Tomographic Determination of Emissivity Profiles in the ISTTOK Tokamak

César Augusto Silva Alves
Instituto Superior Técnico, Lisboa, Portugal

June 2016

Abstract

The ISTTOK tokamak is one of the only worldwide tokamaks which makes use of a tomography diagnostic for its real-time control. The upgrade of its cameras made necessary an evaluation of its hardware and the development of an updated version of its algorithms for real-time use. The new cameras were tested and evaluated to ensure their proper functionality and ascertain which modifications they need. A new software tool was developed to calculate the tomographic reconstruction and optimization of the sensors, camera layout and basis functions.

Due to the increased number of sensors, the previously used algorithm is no longer suitable for real-time operation of the diagnostic. To overcome this limitation, the algorithm was transposed to a GPU, where the parallelizing capabilities allow for the increase of the reconstruction resolution and accuracy, while maintaining its execution timing under $100\mu s$.
Keywords: ISTTOK, Tokamak, Tomography, Real-Time

1. Introduction

The Tokamak device is a toroidal vessel which contains and heats hydrogen, or deuterium and tritium to temperatures high enough (approximately $T = 10^7$ K) to turn them into plasma. This engineering device is a containment apparatus which uses magnetic fields to mantain the hot plasma suspended with as little interaction as possible with its walls, thus preventing degradation and ensuring the high temperatures and pressures are mantained.

Tomography is a technique to obtain an image of the cross-section of a particular medium based on integrated measurements. Ideally these measurements should be taken with a full sweep of the 360 ° of the analysis plane.

In a tokamak such coverage is rarely possible, however the use of tomography in tokamak diagnostics has grown in interest as it is a powerful technique for the acquisition of profile and position data from passive plasma observation. To overcome the poor coverage various methods of reconstruction based on physical assumptions about the plasma behaviour have been implemented in a variety of concepts.

The ISTTOK(IST TOKamak) has had a real-time plasma positioning system based on tomography[2]. This system is now being upgraded to increase the number of lines of sight and thus provide a better level of accuracy. This thesis describes the work done on optimizing the new system geometry, parameters,and also the software upgrade to accomodate the extra channels.

2. Tomography Cameras

The tomography setup in the ISTTOK tokamak was recently upgraded[1] and currently it posesses three pinhole cameras set at 0°, 90°, and 183° angles at the same toroidal location.

The current cameras are made of an electronics board coupled to a flange of the ISTTOK ports, with fixation support both for stability of the electronics, but also for the cylinder which covers the board and photdiode array from any extra radiation. The arrays are assembled such that they end up aligned with the poloidal plane, and the pinhole is centered on the top of the cylinder.

Due to a high amplification factor on the electronics the observed signals were saturated constantly. To obtain usable signals without rebuilding the electronics, a 4700 Å to 4900 Å filter was applied

3. Tomography Algorithm

The usual obtaining of a tomography profile is easily mathemathically described by a simple model.

This obtained final radiation is an integral value I which contains contributions from all the attenuation factors $g(x, y)$ of every point of the object through which the radiation travelled, and although the information presented in this way is reduced when compared to the ammount of data present in g it can still be used for the reconstruction upon certain conditions. By obtaining these results not

once but several times(depending on the desired resolution and capabilities) and by utilizing the Fourier Slice Theorem the common tomographic reconstruction can map the obtained intensities with the reconstructed 2D plan on which the material attenuations are located and obtain a final accurate reconstruction image.

As the radiation passes through an object, its intensity diminishes depending on the density of the material. By modelling the density by a function $g(x, y)$, we can integrate it along its path, thus obtaining its final intensity I relative to the initial emission intensity I_0.

$$I = I_0 \exp\left[-\int_L g\,(x, y)\,\mathrm{d}l\right] \tag{1}$$

So, considering a single viewline for a hypothetical angle, and changing the previously line integral into an area integral.

$$f\,(\rho, \phi) = \int_A g\,(r, \theta)\,\mathrm{d}A \tag{2}$$

To describe this

$$g\,(r, \theta) = \sum_{n=-\infty}^{+\infty} g_n\,(r)\,e^{in\theta} \tag{3}$$

Or, in the more common trigonometrical representation,

$$g\,(r, \theta) = \sum_{n=0}^{\infty} g_n^c\,(r)\cos\,(n\theta) + g_n^s\,(r)\sin\,(n\theta)\,, \tag{4}$$

With this final expansion the $g\,(r, \theta)$ which started as an arbitrary function ends up as the sum of a series of expansions defined by the coefficients $a_{nl}^{c,s}$.

$$g\,(r, \theta) = \sum_{n=0}^{\infty}\sum_{l=0}^{\infty}\sum_{t=\cos,\sin} a_{nl}^t g_{nl}\,(r)\,t\,(n\theta) \tag{5}$$

For each line i the line integral can be calculated with a projection operator on the reconstruction function:

$$f_i\,(\rho, \phi) = \mathrm{Proj}_i\,\{g\,(r, \theta)\} \tag{6}$$

For calculation on a computer g must be discretized1, and the projection operator becomes a Frobenius product with the discrete g.

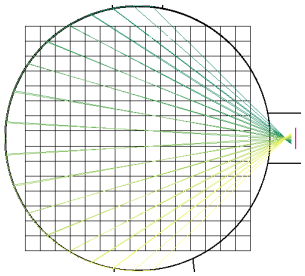

Figure 1: The projection matrix $\mathbf{P}_i$ is obtained from the percentage of intersection of each square 'pixel' with each pair of viewlines.

Inserting equation 5 in equation 2, and assuming that g can be described by a set of basis functions $(\mathbf{BF}_j)$

$$f_i\,(\rho, \phi) = \sum_{j=0}^{n_{BF}-1} a_j \mathbf{P}_i : \mathbf{BF}_j. \tag{7}$$

This can be represented in matrix form as:

$$\vec{f} = \mathbf{C}\vec{a}, \tag{8}$$

To obtain $\vec{a}$, the pseudo-inverse of $\mathbf{C}$ is required. The best way to calculate it is by performing the SVD decomposition which elements can then be used for the subconsequent calculation of the coefficients a, which can be determined in the least-square sense by:

$$\vec{a} = \mathbf{C}^+\vec{f} = \mathbf{V}\mathbf{W}^+\mathbf{U}^T\vec{f}, \tag{9}$$

The basis functions used in this work were the Bessel functions which are categorized by being an orthogonal set which always tends to 0 at $r = 1$ and which are commonly enough to be easily calculated. Their behaviour is such that they can be described not only by their kind n, but also by the number of zeros belonging to each function l.

$$g_{nl} = J_n(x_{nl}r) \tag{10}$$

Where x_{nl} is the $l + 1$ zero of J_n.

From the sensor data $\vec{f}$ it is then possible to calculate the coefficients $\vec{a}$ through 9, and then calculate the tomographic reconstruction with equation 5.

3.1. Optimization of Results

To determine the quality of any reconstruction, a know 'virtual' emissivity function is used, called a phantom($\mathbf{F}$), which is able to be simulated and from which virtual projections are calculated and permits direct comparisons of the phantom with the reconstruction($\mathbf{R}$). To measure the difference between these two, it's done element by element at each cell, (k,l):

$$\varepsilon = \sum_{k=0,l=0}^{M_s,M_s} \frac{|\mathbf{R}_{k,l} - \mathbf{F}_{k,l}|}{M_s^2}. \qquad (11)$$

The only variable which can be adjusted in the geometry of this system is the distance of the cylinder. On the top of the cylinder is the pinhole, which will vary its distance to the photodiode array. To optimize this distance for the system of 3 cameras, a sweep was done across the possible distances with intervals of $1mm$ for a series of $\sim$ 200 phantoms. By comparing the ε of the various reconstructions done this way, a point was found where the choice of distances minimized ε.

The optimal choice of geometry was of top camera = 38 mm, equatorial camera = 38 mm, and bottom camera = 36 mm.

In the reconstruction model of the plasma, the number of basis functions is only limited by the number of sensors, which is at most 47. This means that the choice of which basis functions to use is entirely free while mantained under this limit. While the low order functions give general features to the reconstruction, high order elements can create localized artifacts. Therefore a sweep was made along all the possible combinations of maximum index for the basis functions N and L, with the objective of comparing the ε from the various reconstructions, and finding the values which permit for the best reconstruction without overfitting or other kind of artifacting.

The found minimum was of the values N = 2, L =2;

3.2. Final Reconstructions

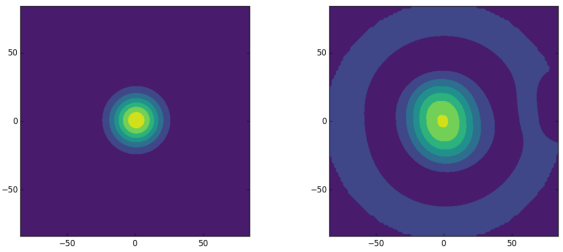

Figure 2: Reconstruction of a central phantom. On the left the initial phantom, on the right the obtained reconstruction. All positions are in mm and in the (R, Z) coordinate plane of ISTTOK.

The figures here represented correspond to the phantoms and the reconstructions, respectively left and right side, simulated for the best parameters found for geometry and basis functions in the previous section.

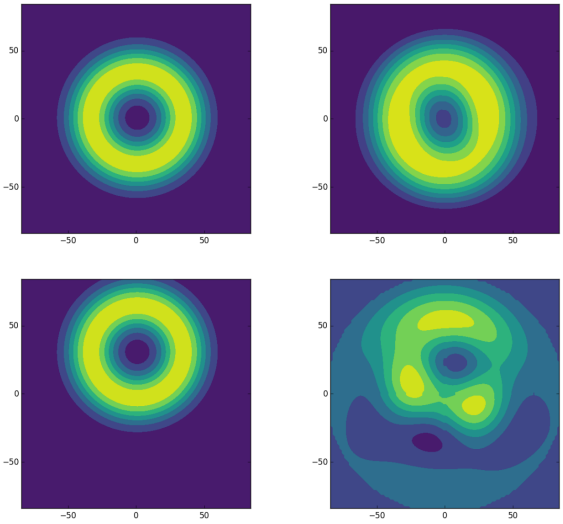

Figure 3: Reconstruction of various rings with variations in shape and position.

4. Software - MARTe and CUDA

4.1. The MARTe infrastructure

The Multithreaded Application RealTime executor(MARTe) is the real-time framework which is currently implemented and serves as the real-time control system for the ISTTOK tokamak. The great advantages of this specific framework are its modulability and multi-platform capabilities, despite it being a C++ based system, it is built upon the BaseLib library which implements the API and data structures in an Operating System(OS) independent manner, crafting as such the main structure necessary for the implementation and execution of a control system. Over this implementation MARTe runs this control system.

The main objective of the MARTe framework to run this control system, is based on the clear boundary between its various component elements, even though it manages and serves for implementation of both hardware controllers, software algorithms, and GUI interfaces, it mantains clear boundary between all of those systems. MARTe itself is the system which manages the memory, and core on which the processes are run, therefore mantaining all the necessary elements such that each module can work within its real-time capabilities.

4.2. The CUDA Platform

Developed by the NVIDIA® corporation, the CUDA™ architecture was created to work as both a typical GPU unit, capable of graphics rendering for various kind of applications, but also with its specifications set up in such a manner that each one of its arithmetic logic units(ALU) are compatible with the same floating-point arythmetic that is defined for a CPU. This versatibility served for the popularization of the CUDA technologies, and has

its level of performance and popularization has increased, the number of scientific applications have accompannied this growth.

The GPU installed in the real-time control computer connected to ISTTOK is a NVIDIA GeForce GTX 580, a mid-range GPU which serves as a proof of concept, not being a professional dedicated GPU for computational uses, but posessing flexibility enough for a series of tests.

4.3. The Tomography GAM

The process of implementation of the CUDA capabilities of the hardware with the GAM implementation is not as straightforward as the usual compilation of code for either one of those processes. To use these two distinct types of technology, considerations are necessary to make sure that they do not face conflicts head-on.

4.3.1 Using CUDA within a GAM

To work with CUDA capabilities within a GAM, the work was done by implementing a series of precompilled functions which made use of the CUDA capabilities, called from the C++ GAM itself. This allowed the typical management of memory and control that MARTe demands while permitting that the CUDA code would run properly and without any errors.

The following figure 4 was obtained from the visualizations of the NVIDIA Visual Profiler, and shows the relative lengths in computation time that each step takes in the algorithm. As it is possible to see, there exist various areas of non-overlap and even temporal areas where nothing appears to happen, those are in general several overheads of the CUDA computation tools as the various kernels and copy operations are initialized. In this way despite the seemingly non-conflict between pretty much any element of this algorithm, a certain care was taken to make it as fast and stable as possible.

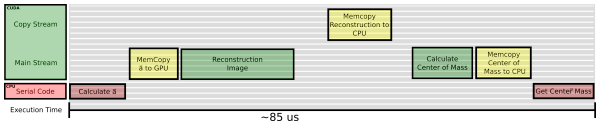

Figure 4: Pipeline of the total algorithm when running in the pc. Red refers to processes run in serial code, green to processes run in CUDA code, yellow to memory transfers(Memcopy) from host-to-device or device-to-host.

When advantageous, calculations are performed on the CPU as the faster processing power makes up for the single core processing, but those are located on the beggining and end of the algorithm. If one of those was attempted on the middle of the algorithm it would imply wasting time in transfering data from device-to-host and back again, a process which could easily increase the total time spent in the operation.

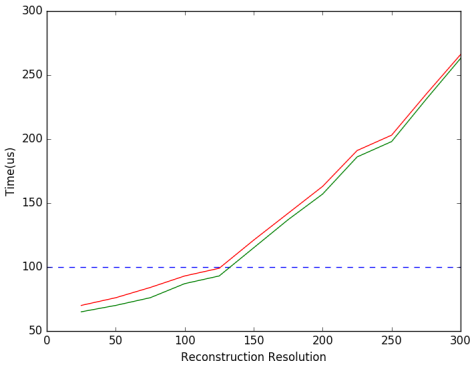

Figure 5: Reconstruction speed as measured of the final algorithm. The red line corresponds to the calculation of $\vec{a}$ in the GPU, while the green line corresponds to the CPU.

As observed, both of the versions of the CUDA code perform well and are stable for any kind of reconstruction size. The version where the calculation of $\vec{a}$ is done on the CPU is consistently faster, but both versions achieve resolutions of 100×100 without crossing the threshold of $100\mu s$.

With this code such implemented, for the integration with MARTe the functions are wrapped around common C++ functions, which communicate in the main body of the GAM. In this way they work alongside any other GAM, and are integrated in the real-time control system.

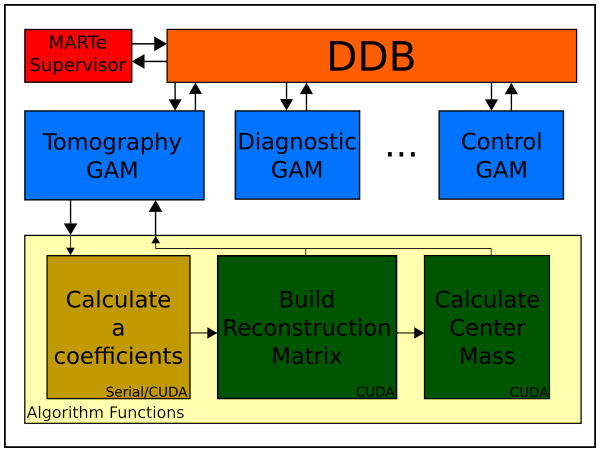

Figure 6: GAM Structure with the CUDA code

5. Conclusions

The work done in this thesis has shown that the tomography diagnostic in ISTTOK is well capable of performing high quality reconstructions with the

new cameras and through the integration of GPU based computing.

Observation of the spectra from the new cameras and work done on optimizing it has revealed that the addition of a blue filter both solves the previous saturation problem and allows for the localization of high intensity regions of plasma radiation.

Through the determination of an optimal geometry and basis functions for the reconstruction algorithm used, the main artifact problems with this real-time method were counter-balanced, permitting the calculation and obtention of this data for real-time control to be ever more accurate. This alongside the simulation for the new geometry indicates that with those new cameras despite the limitations of ISTTOK the presence of the tomography diagnostic is a high boon for its control systems.

The real-time implementation through CUDA was sucessful, obtaining results with the same accuracy as the serial version for the same time with small resolutions and much improved times for large resolutions. This allows the increase of the reconstruction resolution to an order of magnitude higher, up to a point where its results are as good as needed for any kind of application.

5.1. Future Work

The effort made in getting the new cameras to work indicates that the addition of a filter and a slit instead of a pinhole seems to be the best solution in terms of obtaining a good signal while at the same time increasing the physical reliability of the tomographic reconstruction. The implementation of those cameras would certainly add new avenues of real-time control at ISTTOK and permit better plasma reconstructions all around.

The work done here in terms of bridging GPU capabilities for use with MARTe real-time control system although not the best diagnostic example, shows that these technologies are capable of working together and in such a way that they provide a way to perform high-data dependant calculations which before served as bottlenecks for the diagnostic determinations. This work can be explored on more in-depth applications of other different kind of algorithms and problems which might be more suited towards this kind of advantages and resolutions such as reflectometry now that it is known that the bridge between CUDA and MARTe is possible and with good results.

References

[1] Francisco Burnay. Tomografia no tokamak isttok. Master's thesis, Instituto Superior Técnico, 10 2015.

[2] Pedro Carvalho. *Tomography Algorithms for Real-Time Control in ISTTOK*. PhD thesis, Instituto Superior Técnico, 10 2009.