by measuring capacitance from a set of electrodes. Applications of ECT include the monitoring of oil-gas flows in pipelines, gas-solids flows in pneumatic conveying and imaging flames in combustion, gravitational flows in silo [1].

Among other non-invasive imaging techniques, ECT characterizes much higher temporal resolution than Magnetic Resonance Imaging, X-ray Computed Tomography etc. This makes ECT a good candidate for real-time imaging technique which is capable of long term monitoring on fast-varying industrial process applications.

To reach this goal 3D ECT computational subsystem should be able to transform capacitance data into image in fractions of seconds, which is really hard to achieve since typically 3D ECT tomography image can be composed of large number of elements. 3D ECT provides few challenging computational issues that have been reported in the past by many researchers [1,2,3]. This is due to the fact that most of the algorithms perform a series of complex algebraic operations on two-dimensional arrays, which contain many elements. Nonlinear three-dimensional image reconstruction in 3D capacitance tomography is therefore a complex numerical problem, saturated with linear algebra transformations that cannot be efficiently performed in real-time using classic (even multi-core chips) CPU power [6].

In this paper a distributed GPGPU approach has been considered as an efficient way to obtain a significant speed-up of 3D ECT reconstruction process. By assuming, that many of the computations can be performed in parallel using modern, fast graphics processor and by altering the algorithms time to achieve high quality image reconstruction will be shortened significantly

Computations on Graphic Processors

GPGPU (*General-Purpose computing on Graphics Processing Units*) is a technique of using graphic cards (GPUs – *Graphics Processing Unit*), which normally handles graphics rendering, for computations that are usually handled by processors (CPUs – *Central Processing Unit*).

Growing interest in GPU computations started with the inability to clock CPU above certain level, because of the limitations of silicon based transistor technology and constant demand for improvements. Any change in speed of sequential programs execution is now based on architecture improvements of the CPU rather than higher clocks, but even this approach has limitations.

Parallel programming is not a new idea, though till only recently it was reserved for high performance clusters with many processors. This changed with the introduction of many-core processors to the mainstream market. GPUs fit well in that trend, even take it to another level. Compared to CPUs, which today have maximum of 2 to 12 cores, GPUs consist, of dozens and even hundreds of smaller, simpler cores designed for high-performance calculations.

CPUs are built and designed to execute single thread no matter how unpredictable, diverse or complicated it may be, as fast as possible. For that they require additional resources such as: complicated mechanisms for predicting branches, cache memory and data prefetching. On the other hand GPUs mostly take care of data computations that are much simpler in their nature and for that reason their execution units, or cores, can be much simpler, which also mean smaller (Fig. 1). Thanks to that there can be much more of them on a single chip with numbers reaching dozens or even hundreds. This translates into much higher number of operations per second than what can be achieved on traditional CPUs. Thanks to this GPUs can run hundreds even thousands of threads at once, compared to only few on CPU.

Distributed computations

The local GPGPU approach can be adapted to achieve a significant gain in computational power. This solution however has a very important drawback. It is easy to increase the computation power by equipping the computer with multiple GPUs but only up to the point. In order to overcome this obstacle it is necessary to use distributed platform with multiple machines.

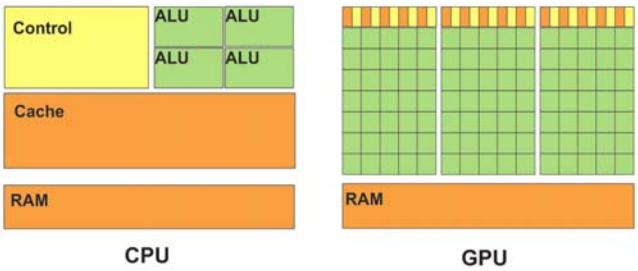

Fig.1 CPU and GPU architecture

The research conducted while analyzing ECT algorithms [8] has also shown that, although dynamic development of GPU computational capabilities and its recent application for image reconstruction in ECT has significantly improved calculations time, in modern systems a single GPU is not enough to perform many tasks [9]. As a result multiple GPUs have to be used to accelerate calculations [8]. For that reason the proposers are trying to develop a set of algorithms for a system that will divide data and distribute it across multiple GPUs and even CPUs.

This approach requires partial (and in some cases even full) reimplementation of existing algorithms. This is the case because memory isn't shared between these devices and all calculations need to be performed in distributed memory environment. Moreover the computational power of single computer (equipped with many GPUs) is also limited.

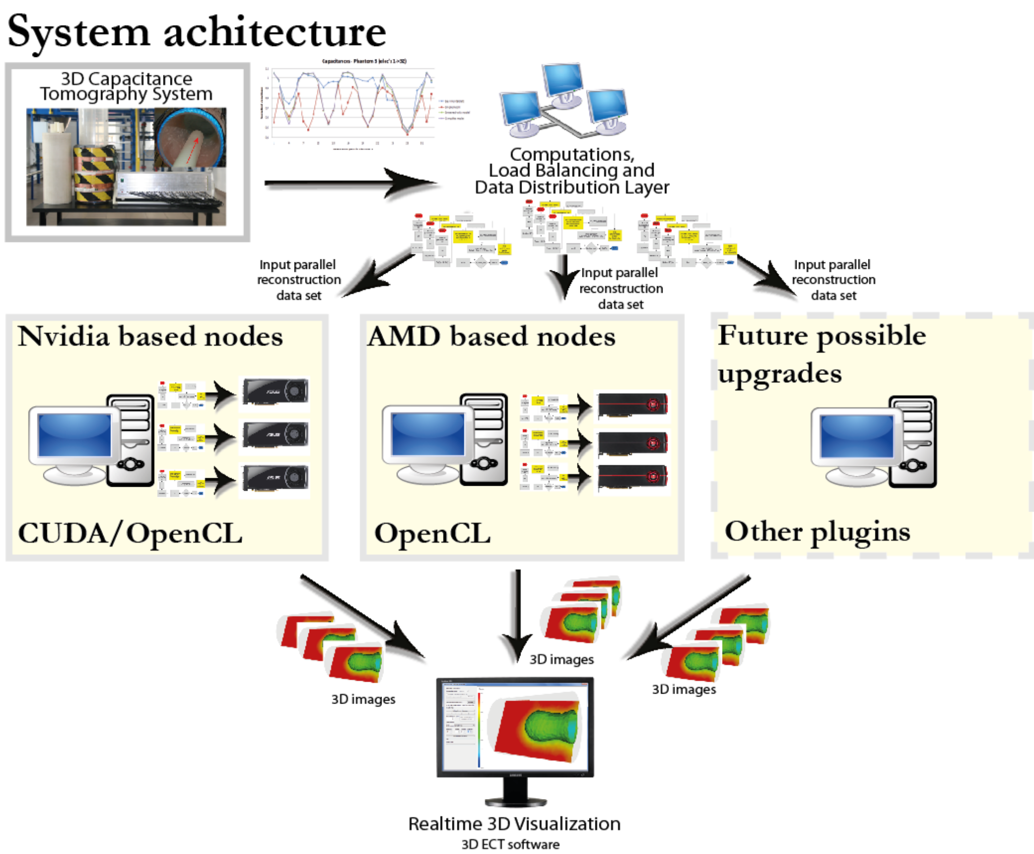

Fig. 2. Distributed computational system for accelerating image reconstruction in Electrical Capacitance Tomography

As a consequence the only available solution is a layer that will allow usage of multiple computers with fast GPUs and performing calculations across network connection. Developed system will be designed to fully exploit computational power of all devices that all nodes are equipped with. Such architecture will be very scalable and will make it possible to easily increase computational power of used system by adding next network nodes.

Developed solution will provide linear algebra operations Developed solution will provide linear algebra operations as series of APIs (Application Programming Interfaces). Such approach does create a lot of challenges when designing the architecture of the system, but will eventually allow other researchers to easily use this system to speed up calculations in their existing projects, thus enabling them to develop and test algorithms much faster, than it was previously possible.

Developed system (Fig. 2) will not only allow significant acceleration of image reconstruction time in ECT but it will also be very flexible and its application for performing calculations of different type will not require much work. Especially for equations that are based on basic linear algebra operations (addition, subtraction, multiplication and division), where only utilization of provided library will be necessary. In other cases it will be sufficient to implement new plugin (or implement new operations for existing one) using provided SDK.

Distributed Image Reconstruction

In Electrical Capacitance Tomography image is reconstructed by performing calculations on large quantities of data acquired from sensors. During our research on the nature of distributed systems [9,10] it was concluded that in order to achieve significant acceleration of image reconstruction in Electrical Capacitance Tomography a new distributed system has to be developed. This is caused by huge overhead introduced by already existing solutions (for example Xgrid [8] which are not designed to efficiently perform calculations that require frequent data exchange.

Authors are proposing an innovative approach to multi-GPU image reconstruction in 3D ECT. Instead of dividing one task (image frame) between many GPUs, each one receives its own frame to compute. This approach does not reduce the time necessary for calculating single frame, but by using synchronization and load balancing algorithms results can be spread evenly in time and a smooth frame rate can be achieved. It is however well suited for computations in a highly distributed environment.

This approach will, by design, introduce a delay equal to the time necessary to reconstruct one image frame on the slowest GPU available and by that will introduce "buffering" time, but can be effectively used when response time is not critical, i.e. visualizing previously gathered data or during algorithm testing.

For testing purposes Landweber image reconstruction algorithm was used, which is given by following equation [5]:

(1) $$\boldsymbol{\varepsilon}_{k+1} = \boldsymbol{\varepsilon}_k - \alpha * \mathbf{S}^T * (\mathbf{S} * \boldsymbol{\varepsilon}_k - \mathbf{C}_m)$$

where: $\boldsymbol{\varepsilon}_{k+1}$- image obtained during current iteration, $\boldsymbol{\varepsilon}_k$ - image from the previous iteration, α - convergence factor, $\boldsymbol{S}$ - sensitivity matrix, $\boldsymbol{C}_m$ - capacity measurements vector.

Local tests were performer on a computer with Nvidia Tesla S1070-400 server and a Tesla C2070 card, all working in parallel (total of 5 GPUs). Distributed tests were performed by connecting with a second computer, equipped with 4 AMD Radeon 5970 cards (total of 8 GPUs) over 10 Gb/s LAN and adding two cards at a time to the computational resources pool. The results are gathered in table 5.

Two versions of network distribution layer are taken into consideration, of which first one is based on existing Xgrid platform and. The results of the second version, that negates Xgrid shortcomings, are projected based on previous study and experiments of Xgrid protocol overhead [9,10].

Table 1. Distributed image reconstruction

	Xgrid [ms]	Gain over 5 GPUs	Optimized [ms]	Gain over 5 GPUs
5 GPUs	4452	-	4452	-
5+1 GPUs	7303	- 39%	6275	- 29%
5+2 GPUs	6210	-28%	5230	-15%
5+4 GPUs	4868	-9%	4110	8%
5+8 GPUs	3318	34%	2570	73%

It can be seen from the results in Table 2, as well as graph in Figure 3, that Xgrid protocol overhead is a serious limiting factor of a distributed image reconstruction algorithm. It is therefore necessary to develop a more optimized solution to utilize fully the potential of the hardware.

Finite Elements Method

Nonlinear three-dimensional image reconstruction in 3D capacitance tomography is a complex numerical problem, saturated with linear algebra transformations. During this iterative calculation process a set of parameters is determined, that is necessary for proper reconstruction of three-dimensional tomographic image optimization.

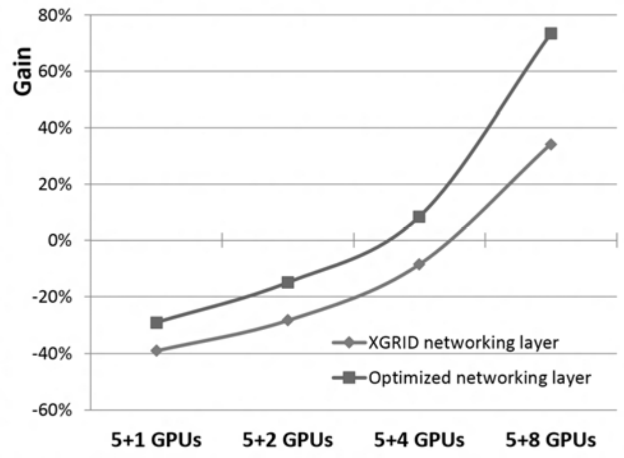

Fig.3 Heterogeneous systems using OpenCL

One of the three key stages of the iterative process of reconstruction is a forward problem involving setting up a simulated vector based on a given spatial distribution of dielectric permittivity.

The accuracy of the forward problem solution has a significant impact on the quality and speed of image reconstruction, and depends on the method of its determination. Most often forward problem is determined numerically using the Finite Element Method (FEM) based on a numerical model of a capacitance sensor (Fig 4.).

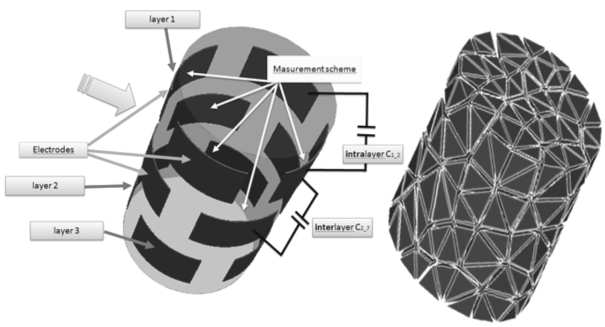

Fig. 4 FEM mesh

The authors have focused primarily on developing methods for accelerating the calculations using algorithms developed for performing calculations on sparse matrices (CULA library, CUSP). This made possible the development of (as a set of functions and procedures) proprietary parallel computing algorithms dedicated to specific processing of tomographic data. Developed methods allow reconstructing three-dimensional images by using fast methods of solving sparse matrix equations (AMG method - Algebraic Multi Grid [12,13], the Jacobi method [14], the Conjugate Gradient algorithm [14]), which are processed on graphic processors.

Main idea of the developed algorithm is to obtaining the solution (electric field distribution) given in the form of equation [9]:

(3) $$\varphi = \mathbf{Y}^{-1} * \mathbf{F}$$

where: φ – is a sought distribution of the electric field - represented by the spatial distribution of nodal potential - partial solution of the forward problem in capacitance tomography; $\boldsymbol{Y}$ – is a transformation matrix, built according to the geometric dependencies of sensor model mesh and Neumann boundary conditions; $\boldsymbol{F}$ – is the extortion vector, defining the given Dirichlet boundary conditions.

Prepared set of tomographic measurement data processing functions, acting as a programming interface, is very flexible, which manifests, inter alia, in the choice of both the method of solving the forward problem, as well as computational technology. This demonstrates the fully hybrid nature of the developed solution. It is also worth mentioning, that each of the functions presented previously can be used independently.

The developed solution was successfully verified using the high-performance computer built during work on the MNiSW project. Its basic parameters are: 4-core Intel i7 930 2.8 GHz CPU, 12 GB RAM, 8 AMD graphics processors and, connected using external PCI-Express bus, NVIDIA Tesla GPU server, model S1070-400.

In order to verify the effectiveness of developed algorithms 10 iterations of nonlinear 3D image reconstruction algorithm were executed. For the two computed models of capacitance sensors, developed for the MNiSW project for identification of flow structures, a GPU implementation of Finite Element Method was employed. Meshes used consist of 110878 and 87898 tetrahedrons. After the initial preparation of the system three tests were performed and the speed of reconstruction was determined as the average time of calculation, which was presented in Table 2.

Table 2. Nonlinear 3D image reconstruction times

FEM mesh density [tetrahedrons]	CPU computations time [s]	GPU computations time [s]	Speed-up
87898	479.25	45.65	10.50
110878	667.30	52.10	12.80

All the tests have been performed locally, however developed solution can be potentially used in its full form as a plugin for the developed distributed environment to achieve much higher speed-ups.

Conclusion

Developed solutions and algorithms show the potential of using a GPGPU approach, especially in a distributed, heterogeneous environment, for the purpose of image reconstruction in Electrical Capacitance Tomography. Multi-GPU and distributed Multi-GPU solutions can reduce reconstruction time to only a fraction of what was possible on pure CPU systems.

However the results clearly show, that a distributed approach requires development of a fully optimized networking and data distribution platform to be able to fully utilize the potential of the hardware. In order to achieve satisfactory gains in 3D image reconstruction in Electrical Capacitance Tomography it has to take into account specific nature of computations in Multi-GPU environments, especially when combining it with a massively distributed memory model. It also has to reduce the negative impact of network communication and other factors that caused such instability when using Xgrid platform.

REFERENCES

[1] Wajman R., Banasiak R., Mazurkiewicz Ł., Dyakowski T., Sankowski D., Spatial imaging with 3D capacitance measurements, *Measurement Science and Technology*, vol. 17 (2006), no. 8, 2113-2118.

[2] Soleimani M., Three-dimensional electrical capacitance tomography imaging, *Insight, Non-Destructive Testing and Condition Monitoring*, vol. 48 (2006), No. 10, 613-617

[3] Warsito W., Fan L-S., Development of 3-Dimensional Electrical Capacitance Tomography Based on Neural Network Multi-criterion Optimization Image Reconstruction, *proc. of 3rd World Congress on Industrial Process Tomography (Banff)*, (2003), 942-947

[4] Lionheart B., Reconstruction algorithms for permittivity and conductivity imaging, *Proceedings of 2nd World Congress on Industrial Process Tomography*, (2001), 4–11

[5] Yang W. Q., Peng L., Image reconstruction algorithms for electrical capacitance tomography, *Measurement Science and Technology IOP Journal,* 14 (2003)

[6] Banasiak R., Wajman R., Soleimani M., An efficient nodal jacobian method for 3d electrical capacitance image reconstruction, *Insight Non-Destructive Testing and Condition Monitoring*, vol. 51 (2009), no. 1, 36–38

[7] Banasiak R., Wajman R., Sankowski D., Soleimani M., Three-dimensional nonlinear inversion of electrical capacitance tomography data using a complete sensor model, *Progress In Electromagnetics Research PIER*, vol. 100 (2010), 219-234

[8] Majchrowicz M., Kapusta P., Sankowski D., Accelerating image reconstruction in electrical capacitance tomography using OpenCL technology in heterogeneous systems, *XVII International Conference on Information Technology Systems. Theory, Design, Implementations, Applications*, (2010)

[9] Kapusta P., Majchrowicz M., Banasiak R., Trudności w rekonstrukcji obrazu w czasie rzeczywistym na podstawie danych uzyskanych z pomocą elektrycznej tomografii pojemnościowej w systemach równoległych i rozproszonych, *Metody wytwarzania i zastosowania systemów czasu rzeczywistego*, (2010), 309 – 498

[10] Kapusta P., Majchrowicz M., Accelerating Image reconstruction algorithms in Electrical Capacitance Tomography using Multi-GPU system, *Advanced Numerical Modelling*, (2011), 47 - 49

[11] Sikora J., Podstawy Metody *Elementów Skończonych: Zagadnienia potencjalne, Wydawnictwo* IEL, (2008)

[12] Geveler M., Ribbrock D., Göddeke D., Zajac P., Turek S., Efficient Finite Element Geometric Multigrid Solvers for Unstructured Grids on GPUs, *Proceedings of the The Second International Conference on Parallel, Distributed, Grid and Cloud Computing for Engineering (PARENG 2011)*, Ajaccio, Corsica, France, (2011)

[13] Ruge J.W., Stüben K., Algebraic Multigrid, *Multigrid Methods (Frontier in Applied Mathematics)*, Society for industrial Mathematics, (1994), 73 – 130

[14] Press W.H., Teukolsky S.A., Vetterling W.T, Flannery B.P, *Numerical recipes: The Art of Scientific Computing, Third Edition in C++*, Cambridge University Press, (2007)

***Authors**: mgr inż. Paweł Kapusta, e-mail: pkapust@kis.p.lodz.pl; mgr inż. Michał Majchrowicz, e-mail: mmajchr@kis.p.lodz.pl; prof. dr hab. inż. Dominik Sankowski, e-mail: dsan@kis.p.lodz.pl; dr hab. inż. Lidia Jackowska-Strumiłło, prof. PŁ, email: lidia_js@kis.p.lodz.pl; dr inż. Robert Banasiak, Politechnika Łódzka, Instytut Informatyki Stosowanej, ul. Stefanowskiego 18/22, 90-924 Łódź, e-mail: rbanasi@kis.p.lodz.pl*