

ACCELERATION OF ALGORITHMS USED IN 4G NETWORKS IN RESEARCH AND CLASSROOM

Tomas Palenik¹, Martin Rakus¹, Peter Farkas¹, Markus Rupp², Adao Silva³, Jan Dobos⁴

¹Department of Transmission System, Institute of Telecommunications, Faculty of Electrical Engineering and Information Technology, Ilkovicova 3, 812 19 Bratislava, Slovakia

²Institute of Telecommunications, Technische Universität Wien, Gusshasstrasse 25/E 389, A – 1040 Vienna, Austria

³Instituto de Telecomunicações, Campus Universitário de Santiago, P – 3810-193 Aveiro, Portugal

⁴Institute of Applied Informatics, Faculty of Informatics, Paneuropean University, Tematínska 10, 851 05 Bratislava, Slovakia

Abstract

This paper provides guidelines for young researchers regarding the preparation of their scientific projects and subsequent transfer of experience from the research into the courses they teach in the classroom. As a complete example it focuses mainly on the area of advanced wireless communication networks, such as the known 3GPP LTE and LTE-A, directly targeting important problem in complex system research – the acceleration of simulations by means of parallelization. Such processing influences the research progress, it also brings important experience that can be later used in classroom.

Key words: acceleration, simulation, GPU, turbo-decoding

1. INTRODUCTION

Recent 3GPP's LTE-Advanced 4G standard (see 3GPP, LTE-Advanced), represents one the most advanced communication standards available. It utilizes many of the most advanced technical concepts, such as cognitive radio, advanced MIMO techniques, coordinated multipoint and most powerful ECC codes. The turbo and LDPC codes present in this standard (see 3GPP, LTE) are in the focus of many research and educational activities. Several decoders of powerful ECC - turbo and LDPC codes exist to date. Some use dedicated DSP hardware (Song et al. 2005), others take advantage of an FPGA implementation (Nezami et al. 2006). Many software implementations also exist (Wu et al. 2011).

The applicability of above mentioned decoders in commercial environment is undeniable. The situation in academia is somewhat different. The academic research can often be described in term of iterations of proposing a novel idea, testing the concept using simulations, and often modifying the original concept based on simulation results. Therefore for the study and potential improvement of the most recent algorithms, employed in LTE-A and beyond, presences of two important factors are necessary: First one is a good simulation environment. One of the most widely used and successful simulation platforms is the well-known Matlab. Among its most important drawbacks is usually a prohibitively long simulation run-time in situations that include a complex advanced design. Second drawback is the closed nature of Matlab toolboxes – the source code of most of the interesting system components is for obvious reasons unavailable and therefore unmodifiable.

The paper is organized as follows: The second section gives general guidelines for a researcher regarding the organization and planning of his or her work in order to achieve a coherent research progress. It presents a case –study proposal for acceleration selected elements of a 4G communication system block chain. The third section provides information on the implementation phase – describes the workflow and provides examples of obtained results. The fourth section deals with transferring of

knowledge to classroom, so that the also the curriculum can benefit from the scientific results. Section five concludes the paper.

2. PLANNING A 4G SIMULATION ACCELERATION PROJECT

For successful communication system research project, one must first be aware of the structure of the system. Figure1 gives the simplified model that can be used as an overview of the PHY layer of an OFDM-based system, such as the LTE and LTE-A.

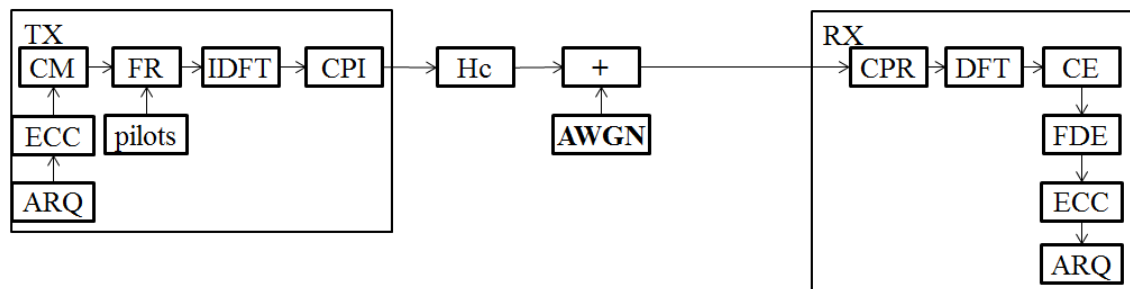


Fig. 1. Simplified Block Structure of a PHY layer of a modern OFDM-based communication system.

The acronyms in fig. 1 should be reasonably well known, however, for the sake of completeness, they are also explained here: ARQ stands for the Automatic Repeat and reQuest, ECC for Error Control Coding, CM for Constellation Mapping – the modulation in digital domain, FR for Framing, (I)DFT for (Inverse) Discrete Fourier Transform, CPI/R for Cyclic Prefix Insertion/Removal, CE for Channel Estimation, FDE for Frequency Domain Equalization, The channel model is assuming a Rayleigh-fading multipath channel described by channel matrix H_c with AWGN noise.

The focus of the project lies in the ECC block in the receiver, however the fit into the overall system model must not be forgotten, even in the project design phase. Moon (2008) gives details regarding these codes.

2.1. Research and output – putting goals together

Before anything else, one must first dedicate careful thought to planning and stating the projected goals of the project. It is necessary to write down the proposed goals. This activity serves the researcher as a basic plan, while also useful for a potential grant application. The next few points provide an example for a GPU acceleration project:

- The main goal of an example research proposal is to implement a massively parallel, well documented GPU-optimized decoder, or other computationally complex system block, implementing one of the known turbo-decoding algorithms (Moon 2008). This decoder must be transparently callable from within the Matlab environment.
- Make the source code of this optimized implementation widely available for the world's scientific and academic community by integrating it into selected existing (open-source if possible) LTE-simulation framework, such as (Mathworks; TUV LTE and Wimax simulators). The open character (GPL) of source code will facilitate the further adoption of the framework.
- Prepare a classroom kit for students – extend existing simulation framework and documentation by preparing a set of small tutorial simulations for students. These simulations should be targeted to selected areas, with focus on turbo-coding and decoding, that are currently being lectured in existing classes at many universities.

2.2. Methodology – which steps should be taken

Following the goals, the next thing to prepare would be an overview of a methodology used throughout the work the research part of the project. The theoretical study of known algorithms can be done by the applicant himself. The important part is the frequent consulting of difficult spots with the more experienced senior lecturers. The need for these consultations arises in a hard-to-predict manner. Another important aspect of the proposed research stay is the day-to-day working together with the implementation team at the institute with people that have deep understanding of the operation of existing simulation framework and are also more experienced in the area of GPU computing. This day-to-day contact must be carried out directly at the institute – a remote operation would introduce prohibitive delays.

Within this setting, the following proposed methodology can be applied not only to decoding of modern LDPC and turbo-codes, but to any algorithm in the communication system:

- Implementing and profiling a single threaded low-level C code by using standard software tools in Matlab and a pool of specialized custom written test scripts.
- Analysis of the parallelisms made theoretically possible by the design of the code. This includes possible parallelisms in trellises of standard defined (Song et al. 2005) turbo codes. This includes an analysis of the parallelism that can be implemented in a GPU – analysis of the HW limitations.
- Developing a formal specification of the parallel algorithms to implement, with the analyses from previous steps taken into account.
- Optimized CUDA C implementation of the proposed parallel algorithm, with interoperability with Matlab environment and existing simulation platform in mind and testing and performance increase evaluation by running the test script pool and profiling.

It may be (and likely will be) necessary to repeat these general steps with different contexts and with different parameter sets, interleaved with a team consulting of the progress.

2.3. Stating projected benefits

For the researcher team, and also for the organizations providing financial support, the projected benefits of the research could be divided into four groups:

1. Benefits for your partners organizations :

The proposed GPU optimized implementation of selected, overly computationally complex system block, such as modern ECC decoders will be very beneficial to the partners organization research team. It will enable the researchers to implement new simulations of very demanding novel and most up-to-date techniques and more importantly modify and enhance existing system components to reflect the creative ideas of the research team. By drastically cutting down the runtime of complex simulations it will make such simulations feasible.

2. Benefits for your home institute:

The Faculty is currently building a hands-on communication system and computer networks curriculum, where the current focus is extending the existing theoretical background presented to students in existing courses with a more practical hands-on communications system simulations. The addition of Matlab simulations classes represents a significant upgrade to current curriculum that could result to introduction of a completely new communication course. Preparation of a set of tutorial examples for students is one of the goals of this proposal.

3. Benefits for the European and global academic community:

Since the simulator is available at no charge to the wide academic community, any existing or future user of this framework will benefit from the added functionality. The speedup brought by the optimization and ease-of-use provided by the Matlab integration, along with the proposed open nature of the implementation facilitates a wide adoption of the upgraded simulation platform.

4. Benefits for the researcher him-or-her-self

The activity proposed above is a direct extension of the research efforts spent during the previous period. This could be classified under the communication system acceleration paradigm and a further research in this area is a part of the authors schedule for the upcoming years. The most direct impact would be the usage of this decoder in testing authors already existing novel unpublished ideas for improving the convergence of modern decoding processes in LTE and LTE-A.

3. WORKFLOW DURING THE PROJECT IMPLEMENTATION PHASE – EXPERIMENTS AND RESULTS

Many experiments must be performed during the development phase. The experiments should be organized in a progressive incremental way – in this manner the work starts with the simplest possible scenario utilizing the most simplified model of the system being simulated. Each further experiment should only be a small incremental upgrade of the previous one, focusing on only a narrow aspect. Experimental results should always be evaluated, analyzed and documented before moving to the next, upgraded experiment. In this way, the effect of implementation and conceptual errors is minimized. If too many steps are merged into one, the time to find errors that will inevitably surface will grow exponentially.

The list in table 1 provides an example experimental workflow during a research project focused on acceleration of selected system components – the LTE decoder of modern ECC such as LDPC or turbo-codes. The acceleration was implemented using the nVidia CUDA module that can be transparently called directly from the Matlab environment (NVIDIA CUDA). Several steps needed to be done, and unforeseen problems arose during the implementation. These problems had to be addresses in separate experiments – experiments 5, 7, 8. For the sake of brevity, table one only gives a short names of the experiments. Details regarding each of the experiment are out of scope of this short paper.

Table 1. Progress of experiments necessary for evaluating the accelerated simulation code.

Experiment No.	Topic
1	Potential for simulation speedup
2	Matlab and GPU compatibility
3	Potential speedup of a GPU implementation
4	Serial Index computation equivalency
5	Signum operation intrinsic test
6	Basic runtime API low level potential speedup test
7	Compute capabilities and Matlab imposed limitations
8	Matlab argument passing test
9	Multiple GPUs in parallel
10	Handling errors in kernel
11	Comparing partial results
12	One-to-one C & Matlab equivalency
13	Profiling the GPU decoder

3.1 Experimental results

The whole purpose on any research is to provide quantitative results in a standardized form that can be easily compared to other publish results. It is also necessary to keep and publish as much information regarding the circumstances and simulation parameters. This section provides examples showing the results. It gives results on two complementary aspects of the simulation acceleration: The error performance of the LTE turbo-decoder in an AWGN channel, while also providing a more practical throughput comparison to other platforms, so that the benefits of parallelization and acceleration can be observed.

3.1.1. The decoder convergence

The following figure provides insight into the LDPC decoding process – the waterfall curves are displayed also for half iterations – for the output of the first constituent decoder, which are usually not available outside the decoder. The simulation used turbo code as specified in LTE-A specification (3GPP, LTE) with interleaver size $K = 2048$.

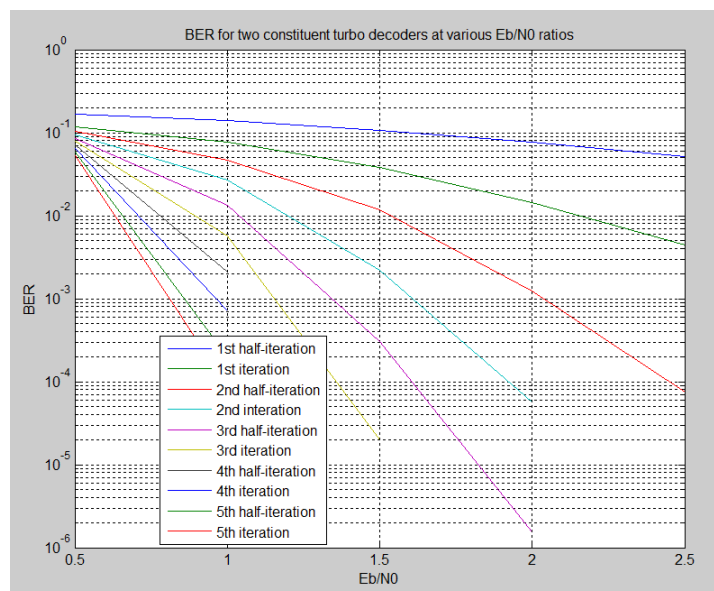


Fig. 2. Convergence of an LTE turbo-decoder for different iterations count.

3.1.2. Decoder throughput comparison

Besides correct decoding, the throughput of the decoder was a critical research concern. The table below summarizes the throughput of the implemented decoders comparing them to other industry standard implementations. Since the throughput of the iterative algorithm is SNR dependent (depending on the stopping criterion implemented), the first three simulations show an experimentally set value of E_b/N_0 that was selected fixed for throughput testing.

Table 2. Comparison of throughput of various implementations of ECC decoders. Palenik et al. (2013) gives detailed information regarding these results

Various decoders throughput comparison			
	Implementation	Throughput [kbps]	E_b/N_0 [dB]
1	Mathworks CPU	225	1.56
2	Mathworks GPU	107	1.54

Various decoders throughput comparison			
	Implementation	Throughput [kbps]	E_b/N_0 [dB]
3	Mathworks GPUopt	339	1.56
4	Matlab CPU naive	1.17	1.38
5	Matlab CPU MEX	167	1.38

4. FURTHER UTILIZATION OF RESEARCH EXPERIENCE – PREPARING A CLASROOM KIT FOR STUDENTS

The department of your institute can greatly benefit from the experience and partial results you gained by implementing your scientific project, especially if it is currently missing a hands-on communication system PHY and DL layer simulation course, where the students would be able to see the more practical aspects of the communication theory they receive in existing courses. The addition of Matlab simulations classes would represents a significant upgrade to existing curriculum. A set of tutorial examples and documentation for students was should be implemented, based on the Matlab decoder version, designed during the work on the research project.

For a computer simulation project, the natural fit is the area of Matlab simulation; therefore an appropriate result would be to prepare a set of Matlab scripts, providing exercises regarding the selected topic – topic of Matlab and CUDA integration and subsequent decoders acceleration. The workflow specified for the research phase is very handy even in the classroom – it serves as a basis for the practical exercises for the students. The lab-exercise schedule can be based on the experiments performed during research:

Table 3. Course exercise plan based on the research workflow.

Exercise No.	Topic
1	Matlab and GPU compatibility
a2	Serial Index computation equivalency
3	Compute capabilities and Matlab imposed limitations
4	Matlab argument passing test
5	Multiple GPUs in parallel
6	Handling errors in kernel
7	Comparing partial results
8	Using debugging tools
9	One-to-one C & Matlab equivalency
10	Profiling the GPU decoder

5. CONCLUSION

This paper provided some general guidelines regarding the preparation of small research projects and subsequent transfer of experience from the research into the classroom. It also described a more concrete examples, describing the work during an existing research project focusing on the area of acceleration of decoders of most modern ECC such as LDPC and turbo-codes used in modern

communication standards such as LTE and LTE-A. Furthermore, information on how to transfer the experience gained during the research to classroom was also provided.

ACKNOWLEDGEMENTS

This work was supported mainly by Slovak Research and Development Agency SK-PT-0014-12 and SK-AT-0020-12 and partially also by Scientific Grant Agency of Ministry of Education of Slovak Republic and Slovak Academy of Sciences under contract VEGA 1/0518/13, by EU RTD Framework Programme under ICT COST Action IC 1104 and by Visegrad Fund and National Scientific Council of Taiwan under IVF-NSC, Taiwan Joint Research Projects Program application no. 21280013 "The Smoke in the Chimney - An Intelligent Sensor - based TeleCare Solution for Homes".

REFERENCES

3GPP, LTE: Evolved Universal Terrestrial Radio Access (E-UTRA); Multiplexing and channel coding (3GPP TS 36.212 version 10.3.0 Release 10) ETSI TS 136 212 V11.0.0

3GPP, LTE-Advanced. Online: <<http://www.3gpp.org/LTE-Advanced>>

K. G. Nezami, S. D. Walker and P. W. Stephens, "An FPGA Implementation of a Memory Efficient, Low Complexity Turbo Decoder Architecture for TETRA Release 2 Application", Proc. of 12th European Wireless Conference - EW2006, Apr. 2006

M. Wu, Y. Sun, G. Wang, J. Cavallaro. Implementation of a High Throughput 3GPP Turbo Decoder on GPU. Journal of Signal Processing Systems (JSPS), 2011. DOI 10.1007/s11265-011-0617-7.

MathWorks, Matlab 2010b Parallel Computing toolbox documentation. Massachusetts, USA, September 2010. Available: <<http://www.mathworks.com/help/toolbox/distcomp/bsic3by.html>>.

MOON, T. K. Error Correction Coding – Mathematical methods and Algorithms. New Jersey : Wiley, 2005. ISBN 978-0471648000. Halfhill, T., Parallel processing with cuda. In Microprocessor report, Arizona, USA, January 2008.

NVIDIA CUDA: Programming Guide, available <<http://www.nvidia.com>>

PALENIK, T., FARKAS, P., RUPP M. et al. Utilizing massive parallelism in decoding of modern error-correcting codes for accelerating communication systems simulations, Proc. Of Informatics 2013 November 5th – 7th, 2013 in Spišská Nova Ves, Slovakia

Song, Y., Liu, G., & Yang, H. (2005). The implementation of turbo decoder on DSP in W-CDMA system. In International conference on wireless communications, networking and mobile computing (pp. 1281–1283).

TUV LTE and Wimax simulators, Available: <<https://www.nt.tuwien.ac.at/downloads/featured-downloads>>