İSTANBUL UNIVERSITY PRESS

RESEARCH ARTICLE

Electrica 2018; 18(2): 210-217

ECEbuntu - An Innovative and Multi-Purpose Educational Operating System for Electrical and Computer Engineering Undergraduate Courses

Bilal Wajid[1], Ali Rıza Ekti[2], Mustafa Kamal AlShawaqfeh[3]

[1]Department of Electrical Engineering, University of Engineering and Technology, Lahore, Pakistan
[2]Department of Electrical-Electronics Engineering, Balıkesir University School of Engineering, Balıkesir, Turkey
[3]School of Electrical Engineering and Information Technology, German Jordanian University, Amman, Jordan

Cite this article as: B. Wajid, A. R. Ekti, M. K. AlShawaqfeh, "ECEbuntu - An Innovative and Multi-Purpose Educational Operating System for Electrical and Computer Engineering Undergraduate Courses", *Electrica*, vol. 18, no: 2, pp. 210-217, 2018.

ABSTRACT

ECEbuntu is a free, easily distributable, customized operating system based on Ubuntu 12.04 long term support (LTS) designed for electrical/electronic and computer engineering (ECE) students. ECEbuntu is aimed at universities and students as it represents a cohesive environment integrating more than 30 pre-installed software and packages all catering to undergraduate coursework offered in ECE and Computer Science (CS) programs. ECEbuntu supports a wide range of tools for programming, circuit analysis, printed circuit board design, mathematical and numerical analysis, network analysis, and RF and microwave transmitter design. ECEbuntu is free and effective alternative to the existing costly and copyrighted software packages. ECEbuntu attempts to reduce the duplication of efforts on building software workstations in laboratories and is intended to serve as a good teaching resource in a classroom setting.

Keywords: Electrical and computer engineering, Ubuntu, circuit design, programming micro-controllers, microwave and RF transmission line analysis, computer networks, numerical computation and visualization, PCB design, computer programming, remote access, latex

Introduction

A school, college or university can deliver good educational programs by adopting intelligent software platforms that promote effective and good teaching strategies. Engineering as a discipline assumes an applied nature and engineering students are often required to design, implement and test systems and circuits. Such a hands-on experience can be offered on a large scale by adopting free-distribution software packages that allow students to access programming tools, circuit design and analysis tools, and mathematical and numerical analysis tools. For efficient transmission of knowledge and information, it is highly recommended that any course make use of the following teaching modalities:

i) visuals (demonstrations, descriptions, words, pictures)
ii) auditory sensations (dialogues, discussions, working out the problem audibly)
iii) tactile perceptions (taking notes, labs, hands-on work)
iv) kinesthetic impressions (movement).

Including all the above modalities in a course or laboratory session promotes a better learning experience as the student's learning faculties are distributed across all the four modalities.

Recent developments in online platforms and courses have produced significant progress towards global education. For instance, Khan Academy provides a broad range of topics in Mathematics, Science, Arts and Humanities for primary, secondary and high-school education, all for free. MIT Open Courseware (MIT-OCW) provides online lectures, assignments, exams and solutions for undergraduate and professional courses in Engineering, Architecture and Planning, Sciences, Humanities, Arts, Social Sciences and even Management. Other universities have also started to provide free online coursework.

Corresponding Author:
Ali Rıza Ekti

E-mail:
arekti@balikesir.edu.tr

Received: 14.01.2018

Accepted: 26.02.2018

DOI: 10.5152/iujeee.2018.1820

In terms of engineering education, the above mentioned online courses do a very good job of engaging the visual and auditory faculties of students. However, the tactile and kinesthetic impressions are limited, primarily because engineering courses are often associated with lab work. The lab work requires resources and software packages that often are not available for free, and therefore, limits in-depth understanding of some courses. However, within the Electrical and Computer Engineering (ECE) and Computer Science (CS) curricula, the online courses do offer the possibility of engaging the students in all four modalities for improved learning. To compensate for the lack of a free and comprehensive educational software for ECE and CS students, this paper presents ECEbuntu, a specially designed Ubuntu based educational operating system encompassing software which is useful for the training of undergraduate students from the ECE and CS Programs.

ECEbuntu

ECEbuntu was designed to help faculty engage students in all four learning modalities. As an addendum to online courses, ECEbuntu may be used by faculty to test students' learning via lab works, projects and homework assignments, all within the framework of the same software package. ECEbuntu may also be used effectively in a traditional face-to-face learning environment as all four years of undergraduate course work have been covered. Courses covered include circuit analysis, analog and digital electronics, pcb design, computer programming, micro-controllers, computer networks, microwave and radio frequency (RF) transmission line analysis and numerical computations. In addition, Latex tools like Texmaker and Tex Live, Putty for remote access, and popular tools like Google Chromium browser and VLC media player have also been added. Details of software have been referred to in Table 1, 2; (main manuscript) and Section 4, Supplementary section.

ECEbuntu was designed to facilitate free global education for all. Since most courses in ECE require technical software that is expensive, students are either expected to spend increased amounts of time in labs to complete their work or pay for software themselves to have their own personal copy. Unfortunately, most students end up pirating software. Software piracy is especially high in countries where students cannot afford to buy software or the universities that do not have a 24 hour open lab policy [1]. Table 3 and Figure 1 provide a glimpse into the gravity of this situation. Therefore, to alleviate some of these problems ECEbuntu have been made free for everyone to use, modify, update and distribute. Should universities or labs choose to use ECEbuntu as their primary educational OS it will help them to significantly reduce cost because all software dependencies within ECEbuntu have been resolved and all the technical software packaged is free. Naturally, some of these free products are not as professional as their equivalent costly counterparts; however, they do fill in the gap very nicely.

Additionally, ECEbuntu allows for easy distribution as there are no copyright restrictions. This is notable for countries where power outages are a daily routine and the download bandwidth is limited [2, 3]. Table 3 shows some countries where such complications (limited bandwidth, power outages) persist. ECEbuntu can be easily distributed using DVD or USB stick (see Sections 1 and 2 in the Supplementary Section). Furthermore, virtualization ECEbuntu may be used in parallel with Windows or MAC operating systems, though it is recommended that users install ECEbuntu separately to ensure maximum usage of memory and processing power (see Section 3, Supplementary Section for further details).

Discussion

In order to integrate all four learning modalities (visuals, auditory, tactile and kinesthetics) in formal coursework, institutions adopt one of two frameworks. i) Students bring their own devices (BYOD) and download/install/maintain necessary software based on the requirements laid out in the syllabus, ii) The labs provide the necessary framework in which the students can work. The server broadcasts and installs software to all workstations. Additionally, the lab-manager is in charge of maintaining/updating the OS and software installed on all workstations. As BYOD approach is generally costly for engineering, our motivation was to simplify routine activities of lab-managers by taking part of the responsibilities onto ourselves. These include searching for potential freeware and integrating the best software as a cohesive Ubuntu 12.04 LTS based platform for ECE/CS education. Ubuntu LTS was carefully chosen because the Ubuntu community is committed to maintaining Ubuntu

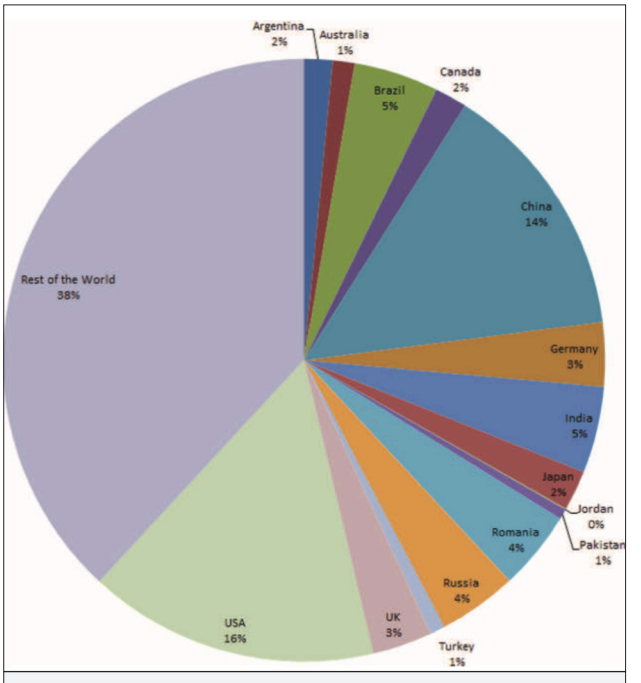

Figure 1. Percentage of Profit Loss per Country for Using Pirated Software

Table 1. Software tools in ECEbuntu

Name(s) of the Course(s)	Software tools
Electrical Circuit Theory	Fritzing: Electronic design software [4].
Introduction to Digital System Design	gEDA: Electronic design software [5].
Semiconductor Devices	GTKWave: Waveform viewer
Analog Electronic Circuits	Gwave: waveform viewer e.g., for spice simulators [6].
Operational Amplifiers	Kicad: Electronic schematic and PCB design [7, 8].
Power Electronics	Oregano: Tool for schematical capture of electronic circuits [9, 10].
VLSI Circuit Design	PCB: Printed circuit board design program (http://pcb.gpleda.org/).
PCB Design	Visolate: Tool for engraving PCBs using CNC machine (https://sourceforge.net/projects/visolate/).
Programming -controllers	Emu8051: Emulator and simulator for 8051 -controllers [11].
	MCU8051: IDE for MCS-51 based micro-controllers (https://sourceforge.net/projects/mcu8051ide/).
	SPIM: MIPS R2000/R3000 emulator (http://pages.cs.wisc.edu/~larus/spim.html).
Electromagnetic Field Theory	LinSmith: Tool to generate Smith Charts (http://jcoppens.com/soft/linsmith/index.en.php).
Microwave and RF transmission line analysis	TransCalc: Microwave and RF transmission line calculator (http://transcalc.sourceforge.net/).
Antenna Engineering	
Computer Networks/Network Analysis	Dynamips: Cisco 7200/3600/3725/3745/2600/1700 router emulator [12, 13].
	GNS3: Graphical network simulator [14, 15].
	NetEmul: Program for simulating computer networks (http://netemul.sourceforge.net/).
	Putty: Telnet/SSH client for X (http://www.putty.org/).
Random Signals and Systems	GeoGebra: Dynamic mathematics software for education [16, 17].
Stochastic Processes	GNU Octave: High level programming language, primarily intended for numerical computations
Signals and Systems	KmPlot: Mathematical function plotter for KDE (https://edu.kde.org/kmplot/).
Digital Signal Processing	QtOctave: Qt front-end to Octave. An environment for numerical computations [18, 19].
Linear Control Systems	RKWard: KDE frontend to the R statistics language [20, 21].
Digital Image Processing	SciLab: Scientific software package for numerical computations [22-24].
Numerical Analysis	Spatial Statistics: GNU R package for spatial statistics [25].
Programming	Eclipse: for C/C++ and Java (may be updated for Andriod applications) [25-28].
Data Structures Design and Analysis of Algorithms	IDLE: IDE for Python using Tkinter (http://www.python.org/).
Technical Report Writing	Texmaker: Cross-platform LaTeX editor (http://www.xm1math.net/texmaker/).
	Tex Live: A decent selection of the TeX Live package (http://www.tug.org/texlive/).

Table 2. Software tools in ECEbuntu can cater to all four years of the engineering program. Some of the tools are suitable for all four years of the program and are therefore mentioned more than once

Year	Name(s) of the Course(s)	Software tools
Freshman Year	Electrical Circuit Theory	Eclipse: for C/C++ and Java (may be updated for other languages) [26, 27].
	Semiconductor Devices	Fritzing: Electronic design software [4].
	PCB Design	gEDA: Electronic design software [5].
	Computer Fundamentals	IDLE: IDE for Python (http://www.python.org/).
		Kicad: Electronic schematic and PCB design [7, 8].
		Oregano: Tool for electronic circuits [9, 10].
		PCB: Printed circuit board design (http://pcb.gpleda.org/).
		Visolate: Tool for engraving PCBs using CNC machine (https://sourceforge.net/projects/visolate/).
Sophomore Year	Electrical Circuit Theory	Eclipse: for C/C++ and Java
	Introduction to Digital System Design	Fritzing: Electronic design software [4].
	Semiconductor Devices	gEDA: Electronic design software [5].
	Analog Electronic Circuit	IDLE: IDE for Python (http://www.python.org/).
	Programming Languages	Kicad: Electronic schematic and PCB design [7, 8].
	Data Structure	Oregano: Tool for schematical capture of electronic circuits [9,10].
	Design and Analysis of Algorithms	PCB: Printed circuit board design (http://pcb.gpleda.org/).
	Writing Skills	Texmaker: Cross-platform LaTeX editor.
		Tex Live: A decent selection of the TeX Live package
		Visolate: Tool for engraving PCBs using CNC machine (https://sourceforge.net/projects/visolate/).
Junior Year	Operational Amplifier	Emu8051: IDE for 8051 -controllers [11].
	Programming µ-Controllers	Fritzing: IDE for Electronic system design [4].
	Random Signals and Systems	KmPlot: Mathematical function plotter for KDE (https://projects.kde.org/projects/kde/kdeedu/kmplot).
	Stochastic Processes	LinSmith: Tool to generate Smith Charts (http://jcoppens.com/soft/linsmith/index.en.php).
	Signals and Systems	gEDA: Electronic design software [5].
	Numerical Analysis	GeoGebra: Dynamic mathematics software for education [16, 17, 29].
	Electromagnetic Field Theory	GNU Octave: programming language intended for numerical analysis.
		Kicad: Electronic schematic and PCB design [7,8].
		MCU8051: IDE for MCS-51 -controllers (http://mcu8051ide.sourceforge.net/).
		Oregano: IDE for electronic circuits [9, 10].
		PCB: IDE for PCBs (http://pcb.gpleda.org/).
		QtOctave: IDE for numerical analysis [18, 19].
		RKWard: KDE frontend to the R statistics language [20, 21].
		SciLab: IDE for numerical computations [22, 23].
		Spatial Statistics: GNU R package for spatial statistics [25].
		SPIM: MIPS R2000/R3000 emulator (http://pages.cs.wisc.edu/~larus/spim.html).
		TransCalc: Microwave and RF transmission line calculator (http://transcalc.sourceforge.net/).
		Visolate: Tool for engraving PCBs using CNC machine (https://sourceforge.net/projects/visolate/).
Senior Year	VLSI Circuit Design	Dynamips: Cisco 7200/3600/3725/3745/2600/1700 router emulator [12, 13].
	Power Electronics	Fritzing: IDE for Electronic design [4].
	Linear Control	gEDA: IDE for Electronic design [5].
	Computer Networks/ Network Analysis	GeoGebra: IDE for numerical analysis [16, 17].

Table 2. Software tools in ECEbuntu can cater to all four years of the engineering program. Some of the tools are suitable for all four years of the program and are therefore mentioned more than once (Continue)

Year	Name(s) of the Course(s)	Software tools
	Antenna Engineering	GNS3: Graphical network simulator [14, 15].
	Microwave and RF transmission line analysis	GNU Octave: programming language for numerical analysis.
	Digital Signal Processing	Kicad: IDE for electronic design [7, 8].
	Image Processing	LinSmith: Tool to generate Smith Charts (http://jcoppens.com/soft/linsmith/index.en.php).
		NetEmul: Program for simulating computer networks (http://netemul.sourceforge.net/).
		Oregano: IDE for electronics design [9, 10].
		PCB: IDE for PCBs (http://pcb.gpleda.org/).
		Putty: Telnet/SSH client for X (http://www.putty.org/).
		QtOctave: Qt front-end to Octave. An environment for numerical computations [18, 19].
		RKWard: KDE frontend to the R statistics language [20, 21].
		SciLab: Scientific software package for numerical computations [22, 23].
		Spatial Statistics: GNU R package for spatial statistics [25].
		TransCalc: Microwave and RF transmission line calculator (http://transcalc.sourceforge.net/).
		Visolate: Tool for engraving PCBs using CNC machine (https://sourceforge.net/projects/visolate/).

Table 3. Statistical Values of Percentage of Piracy, Value of Piracy, Average Bandwidth per Country, Electricity Outages Days and Durations per Country [1-3]

	Argentina	Australia	Brazil	Canada	China	Germany	India	Japan	Jordan	Pakistan	Romania	Russia	Turkey	U.K.	U.S.A.
% of Pirated Software Usage	69	21	50	25	74	24	60	19	57	85	62	62	60	24	18
Value of the Pirated Software ($Million)	950	743	2851	1089	8767	2158	2911	1349	35	344	208	2658	504	2019	9773
Average Internet Broadband Connection Speed Per User (mbps)	4.2	6.9	2.9	10.3	3.8	8.7	2	15	3.09	2.31	11.3	9.1	5.5	10.7	11.5
Number of days per month with electricity outages	1.9	0.4	1.6	NA	0.1	NA	13.8	NA	0.2	30	1.4	0.3	1.7	NA	NA
Average number of hours per electricity outage	1.2	1	1.8	NA	0.5	0.8	1.1	NA	0.2	2.1	1.1	0.9	0.7	NA	NA

Table 4. Comparison of Different Linux Distributions: The table compares different linux distributions.

Operating system	Free	Reliable	Base OS	Software	Open source	LTS	GUI	Security	Threat detection	86/64	Cloud
Baari	√	√	Ubuntu 13.10	30+ Genome Assembly tools	√	√	Unity	√	√	64	x
Lxtoo	√	√	Gentoo Linux 11	Sequence Analysis, Protien-Protien interactions	√	√	X11 Desktop	√	√	x86/64	x
Open Discovery 3	x	√	Fedora Sulphur 9	molecular dynamics, docking, sequence analysis	√	√	GNOME 2.22	√	√	86/64	√
PhyLIS	√	√	Ubuntu 8	Phylogenetics	√	x	Unity	√	√	86/64	x
DNALinux	√	√	Xubuntu	DNA and protein analysis. Also contains Virtual Desktop	√	x	XFCE 4.2.2	√	√	86	√
BioLinux 7	√	√	Ubuntu 12.04	500+ Bioinformatics application with 7 Assembly tools	√	√	Unity	√	√	64	√
ECEbuntu	√	√	Ubuntu 12.04	OS for ECE/CS education	√	√	Unity	√	√	64	x

Table 5. Questionnaire: Please respond to the following statements by using the 5-point rating scale to indicate the extent to which you agree or disagree with each statement. Please circle the number that applies

S. No.	Question	5 = Strongly Agree	4 = Agree	3 = Neutral	2 = Disagree	1 = Strongly Disagree
1	Objectives of ECEbuntu are stated clearly and met.	1	2	3	4	5
2	The information provided by ECEbuntu was relevant and useful.	1	2	3	4	5
3	ECEbuntu facilitated my learning of Electrical and Computer Engineering.	1	2	3	4	5
4	The instruction manual is well written, organized and fulfils its purpose.	1	2	3	4	5
5	It is easy to install ECEbuntu.	1	2	3	4	5
6	More software and packages should be added to ECEbuntu	1	2	3	4	5
7	ECEbuntu's interface is user friendly.	1	2	3	4	5
8	ECEbuntu is successful in performing its intended task.	1	2	3	4	5
9	ECEbuntu is an appropriate OS for courses in ECE.	1	2	3	4	5
10	Given that ECEbuntu is free, will you recommend ECEbuntu to your colleagues?	1	2	3	4	5
11	What do you like best about this software?					
12	Do you think it is a good software tool to teach graduate students and researchers?					
13	Areas/topics about which you would like to receive further software?					
14	What did you think was the most important feature introduced by ECEbuntu?					
15	What do you dislike about ECEbuntu?					
16	Did you experience any problem with this software? If yes, what kind of problem did you have?					
17	Suggestions for improving ECEbuntu					

12.04 OS for the long term, hence the term 'LTS' which stands for 'Long-Term-Support'. The OS maintenance routinely comes up as an update which the lab-manager has to install on the server. This automatically updates all workstations connected to the server. Furthermore, based on the recommendations of users, teachers, students and lab-managers ECEbuntu will be routinely updated to install the latest software packages and remove the ones not needed. As far as the authors are aware, ECEbuntu is a unique solution as no similar prior work has been conducted within ECE. However, similar practice involving integration of multiple software platforms has been conducted extensively in Life-Sciences. These 'Life-Linux distros' (Life Sciences Linux based OS) have saved biologists from spending increasing amount of time and resources in installing, configuring and maintaining software rather than spending the same on research. We hope that ECEbuntu will serve the same role for ECE as life-Linux distros have served for life sciences (Table 4 lists some examples). Table 5 provides a questionnaire for users to rate and give feedback on ECEbuntu.

Conclusion

We highlight a free, easily distributable, customized Ubuntu based OS that contributes to ECE/CS education. ECEbuntu attempts to fulfill the software requirements of four years of undergraduate coursework. This may help teachers who incorparate ECEbuntu into their syllabus and homework. Furthermore, installing ECEbuntu in the laboratories may help with the smooth operation of lab assignments. Additionally, ECEbuntu will continue to be routinely upgraded with the help of suggestions and feedback from faculty and students. Future work may include a software package for Windows.

Peer-review: Externally peer-reviewed

Conflict of Interest: The authors have no conflicts of interest to declare.

Financial Disclosure: The authors declared that this study has received no financial support.

References

1. "BSA Global Software Survey", http://globalstudy.bsa.org/2013/ , 2013.
2. "Infrastructure",http://www.enterprisesurveys.org/data/exploreTopics/Infrastructure, 2013.
3. "State of internet", http://www.akamai.com/stateoftheinternet/, 2014.
4. A. Knörig, R. Wettach, J. Cohen, "Fritzing: a tool for advancing electronic prototyping for designers", Proc of the 3rd Intl Conf on Tangible and Embedded Interaction, Cambridge, UK, 2009, pp. 351-358.
5. S. Brorson, "Circuit design on your Linux box using gEDA", *Linux Journal*, vol. 2006, no. 141, 2006.
6. C. Medrano, I. Plaza, M. Castro, F. Garcia-Sevilla, J. Martiinez-Calero, J. Felix, M. Corbalan, "A review of electronic engineering design free software tools", Proc of the 1st Intl Conf on IEEE Education Engineering (EDUCON), Madrid, Spain, 2010, pp. 1867-1871.
7. J. C. Borg, X. T. Yan, N. P. Juster, "A KICAD Tool for Pro-Active Exploration Support to 'Design Synthesis for Multi-X'", Knowledge Intensive Computer Aided Design, Springer, 2000, pp. 295-322.
8. R. Halvick, "KiCad: a high level tool. Electronic CAD available to all," *Elektor Electronics*, vol. 33, no. 367, p. 134, 2007.
9. B. Zapirain, A. Zorrilla, I. Ruiz, A. Muro, "Learning electronics using image processing techniques for describing circuits to blind students," Proc. of the IEEE Intl Sym Signal Processing and Information Technology (ISSPIT), Luxor, Egypt, 2010, pp. 156-160.
10. M. Neruda, L. Vojtech, "Modeling of smart textile materials for ESD applications", Proc of the IEEE ELMAR, Zadar, Croatia, 2012, pp. 145-148.
11. T. Reinbacher, D. Gückel, S. Kowalewski, M. Horauer, "Testing microcontroller software simulators", http://www.user.tu-berlin.de/komm/CD/paper/061433.pdf, 2011.
12. Y. P. Wu, W. Zheng, "Application and Analysis of Packet Tracer and Dynamips in Computer Network Simulation Experiment [J]", *Computer Era*, vol. 10, p. 8, 2010.
13. L. Wen-Chi, "On the Configuration and Application of Dynamips Virtual Network [J]," *Journal of Nanjing Institute of Industry Technology*, vol. 2, p. 17, 2007.
14. W. Makasiranondh, P. S. Maj, D. Veal, "Pedagogical evaluation of simulation tools usage in Network Technology Education," *Engineering and Technology*, vol. 8, pp. 321-326, 2010.
15. V. Autefage, D. Magoni, "Network emulator: a network virtualization testbed for overlay experimentations", Proc. of the IEEE 17th Intl. Workshop Computer Aided Modeling and Design of Communication Links and Networks (CAMAD), Barcelona, Spain, 2012, pp. 266-270.
16. M. Hohenwarter, J. Preiner, "Dynamic mathematics with GeoGebra," *Journal of Online Mathematics and its Applications*, vol. 7, 2007.
17. L. Fahlberg-Stojanovska, V. Stojanovski, "GeoGebra-Freedom to explore and learn", *Teaching Mathematics and its Applications*, vol. 28, no. 2, pp. 69-76, 2009.
18. P. L. Luis, "QtOctave, Octave para todos los públicos," Todo linux: la revista mensual para entusiastas de GNU/LINUX, no. 84, pp. 37-41, 2007.
19. M. P. L. del Castillo, P. L. L. Rosado, and A. S. Muñoz, "QtOctave: el MatLab de los pobres", Actas de las I Jornadas en Innovación y TIC Educativas-JITICE 2010, p. 29.
20. S. Rödiger, T. Friedrichsmeier, P. Kapat, M. Michalke, "Rkward: A comprehensive graphical user interface and integrated development environment for statistical analysis with r", *Journal of Statistical Software*, vol. 49, no. 9, pp. 1-34, 2012.
21. D. Wick, "Free and open-source software applications for mathematics and education", Proc. of the 21st Annual Intl Conf Technology in Collegiate Mathematics, New Orleans, LA, 2009, pp. 300-304.
22. S. L. Campbell, J. P. Chancelier, R. Nikoukhah, "Modeling and simulation in SCILAB", Springer, 2010.
23. C. Gómez, "Engineering and Scientific computing with Scilab", Springer, 1999.
24. K. Chine, "Learning math and statistics on the cloud, towards an ec2-based google docs-like portal for teaching/learning collaboratively with r and scilab", Proc. of the 10th IEEE Intl. Conf. Advanced Learning Technologies (ICALT), 2010, pp. 752-753.

25. W. Venables, B. D. Ripley, "S Programming", Springer, 2000.
26. J. desRivieres, J. Wiegand, "Eclipse: A platform for integrating development tools," *IBM Systems Journal*, vol. 43, no. 2, pp. 371-383, 2004.
27. D. Geer, "Eclipse becomes the dominant Java IDE", *IEEE Computer*, vol. 38, no. 7, pp. 16-18, 2005.
28. J. Edmondson, W. Anderson, J. Gray, J. Loyall, K. Schmid, J. White, "Next-Generation Mobile Computing", *IEEE Software*, vol. 31, no. 2, pp. 44-47, 2014.
29. M. Boule, "The role of Finite Element Method software in the teaching of electromagnetics", Proc of the 4th Interdisciplinary Engineering Design Education Conference (IEDEC), Santa Clara, CA, 2014, pp. 44-51.

Bilal Wajid completed his B.Sc. Electrical Engineering from University of Engineering and Technology (UET), Lahore, Pakistan in 2007 and his Ph.D. in Electrical Engineering from Texas A&M University (TAMU), College Station, TX, USA in 2015. He his currently serving as an Assistant Professor at Dept. of Electrical Engineering, UET, KSK Campus. He has previously taught at Texas A&M University, CS (TX, USA), Texas A&M International University, Laredo (TX, USA) and DUKE University, Durham (North Carolina, USA). His research interests include Application of Engineering in Medicine, Bioinformatics and Cloud computing.

Ali Rıza Ekti is from Tarsus, Turkey. He received B.Sc. degree in Electrical and Electronics Engineering from Mersin University, Mersin, Turkey, (September 2002-June 2006), also studied at Universidad Politechnica de Valencia, Valencia, Spain in 2004-2005, received M.Sc. degree in Electrical Engineering from the University of South Florida, Tampa, Florida (August 2008-December 2009) and received Ph.D. degree in Electrical Engineering from Department of Electrical Engineering and Computer Science at Texas A&M University (August 2010-August 2015). He is currently an assistant professor at Balikesir University Electrical and Electronics Engineering Department and also senior researcher at TUBITAK BILGEM. His current research interests include statistical signal processing, convex optimization, machine learning, resource allocation and traffic offloading in wireless communications in 4G and 5G systems and smart grid design and optimization.

Mustafa Alshawaqfeh received his B.S. degree in communication engineering from Yarmouk University, Irbid, Jordan, in 2007, M.S., degree in electrical engineering/wireless communication from the Jordan University of Science and Technology, Irbid, Jordan, in 2010, and Ph.D., degree in electrical engineering from Texas A&M University, College Station, Texas, USA. He joined the German Jordanian University since 2017 as an Assistant Professor at the electrical and communication engineering department. His research interests span the areas of wireless communications, signal processing, machine learning, and bioinformatics.