Novel Online Simulator for Education of Power Electronics and Electrical Engineering

U. Drofenik, A. Müsing, and J. W. Kolar

Power Electronic Systems Laboratory (PES), ETH Zurich, ETH-Zentrum / ETL H13, CH-8092 Zurich, Switzerland

Abstract-- **In this paper a novel online simulator is introduced that is integrated into the educational project iPES which is a free collection of Java applets for educational purpose. iPES has been translated into 12 languages and has currently more than 10,000 visits per month. The novel online simulator is optimized for power electronics, easy to use, and allows unlimited simulation of converter topologies. The concept of embedding a circuit simulator in the form of a Java applet into a webpage allows the creation of power electronic and electrical engineering courses that are easy for students to access, highly flexible and require a low administrative effort. The paper will discuss the status of the iPES project in detail including all shortcomings and will show how the integration of the online simulator significantly improves the educational value.**

Index Terms-- **education, interactive animation, numerical simulation, online simulation, Java applet.**

I. INTRODUCTION

Many typical engineering problems are complex, show dynamic behaviour and a large number of different states. It is difficult for the teacher, who is limited to verbal description and static pictures, to explain e.g. complex parameter dependencies or, as often encountered in power electronics, current conduction dependent on switching states. The student finds it difficult to understand how a system works if he is mainly passively watching in the classroom.

Traditionally, the student first calculates curves and waveforms analytically and analyses the states of a power electronics circuit and the related current paths with pencil and paper. This is of course still the best method to learn and understand the principles of power electronics. This active involvement of the student strongly contributes to his learning success.

Many circuit topologies and/or control strategies are so complicated that it is necessary to perform computer simulations to analyse details of the converter system. Numerical simulation of systems is a standard procedure in industry and this alone makes it essential to teach the student about it. Simulation is an excellent final step to the traditional approach in engineering education employing pencil and paper, and allows analysis and promotes the understanding of more complex systems. Unfortunately, power electronics simulation software is

expensive and student versions of commercial simulation software are very limited in usage. Computer working places are often scarce in academic institutions, especially if there are many students attending a course. It also takes a considerable effort from the teacher or instructor to keep the simulation software updated and ready for the student exercises.

In 2001 the Power Electronic Systems Laboratory (PES), ETH Zurich, introduced the free "Interactive Power Electronics Seminar" (iPES) [1] – [4], which is a web-based collection of interactive Java-animations discussing and explaining important details of power electronic converters. iPES permits the manipulation of certain preselected parameters of a circuit topology, model or control system, and to observe the resulting changes of internally coupled parameters. iPES topologies can be changed and/or manipulated only within given boundaries. In section II we will discuss the idea behind iPES, its development over the last eight years, and its current status.

Due to certain iPES limitations and user-feedback, the Power Electronic Systems Laboratory (PES) went one step further and integrated a fully functional power electronics circuit simulator into iPES in 2009. This free online-circuit simulator, developed in cooperation with a company specialized in power electronics simulation software, is called "GeckoCIRCUITS" [5]. In section III we explain the technological background, and the limitations, advantages and disadvantages of this novel kind of web-based online simulation.

In section IV we show how the online-version of GeckoCIRCUITS has been integrated into the student exercises at the PES, ETH Zurich. There are about 110 master course students attending courses with power electronics computer exercises every year which provides a significant administrational challenge. Online-simulation embedded in dedicated HTML pages was found to be very useful both for students and teachers.

II. INTERACTIVE POWER ELECTRONICS SEMINAR

A. The Basic Concept

One of the problems in studying or teaching power electronics is due to the large number of possible reactions of a system to parameter changes. Another difficulty in understanding power electronics topologies is that conduction states of power electronic circuits typically go through a sequence of different states also for

stationary operation. The traditional way of presenting such challenging content is by a sequence of slides showing a single system state and/or the behaviour for a set of parameters.
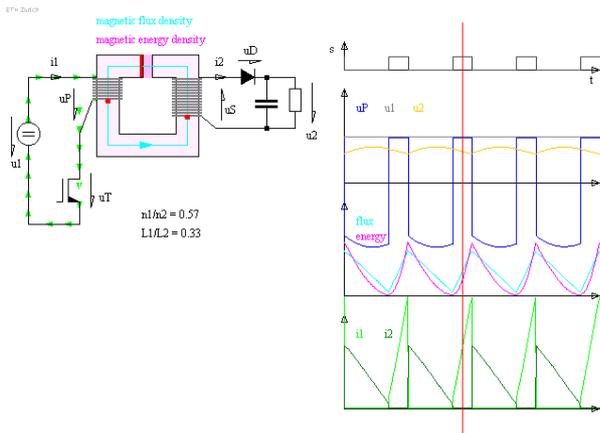
Fig. 1: iPES-Applet "Flyback Converter - Magnetic Components" [6] with current path animation. Dragging the two red dots with the mouse (located at the windings of the inductor) allows changing the turn number of the two windings. Dragging the red bar at the air gap allows the air gap length to be changed. Waveforms and distribution of the magnetic energy are updated immediately. Dragging the red time slider (vertical line in the time behaviour diagram) allows animation of the current flow, see green arrows indicating current flow in the primary windings in this screenshot.

One of the central ideas of iPES is to provide Java-based animations where a time marker ("time slider") can be dragged by the student via the mouse, where the time marker position directly shows the related system state. See the screenshot in **Fig. 1** of a Java-applet in iPES which shows a flyback converter and its magnetic component. The time slider is the vertical red line in the waveform diagrams. According to the position of the time slider the current flow of the associated path in the topology is animated employing little moving arrows.

Another basic idea of the iPES-applets is to offer the possibility of the manipulation of various system parameters via a graphical user interface, which allows showing directly the resulting time and frequency behaviour of the system. This feature allows demonstrations of tendencies of a converter system such as "increase boost inductor and see how current ripple becomes smaller". In the screenshot of the flyback converter (Fig. 1) one can change windings and air gap length of the magnetic component. The magnetic energy density within the magnetic component is shown in magenta color (see energy concentration in the air gap in Fig. 1).

Another example of one of the many iPES-applets is shown in **Fig. 2** where the basics of inductance are explained. The student can change diameter and distance of the wires, and experiment with the impact of these two parameters on magnetic field, energy distribution and resulting inductance.

One important idea is that the student can only change

a small number of parameters, ideally 1- 3, and experiment with the effects of such changes. By limiting the number of parameters the student is forced to concentrate on certain details of the topology and/or model. With a small number of parameters the student can quickly explore all possible states and/or combinations and has quickly a motivating feeling of success which is very important for studying. It was considered to be important not to overwhelm the student with too many features implemented in one applet.
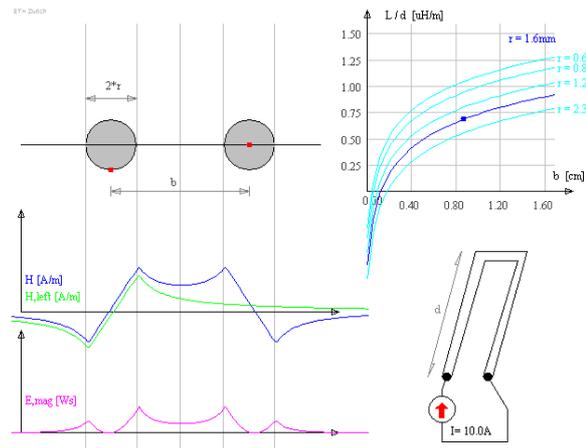
Fig. 2: iPES-Applet "Magnetomotive Force (MMF), Inductance, Magnetic Flux" [7] for demonstration of magnetic fields and inductance of a winding. By dragging the red dots with the mouse, the student can change the diameter of the wire and/or distance between the two wires. The curves of magnetic field strength (blue), the magnetic energy density (magenta) and the inductance in the graph (top – right, blue & cyan) are updated immediately.

Compared to traditional teaching approaches this way of learning promotes the active involvement of the student. Another advantage is that the student can decide on the progress of the presentation of the material which allows an individually-optimized learning speed. A further important advantage of the free web-based iPES concept is that the student is very flexible and can study at home.
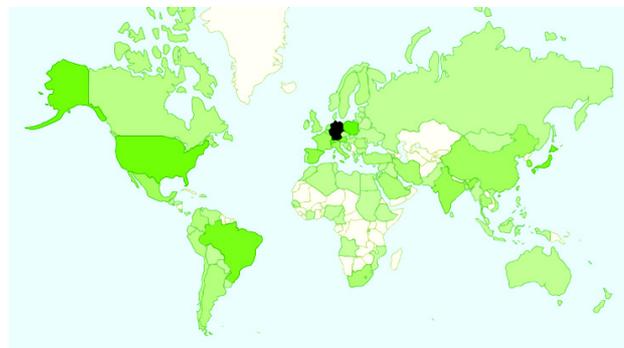
Fig. 3: The figure shows the global distribution of iPES-visitors based on 10,000 visits per month from more than 100 countries. The intensity of the green colour indicates the number of visitors per country. 30% of the visitors are from Germany (1st), 5% are from Japan (2nd).

B. Acceptance and Users

The educational platform iPES went online in August 2001. Today it has been translated into 12 languages (in alphabetic order: Arabian, Chinese, Taiwan-Chinese, Dutch, English, Farsi, French, German, Italian, Japanese, Korean, and Spanish). There are up to 500 visitors per day and on an average more than 10,000 visits per month. The distribution of the users is shown on the map in **Fig. 3** which has been created from a free statistical tool [8].

Other projects of web-based educational interactive animations have been developed in Java but also other Internet technologies like Flash, e.g. [9] – [14].

C. Shortcomings and Criticism

Applets in iPES allow only a small number of parameters to be changed by the user. This gives certain advantages concerning the educational effect (see previous section):

- The students are forced to concentrate on a certain aspect of the problem;
- The feeling of success occurs after a short time due to the simplicity;
- Concentrating on parameter relationships instead of on exact numbers.

Another (non-educational) reason for the restrictions is that parameter changes have to be simulated numerically, but the user experience should be an immediate effect. The underlying numerical simulations therefore have to be very fast, e.g. long-term transients in converter systems with low damping caused by a parameter change by the user have to be calculated extremely fast.

The technical solution when implementing the iPES-applets was to accept numerical inaccuracies by employing large numerical time steps and adding artificial damping resistors. Now the parameter changes result in waveform changes that are felt as "immediate" due to the fast but inaccurate simulation, the relationships and tendencies are all correct, but the values might show errors up to 10%. Numbers are therefore not given in many iPES-applets.

After having gained experience and understanding in power electronics with the educational iPES-applets, many users (especially teachers and instructors) have asked for "real" circuit simulation and for much more flexibility.

Introducing numerical simulation in power electronics education as a next step would be the ideal preparation before building hardware. It would also show the students that converter systems, when e.g. considering parasitics or non-ideal switches, might behave differently as compared to the textbook example or the iPES-applet (which is mainly inspired by the textbook example). That was the motivation to integrate online circuit simulation into the iPES platform as shown in the following.

III. ONLINE SIMULATION WITH GECKOCIRCUITS

A. How Does It Work and What Are the Limitations by Technology?

GeckoCIRCUITS [5] is a circuit simulator specialized for power electronics with the following features:

- Easy to learn and use
- Very fast and numerically stable
- Multi-domain: Electric - Thermal - EMI
- Easy and fast calculation of transient junction temperatures
- No installation required

GeckoCIRCUITS is written in the programming language Java which allows it to run online as a so-called "Java applet". This would not be possible in this form if it was written in e.g. C/C++.

A Java applet is a program written in Java programming language which is embedded in a webpage. When opening the webpage, the web-browser automatically loads the Java executable from the server (where the HTML page is located), checks the security of this program, and hands the Java executable to the so-called Java Runtime Environment (JRE). The JRE has to be installed on the user's computer and the executable is then running on this computer, not on the server. Currently (Nov. 2009), 92% of the visitors of the iPES-homepage have Java preinstalled and enabled in the web-browser.

Theoretically it is possible to run any kind of software as Java applet. Practically, there are important limitations:

(1) The Java program must be downloaded over the Internet. There are limits as to how long the user is willing to wait for a download. Therefore, large programs in form of Java applets will not be accepted by most users. It is essential to keep the program size small. The circuit simulator GeckoCIRCUITS has currently a total size of 1.4MB witch takes less than a few seconds with a fast Internet connection.

(2) Most users will already have the Java Runtime Environment (JRE) preinstalled, so there is no need for any additional installations. A large number of users is not willing to perform complicated installations in the web-browser to make a Java applet work. Therefore, a Java applet is practically limited to the features which are available in the standard JRE. This means that features such as 3D graphics (not part of the standard JRE) will not be accessible to most users. GeckoCIRCUITS works well with the standard JRE. One important feature, the "JAVA-block", which is a highly flexible interface to the Java programming language, is not part of the standard JRE and is, therefore, disabled in the online version of GeckoCIRCUITS. This is discussed in more detail in section IV-B.

(3) The most important restriction is the default-inability of a Java applet to access the hard disk. Therefore, a Java applet cannot do harm to the client computer, cannot manipulate or destroy data, or spy on user data on the hard disk. This guarantees strong security for the user and is a main reason for the success of Java as an Internet language. On the other hand, a Java applet cannot read a file and/or load a model from the user's hard disk, and it cannot save models locally. Of course it is possible to digitally sign and certificate Java applets in order to allow them all kind of hard disk access but certification it is a complicated procedure for the user with very low acceptance. Currently, the iPES-integrated online version of GeckoCIRCUITS does not permit the loading or saving of models locally. It only accepts prepared model files stored on the server. This is sufficient for educational purposes.
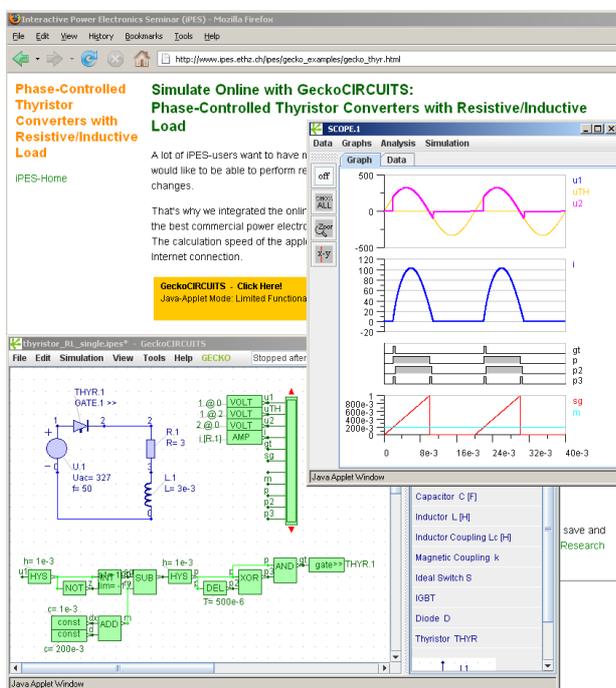
Fig. 4: Online simulation in iPES. The screenshot shows the iPES-webpage with a simple example [15] opened with a Java-enabled webbrowser. The applet is embedded in the webpage and displayed as an orange rectangle which has to be clicked with the mouse to open a window for the simulator program. The example discussed on this webpage (single-phase thyristor rectification with RL-load) is opened automatically and a simulation can be performed immediately without any installations.

B. The User Experience

The online simulator is very simple to use. If Java 1.6 is preinstalled, the user has just to visit the according iPES-webpage. Java will be loaded automatically which takes a couple of seconds if the first time an applet is opened within a browser session. GeckoCIRCUITS is loaded automatically, and an orange rectangle is displayed on the webpage. The text inside the rectangle tells the user to click there with the mouse which will open GeckoCIRCUITS in a new window. A default

simulation model is loaded automatically from the server, and scope-windows for data display are opened, too. An example of this is shown in the screenshot in **Fig. 4**.

The user can immediately start a simulation in the menu "Simulation >> Init&Start". This is a "real" accurate simulation, not a visualisation of tendencies like in iPES. The user can now freely change all parameter values, can add and delete components, can build totally new models with different topologies, and can perform unlimited simulations. There are no limitations concerning number of nodes and/or elements.

The main limitation from the user's point of view is the inability to read from or write to the hard disk. Models can therefore not be saved. Simulation results cannot be written to the local hard disk either. The user has to perform screenshots for documentation of the simulation results. This is, however, sufficient for student exercises at the PES, ETH Zurich.

Generally, the GeckoCIRCUITS applet runs on the user's local CPU which defines the calculation speed (client-side computation, not server-side). Concerning the memory consumption there is a limitation of the GeckoCIRCUITS online version: The RAM available for Java applets is defined by the user's local settings. With typical default-settings, the Java Virtual Machine (that is the virtual "CPU" of the JRE) will get only 60 - 80MB RAM for the simulator even if there are, e.g., 2GB RAM memory totally available. As a result the user will often see a message window stating that due to memory restrictions, not every simulation step can be written to the scope (automatic data compression invoked). This limitation is only present in the online-version of GeckoCIRCUITS.

C. From the Instructor's Point of View

A simple implementation of a student course would be to create a set of HTML pages, one page for each exercise. In each page there would be GeckoCIRCUITS embedded with the according model or topology for the specific exercise already loaded as default. Optionally, different models can be loaded via the menu "File >> Open". For examples of such a course at PES, ETH Zurich, see e.g. [17] and [18].

The HTML page allows the addition of explanatory text describing the content of the exercise. There is no language limitation as more complex languages as e.g. Arabic, Japanese or Chinese can be easily written in HTML (see e.g. [19]).

There is also the possibility to use forms in HTML where the student can complete the answers and send them to the instructor's server (database). This is relatively easy to implement and very flexible. It theoretically allows the partial or complete automation of such time consuming tasks as tests and exams.

Another advantage is the very simple centralized administration of GeckoCIRCUITS in this online-implementation. A whole course consists of a set of

webpages designed by the lecturer or teacher, the GeckoCIRCUITS applet (which is a executable Java file), and the model files for various converter topology examples. The instructor/teacher fully controls the content of the course, the GeckoCIRCUITS license, and all model files employed in the student course.

All this can be easily assembled with very little knowledge of HTML, Java and Internet technology. The instructor/teacher has only to make sure that the student's computer has a standard web-browser and Java 1.6 preinstalled and a working Internet connection. If not enough computers are available in the class room, the students could also work from their homes with their own computers.
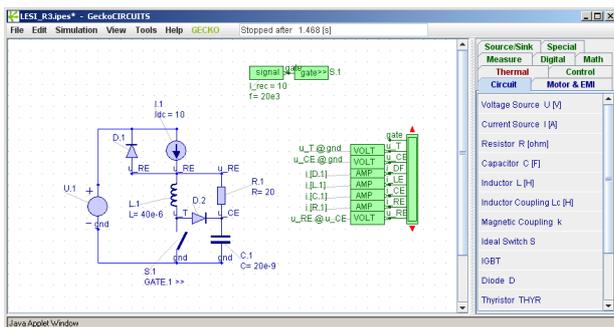
(a)
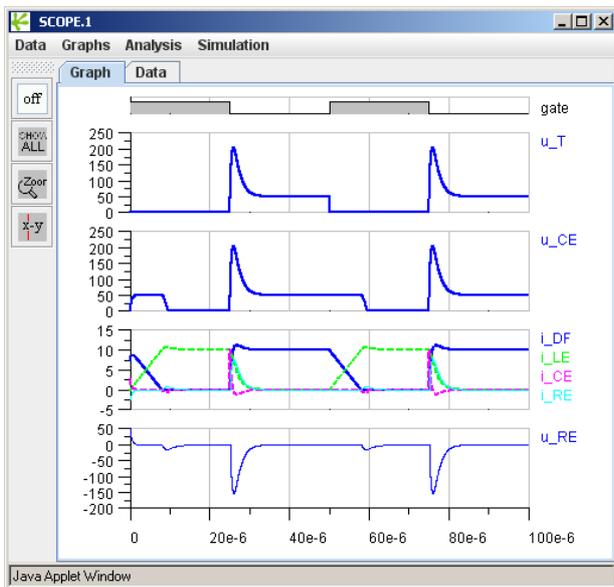
(b)

Fig. 5: Screenshot of an exercise example from the 7th semester master course in advanced power electronics at the PES, ETH Zurich. The example is about snubber networks for power switches, see [16].

IV. POWER ELECTRONICS COURSE AND STUDENT EXERCISES AT THE ETH ZURICH

A. GeckoCIRCUITS in Education

Originally, GeckoCIRCUITS was developed for the PhD research program at the Power Electronic Systems Laboratory (PES), ETH Zurich, due to various restrictions of commercial power electronic simulators

such as limited transient thermal simulation or missing EMI simulation and filter design options. There are currently (autumn 2009) 29 full-time PhDs and 5 senior researchers at the PES. From the educational point of view it has been very important to make GeckoCIRCUITS easy to use to reduce learning time and increase productivity.

In addition to GeckoCIRCUITS, which allows simulation of circuits and control including coupled transient temperature simulations, there is a 3D thermal solver (GeckoHEAT) and a 3D electromagnetic solver (GeckoEMC) based on the PEEC method under development. These additional simulation programs will permit coupled multi-domain simulations in power electronics with the goal of optimizing converter systems not only in respect of electric behaviour but also covering important issues such as thermal design (heat management) and EMI. GeckoHEAT and GeckoEMC will be available in spring 2010 and will be integrated into the education at PES.

Students, who do their master thesis in power electronics at the PES (more than 40 in 2008) are using GeckoCIRCUITS for circuit simulation.

Furthermore, GeckoCIRCUITS is used in the student exercises (master course - computer exercise). Each year there are about 120 - 150 students attending the power electronic courses (5th, 7th and 8th semester) of the ETH Zurich. **Fig. 5** shows a screenshot from an example of the 7th semester course (advanced power electronics) that is web-based as described in section III-C, see also [16]. There are 12 exercises (one per week). All exercises are given as PDF-file, and the simulation model describing the converter system is available with the online-version of GeckoCIRCUITS. Students have to solve the exercises in written form, and GeckoCIRCUITS is used to understand the various aspects of the problem and to validate results. According to initial student feedback, which has been very positive, the online simulation is extremely helpful to understand and solve the exercises.

For the instructors/teachers it was very easy to administrate the course in this way. Compared to previous exercises, there was no need to reserve computer rooms, test software on all computers, and check licenses. Changes in the content of the exercises can be performed quickly and with minimum effort.

Due to the positive user feedback and the relief in administration it is planned to extend usage of online simulation in power electronic student course within PES.

B. Limits and Restrictions of Online Simulation with GeckoCIRCUITS

The JAVA-block is a very flexible interface to the Java programming language in GeckoCIRCUITS. It allows for example to describe logical control structures like "if-then" statements in Java programming language (the syntax is close to C or MATLAB M-files, therefore very easy to learn and apply).
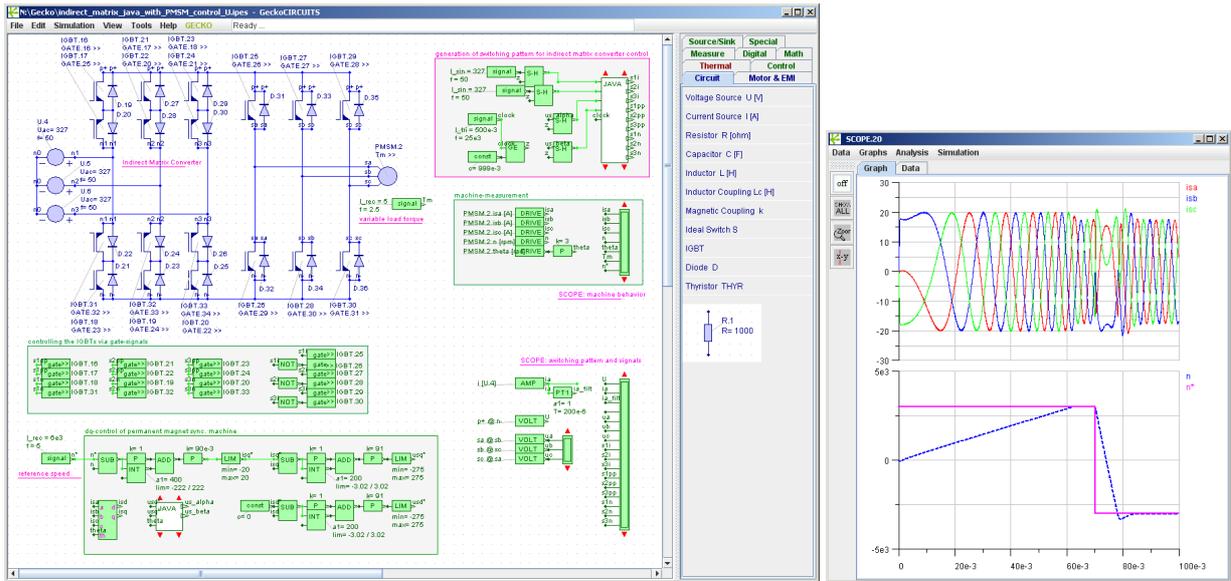
Fig. 6: (a) GeckoCIRCUITS-simulation of a three-phase indirect matrix converter [20], [21] for power supply of a PMSM. Most of the quite complex control logic for generation of the PWM pattern in the indirect matrix converter is done in a JAVA-block (red circle) which significantly reduces the model size. The remaining green control structure is the dq-control of the PMSM. (b) Simulation of the machine input currents isa, isb, isc and the speed (magenta curve: reference speed; blue curve: actual speed achieved by dq-control).

Due to its simplicity and efficiency, the JAVA-block quickly became highly popular among users. The JAVA-block is part of the control structure. The input signals are updated at each numerical time step. The input signals are processed according to the Java code in the block, and output signals are calculated and applied to the output pins of the block.

A key element of the JAVA-block is the compilation of the internal Java code in order to achieve acceptable simulation speed. The Java compiler is part of a library ('tools.jar' of the Java JDK [22]) which is not part of the typically preinstalled standard JRE. With 12MB this library is too large to be delivered together with the applet. Therefore, this very useful functionality is disabled in the online-version of GeckoCIRCUITS.

Another useful feature of GeckoCIRCUITS which is disabled in the online-version is the ability of interfacing to Simulink to realize parts of the control structure. This is also a popular feature of GeckoCIRCUITS because Simulink allows building very sophisticated control structures, but is technically restricted in case of an applet implementation.

Therefore, students in the PhD program and students working on their master thesis at the PES, will use the stand-alone version of GeckoCIRCUITS and not the online version.
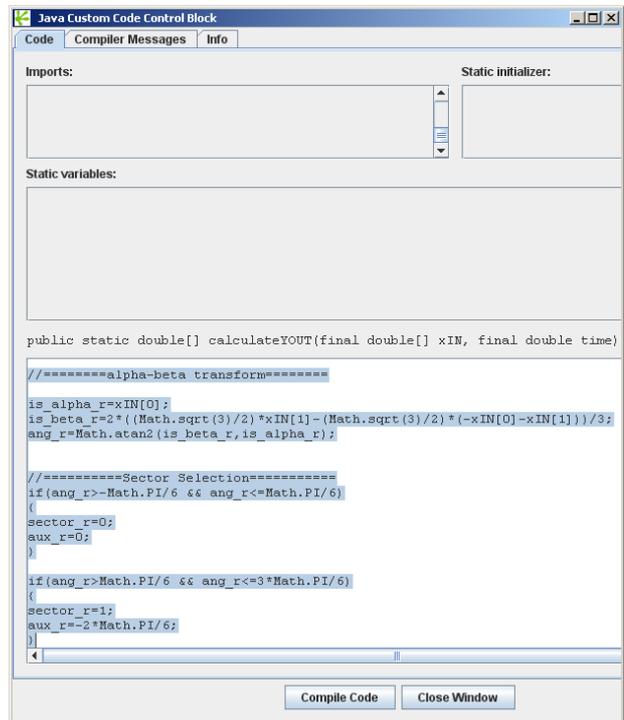
Fig. 7: Dialog window of the JAVA-block in GeckoCIRCUITS. Java code is written by the user (text marked in blue), and compiled before the start of the simulation. The JAVA-block is then accessed at each numerical time step of the simulation. In case of the indirect matrix converter it would add an extremely large number of control blocks to realize the PWM control without JAVA-block.

V. Conclusions

The Interactive Power Electronics Seminar (iPES) which was launched by PES, ETH Zurich, in 2001, provides Java Applets that are limited to a few variable parameters. This makes sense from an educational point of view, especially when teaching basics. iPES is very useful in demonstrating tendencies but does not perform numerically accurate simulations.

In order to allow unlimited simulation, the power electronics simulator GeckoCIRCUITS has been integrated into the iPES-platform as online-simulator. This novel online simulator is very easy to use, and optimized for power electronics. User acceptance is very high because no installations are needed.

The online simulator allows setting up power electronics courses which are very easy to use, and simple and highly efficient in maintenance and administration. Power electronic courses at the ETH Zurich are performed using this kind of online simulation.

References

[1] Drofenik, U., Kolar, J.W.: iPES - Interactive Power Electronics Seminar at www.ipes.ethz.ch.

[2] U. Drofenik, J. W. Kolar, "Survey of Modern Approaches of Education in Power Electronics", Proceedings of the 17th IEEE Applied Power Electronics Conference (APEC 2002), Dallas, Texas, USA, March 10-14, Vol. 2, pp. 749-755, 2002.

[3] U. Drofenik, J. W. Kolar, "Interactive Power Electronics Seminar (iPES) - A Web-Based Introductory Power Electronics Course Employing Java-Applets", Proceedings of the 33rd IEEE Power Electronics Specialists Conference, Cairns, Australia, June 23-27, Vol. 2, pp. 443-448, 2002.

[4] U. Drofenik, J. W. Kolar, "A Novel Interactive Power Electronics Seminar (iPES) Developed at the Swiss Federal Institute of Technology (ETH) Zurich", Journal of Power Electronics (The Korean Institute of Power Electronics), Vol.2, No.4, pp. 250 - 257, October 2002.

[5] GeckoCIRCUITS as free online circuit simulator available at www.gecko-research.com

[6] iPES-Applet "Flyback Converter - Magnetic Components" at www.ipes.ethz.ch/ipes/2010_ipec/flyback.html

[7] iPES-Applet "Magnetomotive Force (MMF), Inductance, Magnetic Flux" at www.ipes.ethz.ch/ipes/2010_ipec/magnetomotive.html

[8] Google Analytics at http://www.google.com/intl/de/analytics/

[9] Charles R. Sullivan, "3-D Circuit Animations" for power electronics at http://thayer.dartmouth.edu/3Dcircuits/animations/index.shtml

[10] Venkat Ramaswamy, "Interactive Power Electronics Online Text" at http://services.eng.uts.edu.au/~venkat/pe_html/peintro.htm

[11] J. Hamar, I. Nagy, H. Funato, S. Ogasawara, 0. Dranga and Y. Nishida, "Virtual Power Electronics: Novel Software Tools for Design, Modeling and Education", Proc. of the Power Conversion Conference (PCC 2007), Nagoya, Japan, April 2 - 5, pp. 502 - 509, 2007.

[12] S. Harb, K. Kalaldeh, A. Harb, I. Batarseh, I., "Interactive Java applets for power electronics E-learning", Proc. of the IEEE Workshop on Power Electronics Education, pp. 26 - 33, June 16 - 17, 2005.

[13] Y. Nishida, "iCASS" at http://www.sia.co.jp/~icass/index.html [in Japanese]

[14] F. A. S. Goncalves, C. A. Canesin, WWW COURSE IN POWER ELECTRONICS at www.dee.feis.unesp.br/gradua/elepot/ajuda/applets.html

[15] GeckoCIRCUITS online at http://www.gecko-research.com/applet-mode/geckocircuits_demo.html - press button 'Go to GeckoCIRCUITS Applet'

[16] Online simulation of snubber networks for power switches at http://www.gecko-research.com/pes_students/geckocircuits_applet_3.html

[17] Examples and links to online-simulation of the course "Power Electronic Systems I" (master course 7th semester at PES, ETH Zurich) at http://www.pes.ee.ethz.ch/en/our-range/education/courses/viewcourse/227-0247-00-g.html

[18] Web-based simulations of the course "Power Electronic Systems I" (master course 7th semester at PES, ETH Zurich) at http://www.gecko-research.com/pes_students/geckocircuits_applet_info.html

[19] Japanese version of iPES at http://www.ipes.ethz.ch/ipes/j_index.html

[20] J. W. Kolar, T. Friedli, F. Krismer, S. D. Round, "The Essence of Three-Phase AC/AC Converter Systems", Proceedings of the 13th Power Electronics and Motion Control Conference, 2008 (EPE-PEMC'08), Poznan, Poland, September 1 - 3, pp. 27 - 42, 2008.

[21] U. Drofenik, "AC/AC-Conversion for Highly Compact Drives - What Options Do I Have?", online report on http://www.gecko-research.com/reports/MatrixConverter_Part_1x.html

[22] JDK 6 Update 18, free download at http://java.sun.com/javase/downloads/widget/jdk6.jsp