SDR TOOLS AND PROJECTS FOR ELECTRICAL ENGINEERING EDUCATION

Vuk Marojevic, Ismael Gomez, Xavier Arteaga, Pere L. Gilabert, Antoni Gelonch
(Universitat Politecnica de Catalunya, Barcelona, Spain;
marojevic|ismael.gomez|plgilabert|antoni@tsc.upc.edu)

ABSTRACT

This paper focuses on the software-defined radio (SDR) educational aspects. We describe a few recent SDR projects and tools that were developed as part of a graduate electrical engineering course at the School on Telecommunication and Aerospace Engineering (EETAC) in Castelldefels, Barcelona. EETAC is part of the Polytechnic University of Catalonia (UPC). It applies the project based learning (PBL) concept and offers technical and scientific courses in Telecommunication and Aerospace Science and Technology (BSc, MSc, and PhD). The tools we describe here offer hands-on experience on several basic and advanced SDR principles. We describe our experience and the lessons learned.

1. INTRODUCTION

Electrical engineering education should give an overview of the complexity of current and emerging systems while providing enough background to the students for being able to understand and eventually participate in the system development. Wireless system engineering, in particular, requires skills in a wide scope of technologies, including digital signal processing, software design, real-time execution, parallel and distributed computing, and analog-to-digital conversion techniques. The School on Telecommunication and Aerospace Engineering (EETAC) in Castelldefels, Spain, applies innovative learning concepts, such as project based learning (PBL), and offers technical and scientific courses in Telecommunication and Aerospace Science and Technology (BSc, MSc, and PhD).

Software-defined radio (SDR) unifies many concepts and methodologies, from radio engineering to computer science. SDR thus adds additional difficulties to common wireless communications practices. Software design and portability are of special importance in SDR research and development. In order to facilitate the acquisition of these abilities within the tight schedule of electrical engineering education, we developed the abstraction layer and operating environment (ALOE). A set of educational tools have moreover been designed as part of the flexible wireless communications systems and networks (FlexNets) project [2]. The ALOE framework and tools are open source and available for download from [2].

FlexNets is an open-source initiative, offering a portal for collaborative research and education on SDR and emerging radio systems. The EETAC applies these tools in the graduate electrical engineering course *Wireless Communications*. Students in groups of 5-6 develop a wireless communications semester project. SDR projects are regularly offered and taken on semester basis since 2007. The focus of these projects is providing an overview of wireless communications transmitters and receivers and the real-time digital signal processing implications, in particular. The ALOE framework and tools allow a quick start into a new student project. They and facilitate gaining insights into modern wireless system engineering problems and discussing and developing solutions in a short time span.

In this paper we discuss a few recent SDR projects and how ALOE has been useful for understanding some practical problems while gaining a first research experience. These projects can be thought of as a continuity of the undergraduate engineering course *Digital Techniques for Communications*, where the students develop a simple waveform (transmitter and receiver) for real-time execution on a digital signal processor [1]. We present some specific tools for building SDR processing modules and analyzing the real-time computing resource management implications.

2. THE FLEXNETS PROJECT

The FlexNets project was initiated in 2008 as a continuity of previous SDR research efforts at the EETAC, dating back to 2003. By that time, the software communication architecture (SCA) was practically the only alternative. We studied the initial SCA specification and found that it is too complex for our objective, which was designing a flexible and simple framework and toolset for research and education. Virginia Tech's OSSIE framework [5] and GNU Radio [6] appeared later and are used as educational tools in many classrooms.

FlexNets is an open source initiative under the GNU license. It is a portal for research and education on flexible wireless communication systems [2]. The focus of FlexNets is identifying the necessary management elements of wireless communications and incorporating them within evolving systems. SDR technology and cognitive resource management are therefore assumed.

The FlexNets initiative currently focuses on the lower processing layers. We have developed the SDR framework ALOE (Abstraction Layer and Operating Environment). ALOE hides the hardware details and facilitates the real-time execution of SDR applications or waveforms on distributed hardware platforms [3]. It is developed as part of the ALOE project.

The FlexWave project addresses the implementation of waveform modules for their execution on an ALOE-based multiprocessor platform. The FlexCRM project develops a flexible computing resource management framework. It features computing system models and resource allocation algorithms that enable the real-time allocation of abstracted computing resources while considering the waveforms' computing constraints. The aloeUI project develops a graphical user interface for the ALOE middleware and tools.

A set of tools been recently been published for educational purposes (ALOEedu project). These facilitate using and enhancing the ALOE middleware and tools. Our students use these tools for understanding SDR and its resource management implications. Several tools have proven to be useful as starting points for developing semester projects. Some tools were developed by the students themselves as an outcome of their work. We describe the different tools and how they help understanding SDR engineering and research topics.

3. SDR EDUCATIONAL TOOLS

3.1. ALOE Sessions

The growing complexity of the FlexNets project motivated us to introduce laboratory exercises. The ALOE Sessions are self-explanatory and self-containing. They basically require a PC running Linux. Eight sessions are currently available. They provide hands-on experience on different aspects of SDR. Table I summarizes theses sessions.

The ALOE Sessions are regularly updated. They are available for free download from [2]. The sessions can be exercised individually, although most of them require the installation of ALOE. These sessions are used by our students for gaining a practical introduction to the SDR project they are going to develop. As a function of the topic of the project, one or two sessions are typically exercised in the first week of the project (see Section 4).

3.2. ALOE Skeleton

Waveforms describe digital signal processing chains. Each module continuously processes and propagates data. The ALOE framework has a specific execution pattern and the processing modules need to be designed following certain rules. These rules basically enable ALOE to control the execution process, obtain the status of modules and interrupt them if necessary.

In order to simplify the waveform design for ALOE, we have designed a *skeleton*. The *skeleton* can be thought of as a template for implementing digital signal processing algorithms in C (and C++ in future versions). It defines the general input/output interfaces, provides data conversion facilities, and organizes the code into different sections, corresponding to different execution phases, among others.

ALOE Session		Objective
1	Introduction to ALOE	introduce ALOE and guide the installation on a POSIX-based (Linux) execution environment
2	Creating a Waveform	demonstrate how to create and run a waveform, implemented as a set of independent components (waveform modules)
3	Creating Components	exemplify how to create and debug waveform components
4	ALOE User Interface	facilitate loading, initializing, and running a waveform and visualizing or modifying the waveform variables and states
5	Multiprocessing	configure a multiprocessor platform to be managed by ALOE
6	Creating Waveform Components with Simulink	show how to create waveform components from a Simulink model
7	Computing Resource Management Framework	introduce the computing resource management framework, compile the C codes, and use the simulation framework
8	Computing Resource Management Tools	provide insights into ALOE's computing resource management capabilities and present tools for exploration and analyses

The original skeleton requires a certain level of C programming skills and is suggested for medium-skilled programmers. The skeleton currently under development will simplify the generation of modules even more. ALOE session 3 shows how to use the original skeleton; the next update will also describe the new tool.

3.3. Computing Resource Management Framework

The computing resource management (CRM) framework is an outcome of a Ph.D. dissertation [4]. Based on a theoretical study and analysis of available computing resource management approaches, we found that different aspects and objectives need to be considered in SDR. Continuous data streams, hard real-time processing requirements, flexibility and radio link adaptability describe the SDR context. Our framework is specifically designed for software-defined and cognitive radios. It features a computing system modeling approach and different resource allocation algorithms. The C implementation is downloadable from [2] and can be used for research and educational purposes.

3.3.1. API and Simulation Environment

An important feature of the CRM framework is its modularity. This enables addressing only part of the framework and easily introducing and analyzing modifications.

The framework features an application programming interface (API). The API defines the computing system modeling structure, the mapping and cost function parameters, and the presentation of result. Using this API, new mapping algorithms can be designed and easily embedded within the framework.

The modeling matrices abstract the processing and interprocessor bandwidth resources on the one hand, and the processing and data flow requirements, on the other. All models implicitly consider the computing resource *time*. Several models of different waveforms and processing clusters are available in different C functions. These models can be used for testing the performance of new algorithms, for instance. Some models are based on execution time measurements of waveform implementations using ALOE. The set of models is continuously extended or refined.

The cost function is independent of the mapping algorithm. It defines the management objectives and guides the mapping process. The cost function accounts for the computing system constraints. It can be considered as an interface between the system models and the general-purpose mapping algorithm. A two-term cost function is implemented as individual functions with specific parameters. The parameter set and the cost function itself can thus be easily extended or modified (see Section 4.3 – SDR project, Fall Term 2010/11).

3.3.2. Matlab Interface

The Matlab interface permits creating custom SDR application and platform models in Matlab and passing them to the CRM framework. The mapping results can be directly processed in Matlab. This is convenient for creating new waveform and platform models and testing the mapping performance without the need for modifying and recompiling the CRM framework.

The interface is a binary MEX-file. A binary MEX-file is a Matlab subroutine that is dynamically linked to a C or C++ subroutine. This permits accessing C/C++ functions from Matlab scripts. Since programs are compiled using a C/C++ compiler for a specific processor, their execution is typically much faster than the execution of equivalent Matlab routines. The MEX-file is called like any other Matlab function. It takes as its inputs the computing system modelling matrices and resource management parameters. The output features the mapping outcome and related parameters for presenting and analyzing the results (Section 4.3).

4. WIRELESS COMMUNICATIONS PROJECTS

The above SDR tools were applied in different student projects. Many of these projects were developed within the scope of the *Wireless Communications* subject at EETAC. These tools facilitate an easier project start achieving interesting results. The students themselves helped improving or extending the tool set. Below we describe three recent SDR projects.

4.1. Computing and radio QoS performance

We try to define projects with a challenging objective that include practical implementation issues. We then divide the project into several subprojects. Depending of the nature of the subprojects we schedule them in single or consecutive semesters. Students deal with the bit and symbol level processing parts, real-time processing issues, or a mix of these while analyzing a particular wireless standard. The division in subprojects has several advantages, including

- understanding the need for cooperation between different team members or teams,
- understanding the need for good documentation, and
- building a complete processing chain of a wireless standards by joining several subprojects.

The main objective of this project is evaluating the relationship between the computing resource requirements and the radio quality of service (QoS). A group of 6 students initiated this activity. The analysis considers the processing chain of the long term evolution (LTE) physical layer

(PHY). An LTE-like base station should then be developed using a set of powerful PCs and a high-speed data converter. The project thus combines radio QoS parameters with real-time computing challenges.

A suitable framework needs to be set up for the students to be able to analyze the radio performance and computing implications of the processing modules and chains they develop. The framework should facilitate observing the real-time constraints and the consequences of their violations.

The ALOE framework and tools allow defining these kinds of projects with challenging scenarios. Students take benefit of the ALOE sessions to set up and execute a scenario, define the processing chain, and visualize the parameter evolution of the module or waveform under test. ALOE sessions 1 to 5 are a good starting points and require no more than a week or two of student dedication (2 or 3 hours per session).

The tools facilitate acquiring a better understanding of the computing implications of modern radio systems as well as the need for using multiprocessor execution environments and flexible management approaches. The students directly experience the available mechanisms for enabling a coordinated execution on a multiprocessor platform. After identifying the real-time processing constraints of LTE, in this case, the students can use their acquired knowledge from ALOE Session 3 for developing new waveform modules for ALOE.

One project team focused on the performance analysis and the computing requirements of the LTE turbo decoder. They chose the turbo coder/decoder source code library from it++ [7] for developing two new ALOE waveform modules, the turbo encoder and decoder. After introducing these modules in the bit-level processing chain measurement results such as those of Fig. 1 were obtained. Figure 1 shows the bit-error rate (BER) as function of the signal-to-noise ratio (SNR) and the number of decoding iterations.

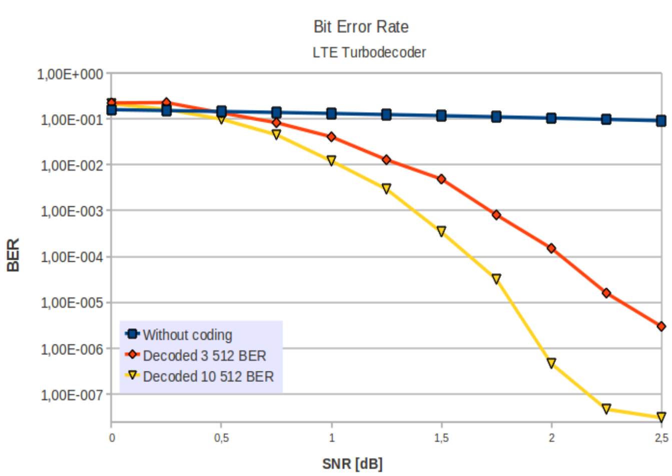

Fig. 1. BER as function of SNR and the number of turbo decoder iterations.

4.2. SDR Computing Resource Management

Another project dealt with SDR computing resource management. Within this project, a group of six students were developing three sub-projects:

- Mapping algorithms
- Waveform modeling
- Scheduling simulator

After a short presentation on SDR and the project goals, the students did ALOE Session 7 on their own, including the exercises. The exercises proved to be very useful, because they made the students think and discuss possible solutions before presenting their results to the professor. Additional exercises, derived from ALOE Session 7, helped understanding the available mapping algorithms (*tw*-, *gw*-, and *opt*-mapping) as a basis for introducing modifications.

ALOE provides a tool for measuring the execution time of waveform modules executed. Waveforms are executed in a pipelined fashion, where each processing module executes exactly once per time slot of 10 ms, for instance. We provided the execution time measurements for 27 processing blocks over approximately 100,000 time slots. The processing blocks ensemble a bit-level digital signal processing chain of a single user UMTS transmitter and receiver. These measurements were taken on a processor with a capacity of 600 million multiply-accumulate operations (MACs) per second. With this information, the students were able to create a new waveform model (*umts2_models.c*) using the metrics million operations per time slot (MOPTS) and mega-bits per time slot (MBPTS). They understood

- the pipelined execution pattern,
- the real-time processing implications,
- the modeling metrics MOPTS and MBPTS, derived from million operations per second (MOPS) and mega-bits per second (Mbps),
- the usefulness of the MAC as a suitable signal processing metric, and
- the concept of measuring processing requirements and capacities.

In practically deployed SDR systems, execution time variations may occur because of transmission parameter or channel state variations, among others. Rather than assuming the worst case requirements, the mean processing requirements were chosen here for the waveform modeling.

Two students processed the time measurements and created a Matlab routine, which generates random data that approximates the statistical behavior of the real data (Fig. 2 at the end of the paper). The variations in the processing time variations from time slot to time slot were assumed for analyzing the robustness of the mapping solution. Figure 3 at the end of the paper indicates how the processing time variations may compromise the continuous real-time processing of data packets. Results have shown that a higher window size *w*—the *tw*- and *gw*-mapping algorithm parameter [4]—distributes the processing load more evenly and so can better absorb the runtime variations in the processing time of waveform modules.

The scheduling principle and the capability of ALOE for killing processes that consume more than the allocated resources has implications on the number of data frames that will eventually be lost (or that need to be retransmitted). This, however, remained for future studies.

The Matlab interface, which was used in the third part of the project after being introduced in the previous version of ALOE Session 7, was used for defining custom waveform or platform models and presenting and reasoning the mapping results. For instance, infeasible mappings are observed if defining insufficient processing resources. Nevertheless, the then available tools did not provide insights in the mapping process itself, only the result. Additional tools were, therefore, developed in the following SDR project.

4.3. SDR Computing Resource Management Analysis and Tools

The objectives of this project were modifying the cost function for achieving a better mapping performance, studying the computing resource management requirements of the LTE PHY, including the results in the new CRM API and framework, and developing new CRM tools for educational purposes.

The cost function was modified by introducing an additional weighting parameter to be automatically set as a function of the mapping problem or state. This required studying the cost function module and understanding the modeling matrices and parameters, rather than the mapping algorithm itself. Again, ALOE Session 7 and the modular CRM framework were useful for gaining the first insights. The result was a set of parameters to be chosen either statically or dynamically as a function of the total requirements and resources of each type [8].

The students' interest in designing additional tools and experiencing new possibilities in Matlab led to new visualization tools, including the animation of the *tw*-mapping process for the given computing resource management problem and parameters (Fig. 4). Other figures plot the remaining computing resources, the scheduling chart, the processor occupation and waveform distribution. They enable a better comprehension and postprocessing of results. These tools are available for download from [2]. They are best experienced downloading and trying ALOE Session 8.

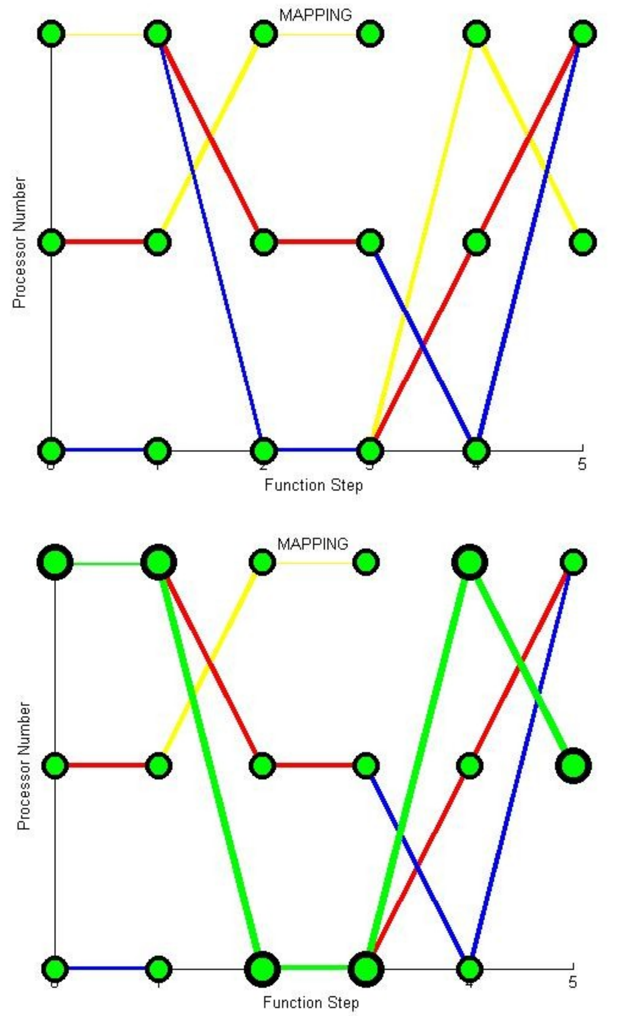

Fig. 4 Matlab tool illustrating the t_2-mapping process for a mapping problem example with 3 processors and 6 waveform modules: The upper subfigure shows the mapping decisions at each node (part II of the algorithm [4]). The lower subfigure indicates the forward and backtracking (*tw*-mapping, part III) from the minimum-cost node {0, 4}.

5. LESSONS LEARNED

The SDR concept is introduced to the students of EETAC in the undergraduate compulsory course *Digital Techniques for Communications* [1]. About half the students developing SDR projects in their Master studies have coursed *Digital Techniques for Communications*. The SDR projects offered within the scope of the *Wireless Communications* course provide further insights into modern wireless communications. For most students this is an opportunity for understanding current research issues and providing some contributions.

We have obtained different feedback and gained some experiences throughout the years while tutoring SDR projects. Most experience and feedback should be considered as positive, because it helped us improve the tools and tutoring principles taking into account the students' needs or difficulties. We, for example, found that ALOE and some of its initial tools were too complex for a 4^{th} year electrical engineering student. We therefore continuously simplify the tools and encourage our students for helping us in doing so. The students' comments helped improving the contents and readability of many ALOE Sessions, in particular.

The ALOE Skeleton is a good example for a tool that developed from a basic facility to a more user-friendly tool. The t_w-mapping algorithm was until recently explained solely with power point slides and pen-and-pencil exercises. The new tool described in Section 4.3 enables following and analyzing the mapping decisions. This tool visually demonstrates the need for distributing the computing load in case of tight resource constraints. It enables observing and understanding the different (sub-optimal) solutions to an optimization problem.

One of the lessons learned during the years of tutoring SDR student projects is that software tools are helpful for facilitating education and first research efforts. These tools should be as easy to use as possible, assume simple development rules and clear working approaches. They should also allow building complex systems and setting up different scenarios with little changes.

Despite the different students' skills, the group should work together on a common goal in order to achieve the project's objectives. The tools should therefore provide an adequate level of information that allows progressing and carrying out profitable work in a team. This, of course, is not possible using software tools only. We sometimes ask our students to solve a problem before presenting the corresponding tools. The week after we present the tools and ask the students to verify their previously obtained outcomes.

Despite our effort of making SDR accessible to undergraduate and graduate electrical engineering students, we observed that a considerable number of students does not take advantage of these opportunities. We think that the students' attitude and motivation is essential for a successful education. We, therefore, try to motivate our students, offering interesting projects and the possibility of coauthoring public documents or conference papers. We, though, think that the students should lead and develop their projects themselves and discuss their partial solutions with the professor on weekly basis, for example, rather than awaiting instructions. Unfortunately, we often do not receive the expected inputs, obliging us to lead, rather than follow or tutor the projects in order to achieve decent results. Our experience is that a high motivation effort is necessary, in general, for pushing the students to investigate, experience, and eventually learn something on their own.

We recently sent out a small questionnaire for archiving the students' opinion about the SDR tools and teaching methodology. The students mentioned the usefulness of the FlexNets web site and the ALOE tools they used while developing their projects. The weekly assignments were also appreciated for developing solutions to sub-problems before integrating these solutions into a single project. They suggest developing additional visual tools, such as the t_w-mapping process visualization tool of Fig. 4. We, therefore, work on designing graphical user interfaces for using our CRM framework and following the simulation process and partial results.

6. CONCLUSIONS

This paper presented a few SDR educational tools employed by electrical engineering students at EETAC. These tools were developed under the FlexNets initiative, a portal for collaborative SDR research and education. They are available for free download from [2]. We described a few recent SDR student projects with special emphasis on how the tools helped understanding SDR concepts and research issues for developing the semester-long projects. These projects were developed as part of the *Wireless Communications* course of the Master Degree in Telecommunication Systems Engineering.

We should mention that other Universities provide their own tools for SDR education. The Mobile and Portable Radio Research Group of Virginia Tech, for example, develops the SCA-enabled SDR framework OSSIE [5]. OSSIE is applied as part of a graduate engineering course at VT and are has been the basis of several student projects worldwide.

ACKNOWLEDGMENT

This work was supported by Spanish Government (MICINN) and FEDER under project TEC2008-06684-C03-03. We thank our students for their collaboration. Without their efforts, most of the presented tools and results would not be available today.

REFERENCES

[1] V. Marojevic, et al., "SDR in undergraduate engineering education," Proc. SDR'10 Technical Conference and Product Exposition, 30 Nov – 3 Dec 2010, Washington D.C.

[2] The Flexible Wireless Communications Systems and Networks (FlexNets) Web Site, http://flexnets.upc.edu/trac/

[3] I. Gomez, V. Marojevic, A. Gelonch, "ALOE: an open-source SDR framework with cognitive computing resource management capabilities," *IEEE Commun. Mag.*, Sept. 2011.

[4] V. Marojevic, "Computing resource management in software-defined and cognitive radios," Ph.D. Dissertation, Universitat Politècica de Catalunya (UPC), Barcelona, July 2009. Available at http://flexnets.upc.edu/trac/wiki/Publications

[5] The OSSIE (SCA-Based Open Source Software Defined Radio) Web Site, http://ossie.wireless.vt.edu/

[6] The GNU Radio Web Site, http://gnuradio.org/redmine/projects/gnuradio/wiki

[7] itt++ Web Site, http://itpp.sourceforge.net/current/

[8] A. Allujami, S. Lluch, I. Francisco, E. Marin, R. Nieto, "SDR Computing Resource Management Analysis and Tools," CSF11 Final Report, EETAC, July 2011.

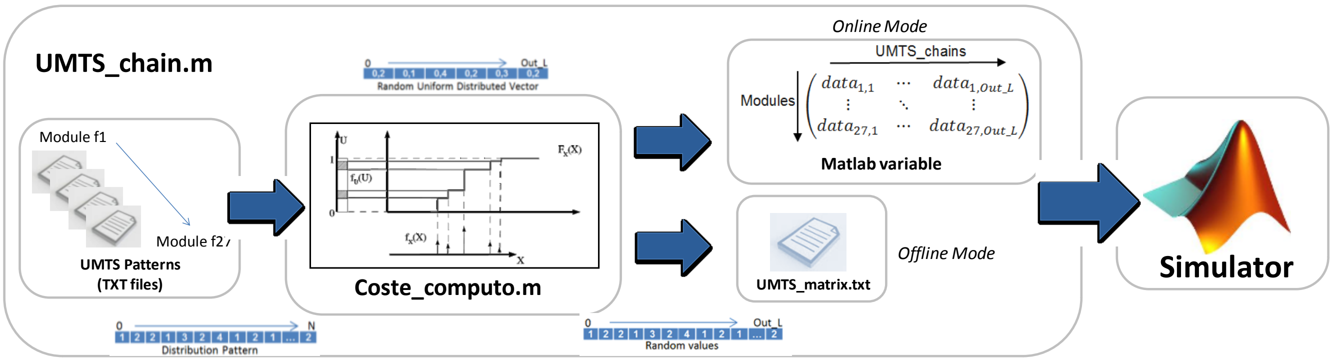

Fig. 2 Generation of random data with the same statistical behavior as real measurements.

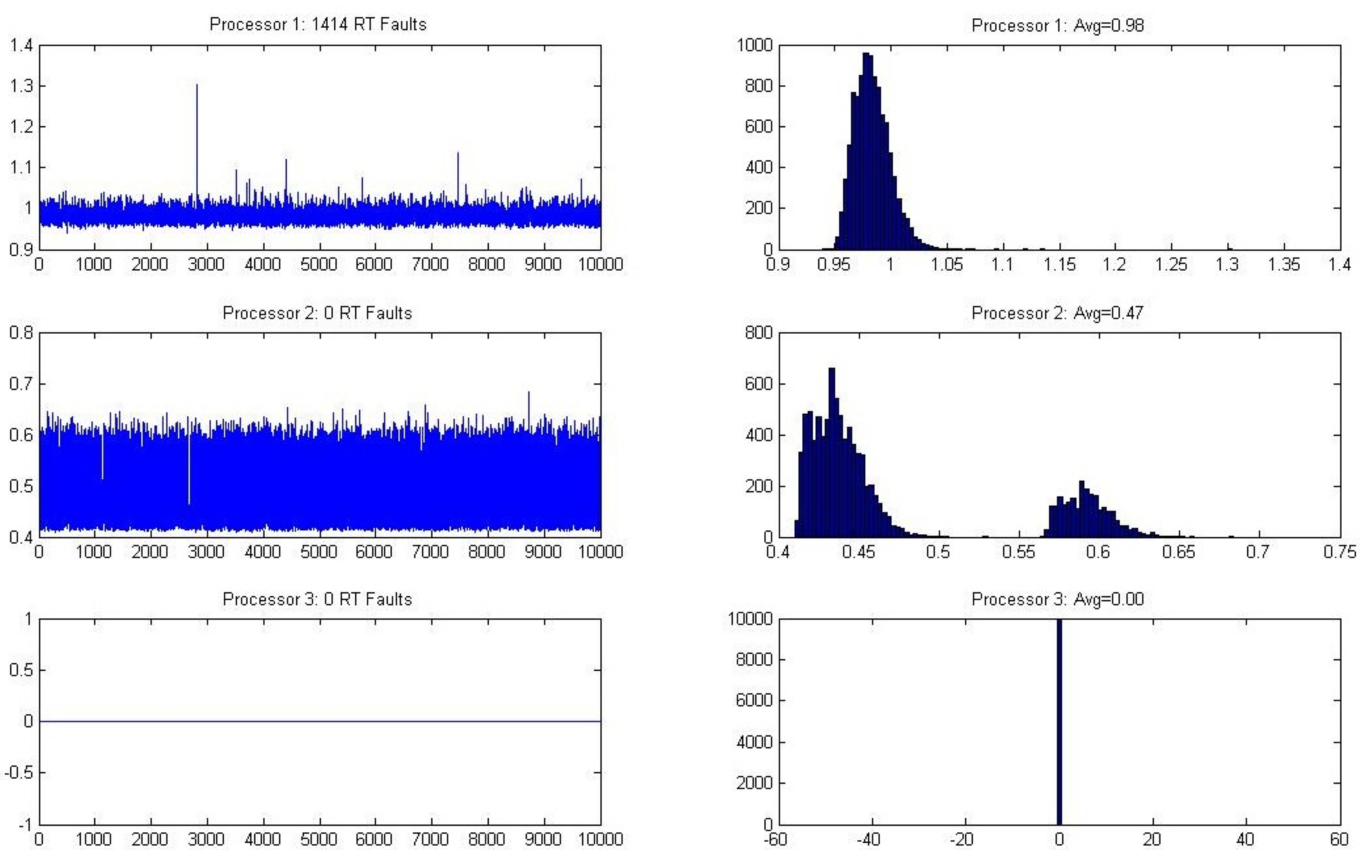

Fig. 3 Runtime variations of the waveform processing resource consumption for a UMTS bit-rate transmitter and receiver waveform model of 27 processes and a platform model of 3 processors. A processor occupation greater than 1 (left subfigures) indicates that some waveform modules will not finish the required processing of the corresponding data packet, leading to packet losses. The simulation time span was 10,000 time slots. 1414 or 14.14 % real time (RT) faults are measured for the t_2-mapping (left upper subfigure). Almost 3700 RT failures were observed for the t_1-mapping and only 200 or 2 % for the t_5-mapping.