

EDITOR OF THIS ISSUE

Prof. Kannan Moudgalya
IIT Bombay

The focus of this issue is on

Scilab

The next issue will be on Python

INSIDE

FOSSEE	01
MATLAB, Scilab & Octave	01
Why Scilab?	02
Spoken tutorials	02
Scilab textbook companion	03
Scilab & Xcos on Cloud	03
TBC code search	04
Scilab toolbox	04
Lab migration	04
Contact us	04

FOSSEE is funded by the National Mission on Education through ICT, MHRD, Govt. of India

FOSSEE is an acronym for Free and Open Source Software for Education. Initially, we were asked by our funding agency to focus on engineering education; so, there are two Es in the name. After some success, we were asked to include Science education also. Subsequently, we were asked to promote open source software in all fields of education. So, the current name of FOSSEE.

The main objective of the FOSSEE team is to help promote the use of open source software in educational institutions. The FOSSEE team creates useful instructional content that will help students easily learn open source software. It also helps academic institutions to migrate to open source software systems.

Two major open source software that FOSSEE promotes are Scilab and Python. It also promotes OpenFOAM, an open source alternative to commercial CFD software Fluent. It promotes DWSIM, an open source alternative to commercial chemical process simulators, such as Aspen Plus and Chemcad. It also promotes open source web development systems, such as Django and Drupal.

MATLAB, SCILAB & OCTAVE

Dr. Cleve Moler was a professor of numerical computer science, when he wrote the famous software Matlab in Fortran. He was one of the creators of the highly acclaimed numerical libraries, LINPACK and EISPACK. He wondered why it was not possible to create an environment that would allow matrix operations be carried out as easily as we do in Maths classes. He wanted to remove the requirement of most programming languages to do matrix operations: dimensioning and for loops. Out of this desire came Matlab, written in Fortran.

The Fortran source code of Matlab was available to the public and it ran in many different computers. The author of this article indeed used this version in the early eighties, during his Ph.D

The FOSSEE team has developed an electronic CAD tool eSim, that has many of the capabilities of its commercial counterparts, such as ORCAD. Open source hardware is another topic that the FOSSEE team members work on.

FOSSEE creates Spoken Tutorials for the software it supports, and makes them available to the public for self learning.

About 200,000 students have been trained in the use of Scilab and about 100,000 students have been trained on Python (<http://spoken-tutorial.org/statistics/training/>).

The FOSSEE team also answers questions of supported software on FOSSEE forum (<http://forums.fossee.in/>) & Spoken Tutorial Forum (<http://forums.spoken-tutorial.org/>).

Two of the most popular initiatives of FOSSEE are Textbook Companion and Lab Migration, which are explained in other articles of this newsletter, as applied to Scilab. It has started the flowsheet project of DWSIM, case study project of OpenFOAM, and the circuits project of eSim, all through crowdsourcing.

days. As Matlab was very convenient for mathematical operations, it became very popular.

A few commercial products came out of this Fortran Matlab: Matrix X and CTRL-C, to name a few. Prof. Cleve Moler also started a company to improve and to promote Matlab. This company is the famous Mathworks, which rewrote Matlab completely and made it commercial.

Out of the Fortran version of Matlab explained earlier, came Scilab. It was initially called Blaise. For various reasons, developers of this product decided to release it as an open source software. Scilab was developed at INRIA. Subsequently, Scilab was supported by Scilab Enterprises, which has now been bought over by the ESI group. Scilab continues as an open source software.

The FOSSEE team has been promoting Scilab for the following reasons:

- (1) It is an open source software
- (2) It has all the convenience that the original Fortran Matlab had.
- (3) It also uses the state of the art computational libraries LINPACK, EISPACK, etc.
- (4) It is used by the academia and the industry, an example of the latter being CNES, the French aerospace corporation. One of the products of CNES is the Ariane rocket, which is famous in the space

WHY SCILAB?

The author of this article asked the students of an Embedded Systems course in 2007 whether they had heard of Scilab. Only one of the 25 students had used Scilab, as he was the only one who had worked in industry before coming to IIT for higher studies. When this student wanted to use Matlab in his company, his boss told him that as it would cost Rs. 2 crore, they could not afford it, they being a startup. The company was working in the general area of embedded systems. The student in question found that the open source software Scilab could solve his problem and hence used it.

The above story is an example of the difficulties faced by the industry in general and small and medium companies in particular, in using expensive commercial software. It also demonstrates the need to train our students on open source software, such as Scilab. Students who are trained only on commercial software face the danger of not being useful to industry, as the commercial software they are trained in may not be available. The only accessible software, namely open source software, may not even be known

SPOKEN TUTORIALS

A Spoken Tutorial is an audio video tutorial created with the Screencast technology. A Spoken Tutorial is of 10 minute duration. Using a series of them, one can teach advanced topics also. The objective of the Spoken Tutorial activity is IT training and to improve the employment potential of our students. Spoken Tutorial project, a sister project of the FOSSEE project, is in charge of Spoken Tutorials.

A Spoken Tutorial is designed and created for self learning, and hence a beginner can learn without an instructor. The audio portion of Spoken Tutorials is dubbed into all 22 of our languages. There is an offline version of Spoken Tutorials, using which, many can simultaneously undergo a hands-on training programme, even if Internet were not available.

The Spoken Tutorial team has trained a total of about 35 lakh students during the past six years, on various IT topics. More than 22,000 lab courses all over India use Spoken Tutorials as a part of their curriculum, using the weekly mapping given by our teams. Scilab Spoken Tutorials have been created by the FOSSEE team and are available here: http://spoken-tutorial.org/tutorial-search/?search_foss=Scilab&search_language=English.

industry; many of our satellites have been placed in the orbit by Ariane rockets.

Scilab comes with a graphical programming interface Xcos, which is somewhat similar to Simulink, a product developed by Mathworks for use with Matlab. In contrast, Octave, the other open source alternative to Matlab, does not have an equivalent of Simulink. In any case, one should not argue whether one should use Scilab or Octave. Instead, we should see if these two together can provide a strong computing environment. The FOSSEE team has indeed completed this task. It has made Octave routines callable from Scilab. As a result, it is possible to provide the power of Octave within Scilab. This is the beauty of open source software systems: it is easy to make them talk to each other.

to students. If we have to be responsible to our society, we must train our students on open source software, such as Scilab.

As mentioned in another article, Scilab is based on numerically sound libraries, such as LINPACK, EISPACK, ODEPACK and DASSL. As a matter of fact, CNES, the French aerospace corporation that handles Ariane rockets relies exclusively on Scilab for all its calculations. If our satellites can be placed in their precise orbits through Ariane rockets launched with Scilab calculations, why can't the academics use Scilab for simple requirements, such as LU decomposition, calculation of eigenvalues and vectors, integrating ODEs and solving optimisation problems? Only in cases where we have an immediate need, say for example, a toolbox is required to publish a conference paper that is immediately due, should one even contemplate commercial software.

In this newsletter, we explain the various steps undertaken by the FOSSEE team to enhance the capabilities of Scilab and the ease of its use. We invite the reader to benefit by it. We also invite the reader to join in our efforts to partner us in our efforts to promote Scilab and other open source software.

A total of 24 Scilab Spoken Tutorials are available. As mentioned earlier, one can self-learn using these tutorials.

If while practising the tutorial, a learner has difficulty at any point, they can post their query in the Spoken Tutorial Forum, choosing the Minute and Second when this difficulty is experienced, see <http://forums.spoken-tutorial.org/>. Answers to such time synchronised queries can help other users: they can go through the previous discussion first. Queries in this forum are answered by the FOSSEE team members. Of course, any registered user can also answer the questions.

In case there are general queries on Scilab, but not related to Spoken Tutorials, one can post them on the FOSSEE Forum, see <http://forums.fossee.in/>. There is no timing information in these queries. These questions are also answered by the FOSSEE team members.

A total of about 200,000 people have been trained on the use of Scilab. Colleges that want to organise Scilab workshops in their colleges may contact a Spoken Tutorial Training Manager for their state, contact details of whom are available at http://process.spoken-tutorial.org/index.php/Software-Training#Contacts_For_Training.

SCILAB TEXTBOOK COMPANION

A major problem with open source software is the lack of documentation. Our teams at IIT Bombay have been addressing this problem. The first solution to this problem is Spoken Tutorial, which has now become a training programme. The other documentation approach is the Textbook Companion (TBC).

We have a large number of college going students in India. If only their energy can be harnessed in creating documentation, we can improve the accessibility of all open source software. Unfortunately, however, they do not like document writing, even if a student is good in coding. It also requires quite a bit of maturity to create documents. We addressed both issues by solving the inverse problem: ask students to write code for the solved problems of standard textbooks.

The TBC is thus a collection of code after code, for all solved problems of a standard textbook. As nothing from the book is reproduced, there is no violation of copyright. Because one has to see the underlying example in the textbook to understand any code, the TBC is truly a companion – it cannot replace the textbook. As a lot of care is taken to write textbooks, the documentation for the code, namely, details of the solution in the book, are excellent.

One may use a code from a TBC for the following:

1. One can execute the code and compare the results with the solution given in the textbook. This helps the user to quickly verify the calculations shown in the book.
2. One can change the parameters and do a what if study. This helps a student get a better understanding of the underlying concepts. This facility can be used effectively by an instructor in a classroom.
3. One can find out how to make a Scilab call for a function, such as fft, plot and solution to a linear system or ordinary differential equations. In this case,

the solved example of the book serves as the document for the Scilab code under discussion. One can see that the Scilab TBC acts indirectly as a document for the underlying function, with the explanation given through the worked out example in the book. It is well known that explaining through examples is an effective method of instruction.

Anyone who is good in a subject and willing to learn Scilab can contribute to this project. They first submit the code required for the examples of a particular chapter. If the code is ok, we allow them to complete the book, within a time frame. If the completed code works without any bugs the first time, the contributor gets 20% more honorarium. AICTE books get 20% more honorarium. We pay up to Rs. 15,000 as an honorarium for their work, depending on the number of examples they have coded, their difficulty level, etc.

The FOSSEE team has coordinated the development of 500+ TBCs, with funding from the National Mission on Education through ICT, MHRD. These TBCs have been created by students and teachers of various colleges in India and working professionals.

TBC contributions are checked for execution errors first by an automated script. They are also checked by domain experts for correctness of results. It is easy to do this check, as solution for every code is available in the companion textbook. Our approach has allowed the college students of India to create documentation to an important open source software, Scilab. The 500+ Scilab TBCs have a total of about 65,000 Scilab scripts. Scilab TBCs can be downloaded from

http://scilab.in/Completed_Books for offline use. They are also made available through the GARUDA Scilab Cloud and it is explained in another article in this newsletter. Scilab TBC with 65,000 scripts can be used to understand the calling procedure of Scilab functions – this is explained with an example in yet another article in this newsletter. We extend the TBC approach to all open source software that we support through FOSSEE: Python, eSim, OpenFOAM and DWSIM.

SCILAB & XCOS ON CLOUD

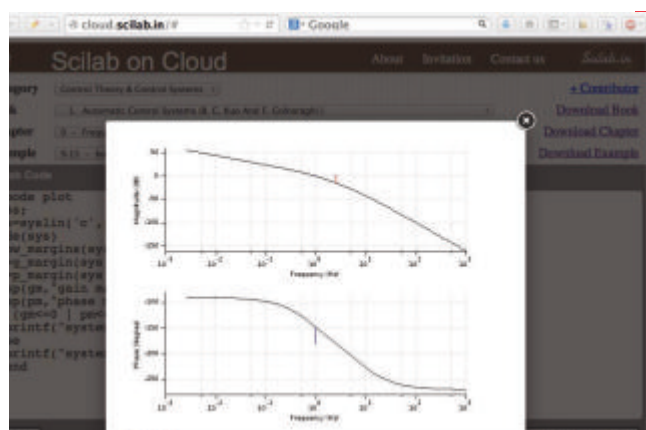
The FOSSEE team has created a Scilab on Cloud facility, using the GARUDA cloud of CDAC. This allows any smart device with Internet connection to try out Scilab calculations on the cloud.

We have also uploaded all the Scilab TBC on this cloud. A user can choose a category of books, then a book in this category, a specific chapter and finally an example. One can then execute this code as it is, or after making changes to it. Results appear on the right hand side of the screen. If there are any plots, one may download them. The screenshot of a sample code on the cloud, with its output, is shown in the diagram on the right.

As mentioned in another example, all Scilab scripts of 65,000 examples from textbooks are available through the cloud. This facility is available on the go, as one can use Internet enabled mobile phones also. As explained in another article, one can also use this

facility to execute the code that comes out of the TBC Code Search.

We are currently in the process of creating an Xcos on Cloud. Using this feature, one will be able to construct an Xcos block diagram on a web browser, validate and execute it. One will also be able to execute previously created Xcos code.



TBC CODE SEARCH

The Scilab TBC collection that has about 65,000 scripts are an excellent resource for search and use. We will now explain this with an example.

The author of this article wanted to set a question on partial fraction expansion in his control course. On doing a Google search for the phrase Scilab partial fraction expansion, he found that the command to use was `pfss`. Then, he entered `pfss` in the search box of http://scilab.in/tbc_solr_search and searched for the Scilab code that contains this function name. There were 140 hits from amongst the Scilab TBC: On scrolling down this list, a familiar book that the

author used when he was a student was located – in this case, 27th in the list of 140. On selecting the option to view this example, the author was allowed to execute the code on the cloud, explained in another article in this newsletter. On executing this code on the cloud, one gets the partial fraction expansion.

This is an extremely valuable resource for beginners. Those who are conversant with Scilab also may not remember the syntax to Scilab function calls. As this is a focussed search, the relevance of results is very high. Finally, as example based learning is possibly the most effective method of learning for beginners and occasional users, the approach illustrated here is very effective.

SCILAB TOOLBOX

One of the shortcomings of Scilab is that the development of toolboxes in Scilab is at the nascent level. Thus, one finds it difficult to use it for research purposes, although Scilab is good enough for most classroom teaching. To address this difficulty, the FOSSEE team has undertaken to develop the following toolboxes: Optimization, Signal processing/DSP, Image processing, Computer vision, Communication systems, System identification, Scilab to C, and Control system design.

We attempt to achieve the above by relying on the state of the art open source software libraries, written mostly in C/C++. Scilab calls are translated into C/C++ calls, and the results returned in Scilab format – the Scilab user does not even know how the computations are carried out. The various open source software libraries we have used are Octave, IT++, OpenCV, Tesseract, PCL & COIN-OR.

Currently, we have released the first version of FOSSEE Optimization toolbox and it has garnered more than 1500 downloads. We are close to releasing the second version of this toolbox. The second version of the toolbox is an improvement over the existing commercially available Optimization toolbox as the FOSSEE toolbox includes CBC, BONMIN and ECOS libraries. The toolbox is available for download here: <https://atoms.scilab.org/toolboxes/FOT>.

There are times when the required missing functions are not readily available. These call for writing these functions in Scilab from scratch. We invite domain experts who are willing to create these functions to partner us. Other Scilab toolboxes such as signal processing, image processing and computer vision toolboxes are close to completion and will be released soon. The FOSSEE toolbox development effort has been summarized in the paper 'Developing Scilab Toolboxes, A Multi-FOSS Approach'. This paper was presented at the IEEE Indian Control Conference, IIT Guwahati, 4-6 January, 2017.

SCILAB FOR HARDWARE

Apart from computation, modelling and simulation, Scilab finds application in data acquisition and control. We have been using Scilab for about a decade for a homegrown embedded control setup namely single board heater system (SBHS). SBHS is a temperature control setup that consists of a heating element, blade, computer fan, temperature sensor, microcontroller and associated circuitry. The time constant of this system is less than a minute. We use Scilab and Xcos to read temperature values from the temperature sensor and actuate heating and cooling elements in order to maintain the blade temperature at a desired value. We also use Scilab to model this plant and implement several control strategies. More than 500 students have been benefited through this effort. For more information, please visit, <http://sbhs.fossee.in/>.

We enabled Scilab and Xcos to interact with an Arduino Uno board and a customised shield consisting of a few sensors and actuators. We performed experiments to access Light Emitting Diode, Push button, Light Dependent Resistor, DC motor, Potentiometer, Thermistor and Servo motor. We are writing a book that describes a step by step method to perform these experiments using Scilab, Xcos, Python, and Julia. Finally, the ongoing Scilab2C toolbox effort is expected to be useful to the embedded systems enthusiasts.

LAB MIGRATION

As mentioned in another article in this newsletter, many college students contributed to the Textbook Companion. Although their colleges were proud of this work, they could not shift to Scilab, so long as it was not used in their labs. The FOSSEE team has introduced Lab Migration to help them switch their labs to Scilab. We give an honorarium and certificates to those who migrate their labs and use them for a minimum of two years. A total of 59 labs have been migrated to Scilab. More details on Lab Migration is available here: http://scilab.in/Lab_Migration_Project.

CONTACT US

We would like to receive your feedback.
Please mail us at – contact@scilab.in
Visit our website – <http://scilab.in>