

Acceleration of 3D ECT image reconstruction in heterogeneous, multi-GPU, multi-node distributed system

Michał Majchrowicz*, Paweł Kapusta†, Lidia Jackowska-Strumiłło‡ and Dominik Sankowski§

Lodz University of Technology
Institute of Applied Computer Science
ul. Stefanowskiego 18/22, Łódź, Poland

* Email: mmajchr@iis.p.lodz.pl † Email: pawel.kapusta@p.lodz.pl ‡ Email: lidia_js@iis.p.lodz.pl

§ Email: dsan@iis.p.lodz.pl

Abstract—Electrical Capacitance Tomography (ECT) is an effective and non-invasive visualization technique, which is used in many industrial applications. Unfortunately, image reconstruction in 3D ECT is a complex computational task requiring operations on large size matrices. In this paper, a new approach to 3D ECT image reconstruction is proposed. A new heterogeneous, multi-GPU, multi-node distributed system has been developed, with a framework for parallel computing and a special plug-in dedicated to ECT.

I. INTRODUCTION

ELECTRICAL Capacitance Tomography (ECT) is a measurement technique that can be used for non-invasive monitoring of industrial processes in 2D [7], 3D [1] and even 4D dynamic mode. ECT is performing the task of imaging of materials with a contrast in dielectric permittivity by measuring capacitance from a set of electrodes placed around the investigated object.

In order to achieve a high quality of 3D image, complex reconstruction algorithms performing many matrix calculations have to be applied. Therefore different solutions accelerating these calculation have been reported in the past by the Authors[8][14], especially these dealing with sparse matrices and Finite Elements Method [9] as well as neural networks approach [5][6] and even fuzzy logic [21].

In this work we propose a novel heterogeneous, multi-GPU (Graphics Processing Unit), multi-node distributed system, with a framework for parallel computing and a special plug-in dedicated to ECT. The system features and its efficiency have been compared to the previously developed distributed system based on the Xgrid platform.

A. Image reconstruction in ECT

The scheme of image synthesis in Electrical Capacitance Tomography is called image reconstruction. It is based on solving the so called inverse problem, in which the spatial distribution of electric permittivity from the measured values of capacitance C is approximated [1] [16].

Image reconstruction using deterministic methods requires execution of a large number of basic operations of linear

algebra, such as transposition, multiplication, addition and subtraction [10][15]. Matrix calculations for a large number of elements is characterized by a high computational load. Matrix multiplication is a key operation in ECT imaging and therefore many researchers decided even to build a custom hardware for this purpose.

The LBP algorithm is one the most used reconstruction algorithms, even though it is characterized by low spatial resolution. Nevertheless it is not as computationally complex as other solutions. Moreover there is still active research on improving it's characteristics [17]. It is based on the following equation [3]:

$$\varepsilon = \mathbf{S}\mathbf{C}_m \quad (1)$$

where:

ε - electric permittivity vector (output image),

\mathbf{S} - sensitivity matrix,

\mathbf{C}_m - capacitance measurements vector.

The Landweber algorithm is based on the following iterative equation:

$$\varepsilon_{k+1} = \varepsilon_k - \alpha \mathbf{S}^T (\mathbf{S}\varepsilon_k - \mathbf{C}_m) \quad (2)$$

where:

ε_{k+1} - image obtained in current iteration,

ε_k - image from the previous iteration,

α - convergence factor (scalar),

\mathbf{S}^T - sensitivity matrix, transposed,

\mathbf{S} - sensitivity matrix,

\mathbf{C}_m - capacitance measurements vector.

In the case of the Landweber algorithm each iteration improves the overall quality of the output image. As a result acceleration of image reconstruction process is a very important issue. Nevertheless, due to its nature it is necessary to exchange the data (ε_{k+1}) in every iteration.

II. DESIGN ASSUMPTIONS

As a result of the earlier performed studies [8][13] the Authors have developed a new distributed system dedicated to ECT computations. It is specially designed to accelerate matrix computations that are a crucial part of reconstruction

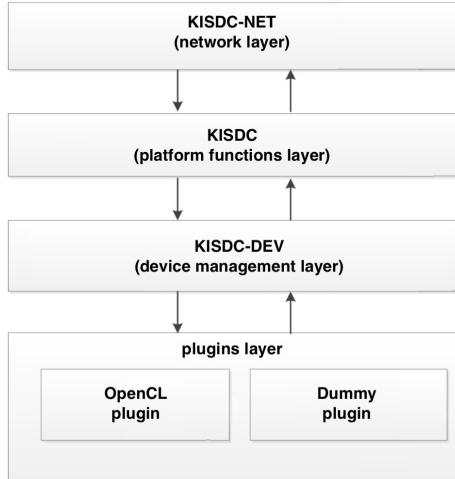


Fig. 1. Activity Diagram for performing calculations using the KISDC platform

algorithms used in ECT [9]. The earlier developed solution was based on the Xgrid platform, used as a network layer. However, the analysis of this system showed the limitations of this solution, and the main conclusion from the previous research [14] was, that the new software for the system should be developed.

A special framework was designed and built that provides software tools needed both for the system architecture expansion and new algorithms development and implementation in a distributed heterogeneous environment. The proposed approach allows for a greater flexibility of the developed solutions, provides tools for their easy testing and enables further acceleration of ECT image reconstruction.

The framework was designed to ensure an efficient use of the computing power of all the devices in which the nodes are equipped. This architecture is scalable and allows users

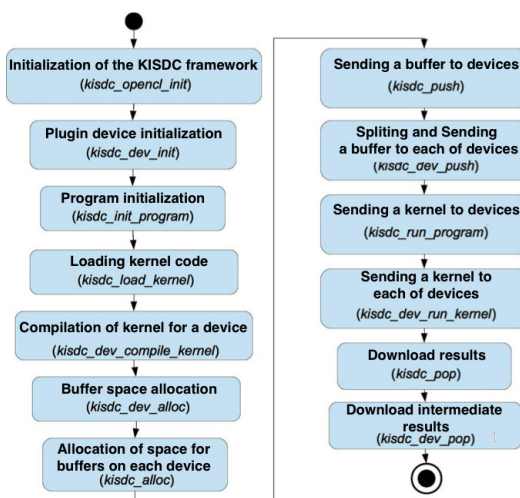


Fig. 2. Activity Diagram for performing calculations using the KISDC platform

to expand the computing power of the system by adding more nodes. The above assumptions pose many challenges in the architecture of the system itself, but their application makes it straightforward to use the environment to speed up computations in existing projects, thus testing and developing new distributed algorithms is much faster.

The system was designed as a modular, layered architecture (Fig. 1). This approach allows limiting the dependencies between the individual modules. Moreover, thanks to this architecture, it is possible to abstract the compute devices using KISDC-DEV module, that hides the type of the hardware from the user and makes all the algorithm written using the provided Application Programming Interface (APIs) hardware-agnostic

Expansion of the computing power of the system is possible through the use of "plug-in" architecture (by adding support for new devices, such as FPGAs). The basic operations of linear algebra were implemented in the system as a set of functions in the form of an API.

```

err = clGetDeviceIDs(platform, GPU ? CL_DEVICE_TYPE_GPU : CL_DEVICE_TYPE_CPU,
    numberOfDevices, &device_id, NULL);
context = clCreateContext(0, 1, &device_id, NULL, NULL, &err);
commands = clCreateCommandQueue(context, device_id, 0, &err);
program = clCreateProgramWithSource(context, 1, (const char **) &kernelSource,
    NULL, &err);
err = clBuildProgram(program, 0, NULL, NULL, NULL, NULL);
kernel = clCreateKernel(program, "matrixmul", &err);

A = clCreateBuffer(context, CL_MEM_READ_ONLY, sizeof(float) * count1, NULL, NULL);
B = clCreateBuffer(context, CL_MEM_READ_ONLY, sizeof(float) * count2, NULL, NULL);
C = clCreateBuffer(context, CL_MEM_WRITE_ONLY, sizeof(float) * count2, NULL, NULL);
err = clEnqueueWriteBuffer(commands, A, CL_TRUE, 0, sizeof(float) * count1, data, 0,
    NULL, NULL);
err = clEnqueueWriteBuffer(commands, B, CL_TRUE, 0, sizeof(float) * count2, data, 0,
    NULL, NULL);

err = clSetKernelArg(kernel, 0, sizeof(cl_mem), &A);
err |= clSetKernelArg(kernel, 1, sizeof(cl_mem), &B);
err |= clSetKernelArg(kernel, 2, sizeof(cl_mem), &C);

err = clGetKernelWorkGroupInfo(kernel, device_id, CL_KERNEL_WORK_GROUP_SIZE, sizeof(
    local), &local, NULL);

global = count1;
err = clEnqueueNDRangeKernel(commands, kernel, 1, NULL, &global, &local, 0, NULL,
    NULL);

clFinish(commands);
err = clEnqueueReadBuffer(commands, output, CL_TRUE, 0, sizeof(float) * count3,
    results, 0, NULL, NULL);

```

Fig. 3. Simplified code fragment that multiplies the matrix using the OpenCL libraries and single GPU

The use of a heterogeneous system for distributed computing in ECT required the implementation of a series of algorithms without which the proposed system would not work properly. The most important of these are as follows: division of matrices between nodes, basic operations of linear algebra (transposition, addition, subtraction, multiplication), data transfer between nodes, planning and division of tasks, support for heterogeneous devices, support for calculations using graphics cards, supports modern multi-core processors as a set of devices and a possibility to extend existing solutions with pseudo inheritance from implemented layouts.

A. Application Programming Interface

An important aspect of the designed environment is also the API, which greatly simplifies the performance of matrix calculations in a distributed heterogeneous environment. Even for a single computer configuration, the KISDC framework makes it possible to significantly simplify the code (Fig. 4) in

comparison to amount of code required when using another solution, such as for example OpenCL (Fig. 3).

```

pKISDC *KISDC;
KISDC = kisdcd_openccl_init();

KISDC->DEV = KISDC->kisdcd_dev_init();
KISDC->dummyProgram = KISDC->kisdcd_init_program(KISDC, kernel);

A = KISDC->kisdcd_alloc(KISDC, SIZE_A_1, SIZE_A_2, 1, 0);
B = KISDC->kisdcd_alloc(KISDC, SIZE_B_1, SIZE_B_2, 1, 0);
C = KISDC->kisdcd_alloc(KISDC, SIZE_C_1, SIZE_C_2, 1, 0);

KISDC->kisdcd_push(KISDC, A, Abuf);
KISDC->kisdcd_push(KISDC, B, Bbuf);

paramsArray = (pMemBuf**)malloc(5*sizeof(pMemBuf*));

paramsArray[0] = A;
paramsArray[1] = B;

KISDC->dummyProgram->args = paramsArray;
KISDC->dummyProgram->argc = 2;
KISDC->dummyProgram->out = C;

KISDC->kisdcd_run_program(KISDC, KISDC->dummyProgram);
Cbuf = KISDC->kisdcd_pop(KISDC, C);
    
```

Fig. 4. A fragment of code that multiplies the matrix using the KISDC platform and multiple GPU

The code examples listed in Fig. 4 and Fig. 3 involve a very simple scenario of using GPUs to calculate the product of two matrices (for example in the LBP algorithm). In the case of the algorithms of a higher complexity, the difference in the code sizes for both solutions will be larger and the KISDC implementation will have even a bigger advantage over OpenCL.

Moreover, if instead of one, three or more GPUs are used for computations, a significant changes in the code must be done in each case, when using the CUDA [11] or OpenCL library [19]. Whereas, with the KISDC framework, no changes are necessary. However, this small number of lines written by the programmer corresponds to hundreds of lines of code within the KISDC system.

The KISDC architecture simplifies also performance tests of the developed image reconstruction algorithms in Electrical Capacitance Tomography in various hardware configurations and to implement these algorithms in a distributed system. As shown in the activity diagram (Fig. 2), the process of

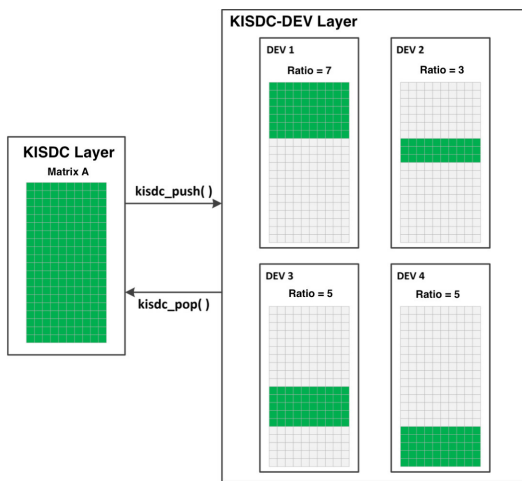


Fig. 5. Data distribution using KISDC platform

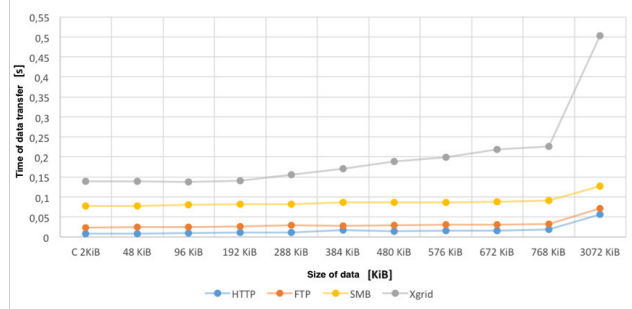


Fig. 6. Comparison of average data transmission times for different network protocols

performing calculations using the KISDC system allows for much more flexibility in the number and type of devices used. This was possible by adopting significant flexibility in the data distribution layer (Fig.5).

III. TEST RESULTS

In the previously built distributed system a ready Xgrid platform was applied as a network layer [14]. In the current work the author’s KISDC system with KISDC-NET network layer was designed and implemented. While designing the KISDC-NET layer, the existing network protocols were applied and tested in advance in order to choose the best solution.

The network characteristics of the previously developed solution based on the Xgrid system was compared with the new system using other data distribution protocols: HTTP (Hypertext Transfer Protocol), FTP (File Transfer Protocol) and SMB (Server Message Block). On the Basis of the obtained results (see Fig. 6) HTTP protocol has been selected as the best one for the KISDC-NET.

Both distributed systems: the one based on the Xgrid platform and the KISDC have been extensively tested and compared. In both cases the hardware was identical, consisting of two nodes of high computing power, both using 8 thread Intel i7 930 CPUs and Nvidia GPUs (Tesla S1070 + Tesla

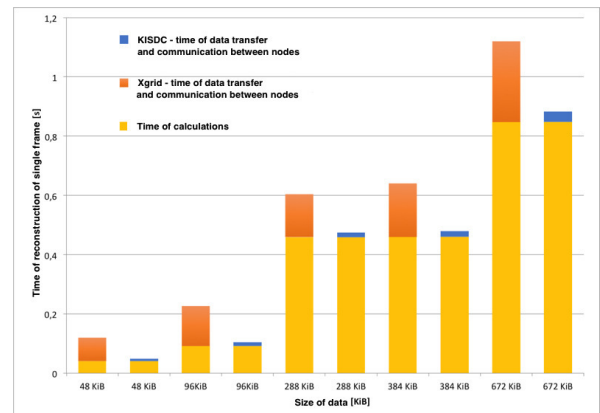


Fig. 7. Comparison of image reconstruction time in Xgrid and KISDC systems for dual computer configuration

C2070 compute devices in the first node and dual GTX 570 in the second).

The comparison of times of a single frame reconstruction in the two node system are shown in Figure 7. Yellow color represents calculation time (the same for the both systems), blue color is related to data transfer time for the KISDC system, and orange color denotes data transfer time for the Xgrid system. For each of the analyzed data sizes, the speed up of image reconstruction expressed in the number of reconstructed frames per second was noted. The most significant acceleration was achieved for 48 KiB and 96 KiB image vectors.

IV. CONCLUSIONS

A flexible, distributed computing system for tomographic image reconstruction called KISDC has been designed and developed. The system's framework allows to accelerate any kind of computation dealing with a basic linear algebra operations. However, it should be noted that the KISDC is highly scalable and can be easily extended either by specific OpenCL kernel or by a plug-in providing support for a special kind of calculations.

The work described in this paper was focused on improvement of data management in the distributed system and on reducing delays in the data transmission over the computer network. The comparison of times of data transfer and communication between the nodes shows very clearly that the use of the new developed system with HTTP protocol ensures much better results than with the Xgrid platform. It is also evident that the KISDC system allowed for a significant reduction of the total time of a single frame reconstruction and a major speed up in implementations of both the LBP and the Landweber algorithms.

ACKNOWLEDGMENT

This work was financed by the Lodz University of Technology, Faculty of Electrical, Electronic, Computer and Control Engineering as a part of statutory activity (project no. 501/12-24-1-5418)

REFERENCES

- [1] Banasiak, R., Wajman, R., Fidos, H., Jaworski, T., Fiderek, P., Kapusta, P., Majchrowicz, M., Sankowski, D., "Fusion of three-dimensional electrical capacitance tomography and fuzzy logic inference for phases fraction and flow structures identification in horizontal and vertical gas-liquid flow pipelines.," *7th World Congress on Industrial Process Tomography*, Kraków, 2013, pp. 818–827.
- [2] Barrachina, S., Castillo, M., Igual, F.D., Mayo, R., Quintana-Ortí, E.S., Quintana-Ortí, G., "Exploiting the capabilities of modern GPUs for dense matrix computations.," *Concurrency and Computation: Practice and Experience* 2009.
- [3] Cui, Z., Wang, Q., Xue, Q., Fan, W., Zhang, L., Cao, Z., Sun, B., Wang, H., Yang, W., "A review on image reconstruction algorithms for electrical capacitance/resistance tomography," *Sensor Review*, Vol. 36 Issue: 4, 2016, pp. 429–445.
- [4] Fielding, R., Gettys, J., Mogul, J., Frystyk, H., Masinter, L., Leach, P., Berners-Lee, T. (1999): Hypertext Transfer Protocol – HTTP/1.1. Request For Comments (RFC), 1999, RFC Editor.
- [5] Garbaa, H., Jackowska-Strumiłło, L., Grudzień, K., Romanowski, A., "Neural network approach to ECT inverse problem solving for estimation of gravitational solids flow," *2014 Federated Conf. on Computer Science and Inf. Systems (FedCSIS)*, IEEE Xplore Digital Library, 2014, pp. 19–26.
- [6] Garbaa, H., Jackowska-Strumiłło, L., Grudzień, K., Romanowski, A., "Application of electrical capacitance tomography and artificial neural networks to rapid estimation of cylindrical shape parameters of industrial flow structure", *Archives of Electrical Engineering*, Vol. 65(4), 2016, pp. 657–669.
- [7] Grudzień K., Chaniecki Z., Romanowski A., Niedostatkiwicz, M., Sankowski, D. "ECT Image Analysis Methods for Shear Zone Measurements during Silo Discharging Process," *Chinese Journal Of Chemical Engineering*, Vol. 20 (2), 2012, pp. 337-345.
- [8] Kapusta, P., Majchrowicz, M., Sankowski, D., Jackowska-Strumiłło, L., Banasiak, R., "Distributed multi-node, multi-GPU, heterogeneous system for 3D image reconstruction in Electrical Capacitance Tomography - network performance and application analysis," *Przegląd Elektrotechniczny*, 89 (2 B), 2013, pp. 339-342.
- [9] Kapusta, P., Majchrowicz, M., Sankowski, D., Jackowska-Strumiłło, L., "Acceleration of image reconstruction in 3D Electrical Capacitance Tomography in heterogeneous, multi-GPU system using sparse matrix computations and Finite Element Method," *2016 Federated Conference on Computer Science and Information Systems (FedCSIS)*, IEEE Xplore Digital Library, 2016, pp. 679-683.
- [10] Kapusta P., Duch P., Jaworski T., Kucharski J., Ślot K., "Generative network input shaping for controlling visual object rendition in Adversarial Networks", *Proceedings of International Interdisciplinary PhD Workshop 2017*, Lodz, 9-11 Sep. 2017, pp 380–385.
- [11] Kirk, D.B., Hwu, W.-M.W., "Programming Massively Parallel Processors," *Morgan Kaufmann* 2010.
- [12] Krüger, J., Westermann, R., "Linear algebra operators for GPU implementation of numerical algorithms," *ACM Transactions on Graphics (TOG) - Proceedings of ACM SIGGRAPH* 2003.
- [13] Majchrowicz, M., Kapusta, P., Waś, Ł., Wiak, S., "Application of General-Purpose Computing on Graphics Processing Units for Acceleration of Basic Linear Algebra Operations and Principal Components Analysis Method," *Man-Machine Interactions 3, Advances in Intelligent Systems and Computing Volume 242*, Springer International Publishing, 2014, pp. 519–527.
- [14] Majchrowicz M., Kapusta P., Jackowska-Strumiłło L., Sankowski, D., "Analysis of Application of Distributed Multi-Node, Multi-GPU Heterogeneous System for Acceleration of Image Reconstruction in Electrical Capacitance Tomography," *Image Processing & Communications*, vol. 20, Issue 3, 2015, pp. 5–14.
- [15] Majchrowicz M., Kapusta P., L. Jackowska-Strumiłło, D. Sankowski, "Acceleration of image reconstruction in 3D Electrical Capacitance Tomography in heterogeneous, multi-GPU, multi-node distributed system", *Proceedings of International Interdisciplinary PhD Workshop 2017*, Lodz, 9-11 Sep. 2017, pp. 164–169.
- [16] Sankowski, D., Grudzień, K., Chaniecki, Z., Banasiak, R., Wajman, R., Romanowski, A., "Process tomography development at Technical University of Lodz", *Electrical Capacitance Tomography Theoretical Basis and Applications*, edited by Dominik Sankowski and Jan Sikora, Warszawa, 2010, pp. 70-95.
- [17] Sun B., Yue S., Cui Z., Wang H., "A new linear back projection algorithm to electrical tomography based on measuring data decomposition", *Measurement Science and Technology*, Vol. 26, Number 12, 2015, pp. 1270-1283
- [18] Soleimani, M., Mitchell, C.N., Banasiak, R., Wajman, R., Adler, A., "Four-dimensional electrical capacitance tomography imaging using experimental data," *Progress In Electromagnetics Research PIER*, 90, Hong Kong, 2009, EMW Publishing, pp. 171–186.
- [19] Tsuchiyama, R., Nakamura, T., Iizuka, T., Asahara, A., Miki, S., Tagawa, S. (2010): *The OpenCL Programming Book*. 1st ed. Fixstars Corporation.
- [20] Tchorzewski, P., Rymarczyk, T., Sikora, J. " (2016). Using Topological Algorithms to Solve Inverse Problem in Electrical Impedance Tomography." In: Mikulka, J. (Ed) *Proc. International Interdisciplinary PhD Workshop Location: Brno, Czech Republic, 12-15 Sep. 2016*, pp. 46-50.
- [21] Fiderek, P., Kucharski, J., Wajman, R. (2017) Fuzzy inference for two-phase gas-liquid flow type evaluation based on raw 3D ECT measurement data. *Flow Measurement and Instrumentation*. Vol. 54, pp. 88-96.