Parallelized and Multithreaded FFT Applied to Extraction of EEG Data Key Parameters Towards Seizure Detection and Prediction.

SACNAS national conference

Gabriel Lizarraga, Mercedes Cabrerizo, Melvin Ayala, Malek Adjouadi

Center for Advanced Technology and Education, Florida International University, Miami, Florida

1 Abstract

In this paper the possibility of parallel computing applied to intracranial electroencephalograph (iEEG) data processing is explored. It focuses in particular on Fast Fourier Transform (FFT). Applying the FFT algorithm to the iEEG data is the most computational intensive step in the seizure prediction or detection process. This study focuses on three algorithms: sequential, multithreaded, and parallel processing. Running times will be compared and an estimate of how many CPUs are required to process the iEEG data on the parallel implementation will be provided. Data was obtained from real patients, ranging from 3 to 17 years old. The time saving achieved in this study might provide enough time to warn a patient before an upcoming seizure.

2 Introduction

Seizures, caused by excessive synchronous neuronal activity in the brain, are unpredictable and occur at random intervals of time. Some patients who have recurrent seizures are able to tell when a seizure is about to happen , however, for most patients the seizure occurs without warning [1]. Seizures can be very damaging and impose a state of fright in people who suffer them, as loosing consciousness may occur while walking or driving, with deadly consequences [2]. EEG and iEEG data have been previously analyzed at the Center for Advanced Technology and Education at Florida International University, but since the size of the EEG data is a critical issue when dealing with seizure detection and prediction paradigms, a reliable and computational efficient method needs to be implemented to speed up the process [3].

In this paper we explore the possibility of parallel computing applied to EGG and iEEG data processing. We focus in particular on Fast Fourier Transform (XXX) applied to an iEEG dataset. Our paper studies three distinct algorithms: sequential, multithreaded, and parallel processing. Running times will be compared in order to obtain the best algorithm.

3 Methods

3.1 Participants

The data used in this study was obtained sequentially from a significant sample of 8 patients who underwent two-stage epilepsy surgery with subdural recordings. The age of the subjects varied from 3 to 17 years. The number and configuration of the subdural electrodes differed between subjects, and was determined by clinical judgment at the time of implantation. Grid, strip, and depth electrodes were used, with a total number of contacts varying between 20 and 88. The amount of data available for analysis was influenced by its recording duration, and by the degree to which the interictal EEG was "pruned" prior to storage in the permanent medical record. The iEEG data was recorded at Miami's Children Hospital (MCH) using XLTEK Neuroworks Ver.3.0.5, equipment manufactured by Excel Tech Ltd. Ontario, Canada. The data was collected at 500 Hz sampling frequency and filtered to remove the DC component. All data sets for this particular study were iEEG segments of 20 minutes approximately (200 Megabytes).

3.2 Implementation Steps

Three algorithms were created in C language: Sequential, Multithreaded, and MPI (parallelized). All three algorithms were compiled and ran in the same hardware. For all algorithms FFTW (Fast Fourier Transform in the West) [4] was used, which is an implementation of the FFT algorithm. All algorithms were coded, compiled and tested at Marenostrum, located in the Super Computing Center in Barcelona, Spain. Marenostrum is a supercomputer with 10240 CPUs, capable of 94.21 Teraflops [5]. The main objective is to analyze the iEEG data and to determine how parallelization can speed up the process of running FFT on this data.

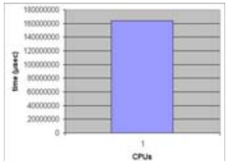
The results are obtained from the same iEEG dataset and compared based on performances in order to determine (a) How much faster parallel FFTW is than sequential or multithread; (b) How many threads provide enough performance; and (c) How much parallelization is needed.

•The sequential algorithm was ran 10 times and the results averaged. This implementation was done to have a basis of comparison with the other two. Results are shown in figure 1.

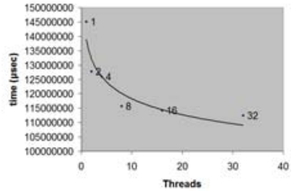
•The multithreaded algorithm was ran with 1,2,4,8,16, and 32 threads. Running times were recorded and averaged. Results are shown in figure 2.

•The parallelized algorithm was the hardest to code and the more interesting. The EEG data was partitioned by the algorithm among several CPUs and MPI (Message Passing Interface) was used to synchronize the processing. This algorithm was executed with 1,2,4,8,16,34, and 64 CPUs. Figure 3 shows the results.

4 Results

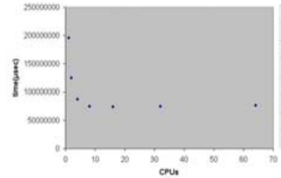


Average running time:
163872162.5 µsec



Threads	time
1	145184192
2	127785492
4	125973272
8	115719315
16	114281754
32	112527251

Figure 2: Multithreaded Results.



CPUs	time
1	195413731
2	124914390
4	87626182
8	75209535
16	74068303
32	75246532
64	77009326

Figure 3: MPI Results.

5 Discussion

Our results show that, as expected, the multithreaded and parallelized perform better than the sequential algorithm. The multithreaded algorithm improves as more threads are added, but more than 4 threads do not provide a significant improvement, we believe that this is because the overhead of creating the threads and address spaces takes CPU cycles which could have been used to process the data. Last, our parallel algorithm has a significant improvement when 4 CPUs are used. However, more than 7 do not provide a significant improvement. It is interesting to observe that the runtime with 64 CPUs is larger than the runtime with 8. We think this occurs because breaking the array into 64 parts is more expensive than the computational time of FFTW. The computational time was cut in about 50% by using the parallel implementation (over the sequential). This time saving might provide the leverage required to predict a seizure with enough time to warn a patient.

6 Acknowledgments

The authors appreciate the support provided by the National Science Foundation under grants CNS-0426125, HRD-0833093, CNS-0520811, CNS-0540592, IIS-0308155 and OISE-0730065. The authors are also thankful for the clinical support provided through the Ware Foundation and the joint Neuro-Engineering Program with Miami Children's Hospital.

7 References

1. A. Shoeb, H. Edwards, J. Connolly, B. Bourgeois, T. Treves, and J. Guttag, Patient-specific seizure onset detection, Proceedings of the 26th Annual International Conference of the IEEE EMBS, San Francisco, CA, USA, September 1-5 (2004).
2. R. Chander, E. Urrestarrazu, and J. Gotman, Automatic detection of high frequency oscillations in human intracerebral EEGs, Epilepsia, 47:4:37-37 (2006).
3. M. Cabrerizo, M. Adjouadi, M. Ayala, and M. Tito, Pattern extraction in interictal EEG recordings towards detection of electrodes leading to seizures, Biomedical Sciences Instrumentation, 42 (ISA Vol. 464) pp.243-248 (2006).
4. Frigo, M, and S,G Johnson. "FFTW: an adaptive software architecture for the FFT." IEEE Explore. 3.12-15 May (1998): 1381 - 1384.
5. BSC "MareNostrum User's Guide".2009. Barcelona Supercomputing Center. <http://www.bsc.es/media/859.pdf>.

Supported By

FIU Florida International University Miami's public research university