

IITB Summer Internship 2014



Project Report

SCILAB-OCTAVE INTERFACE

Principal Investigator

Prof. D.B. Phatak

Project In-Charge

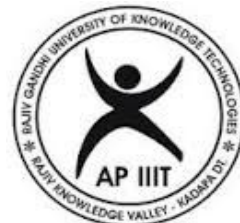
Mr. Nagesh Karmali

Project Mentor

Ms. Shamika Mohanan

Project Team Member

Mr. B. Babu



June 30, 2014

**Summer Internship 2014
Project Approval Certificate**

**Department of Computer Science and Engineering
Indian Institute of Technology Bombay**

The project entitled “**Scilab-Octave interface**” submitted by **Mr. B.Babu** is approved for Summer Internship 2014 programme from **26th May 2014** to **6th July 2014**, at Department of Computer Science and Engineering, IIT Bombay.

Prof. Deepak B. Phatak
Dept of CSE, IITB
Principal Investigator

Mr. Nagesh Kamali
Dept of CSE, IITB
Project In-Charge

Place: IIT Bombay, Mumbai
Date: June 30, 2014

Declaration

I declare that this written submission represents my ideas in my own words and where others' ideas or words have been included, I have adequately cited and referenced the original sources. I also declare that I have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in my submission. I understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

Mr. B.Babu
RGUKT-R.K.Valley,
Andhra Pradesh.

Date : _____

ACKNOWLEDGMENT

I, myself a summer intern of Spoken Tutorial and Oscad team, would like to thank Prof D.B.Phatak for giving me an opportunity to work on such a noble project. I want to express my gratitude towards Prof. Kannan M Moudgalya for his valuable guidance and support. My sincere thanks to my mentor Ms. Shamika Mohanan for her constant motivation and support to put up my ideas into reality. I am overwhelmed to acknowledge her dedication and humbleness. Finally I would like to thank all IIT Staff, my colleagues for helping me at critical junctures and made things possible.

Abstract

The Scilab-Octave interface is used to execute Octave commands on Scilab console. User can execute all Octave basic commands, 3rd party Octave packages and graphs using this interface. For this we need complete installation of Octave. User can extend his/her Scilab console capabilities by linking with Octave using Scilab-Octave interface.

It is very useful for newly users who had migrated to Scilab from Octave. We don't need new setup for this interface.

Scilab-Octave interface is Open Source tool. It is easy to operate and maintain.

Table of Contents

1.	INTRODUCTION	
1.1	Documentation Purpose	1
1.2	Project Scope	1
1.3	Intended Audience and Document Overview.....	1
1.4	Document Conventions	2
1.5	References	2
2.	SPECIFIC REQUIREMENTS	
2.1	External user Requirements	3
2.1.1	User Interface	3
2.1.2	Hardware Requirements	3
2.1.3	Software requirements	3
2.2	Functional Requirement	3
2.2.1	End user	3
2.3	Non Functional Requirements	3
2.3.1	Performance Requirement	3
2.3.1.1	Performance	3
2.3.1.2	Safety	4
3.	INTERFACE STRUCTURE	
3.1	Approach	4
3.2	Interface Structure	5
3.3	User interaction diagram	6
3.4	Dataflow diagram.....	6
4.	USER MANUAL	
4.1	Initialization	7
4.2	Using sci2oct function	7
4.3	Some guidelines	9

List of figures :

Fig 1 : Structure of sci2oct tool

Fig 2 : User interaction diagram

Fig 3 : Dataflow diagram

Fig 4 : Initialization of sci2oct tool

Fig 5 : Scilab-Octave interface example 1

Fig 6 : Scilab-Octave interface example 2

1 INTRODUCTION

Scilab and **Octave** both are open source, cross-platform numerical computational packages, primarily intended for numerical computations. We have pros and cons in both Scilab and Octave. For example Octave don't have good GUI, and Scilab is not compatible with Matlab. User can get a good platform for computations if both available in combination. This Scilab-Octave interface provides a good platform for user to execute Scilab and Octave commands together in Scilab console.

1.1 Documentation Purpose

The purpose of this document is to present a detailed description of Scilab-Octave Interface. It will explain the purpose and features of the system, the interfaces of the system, what the system will do, the constraints under which it must operate and how the system will react to external stimuli. This document is intended for both the stakeholders and the developers of the system .The features discussed in this document are mainly functional requirements. It also covers the non-functional requirements like performance, availability etc.

1.2 Project Scope

Scilab is free and open source software for numerical computation providing a powerful computing environment for engineering and scientific applications. Linking Octave with Scilab we can improve the Scilab capabilities. Everything is operates on Scilab console itself so user can easily use this tool.

1.3 Intended Audience and Document Overview

This document will be the first guide for prospective software developers, project managers, testers and most importantly users of Scilab and Octave. This document contains software functionality, software and hardware requirements and user documentation.

Developer: The developer who wants to read, change, modify or add new requirements into the existing program may need first to consult this document and update the requirements in appropriate manner so as not to change the actual purpose of the system or make the system inconsistent.

User: The user can get instructions about usage of this tool from this document.

Tester: The tester needs this document to validate that the initial requirements of this programs actually corresponds to correctly executable program.

1.4 Document Conventions

In general this document prioritizes in writing the requirements of the system and analyzing in details the tools being provided to its users. Every requirement is having its own priority (none conflicting).

1.5 References

- | <https://www.scilab.org/>
- | <http://www.stackoverflow.com>
- | <http://www.gnu.org/software/octave/>
- | Software requirements specification content and format standard

2 Specific Requirements

2.1 External user Requirements

2.1.1 User Interface:

The interface should be user friendly, so that user can access his requirements easily. Scilab interface should be used.

2.1.2 Hardware Requirements :

Device should satisfy latest version of openGL requirements.

2.1.3 Software requirements :

Scilab, Octave, gcc latest versions.

2.2 Functional Requirement

2.2.1 End user:

User gives Octave command in Scilab and expects corresponding Octave output in Scilab. User expects all types of outputs.

2.3 Non Functional Requirements

2.3.1 Performance Requirement

2.3.1.1 Performance :

The system must be interactive and the delays involved must be less. So in every action-response of the system, there are no immediate delays. In case of opening windows of popping error messages and saving the settings or sessions there is delay much below 3 seconds.

2.3.1.2. Safety:

Interface should not affect other programs. The main security concern is for users is filesystem changes. This interface uses temporary files so it should not affect entire filesystem.

3 Interface Structure

3.1 Approach :

To use any computation tool, first of all we need to give input. Here we are using Octave in Scilab so we have to take Octave command in Scilab and pass it to Octave through our own techniques. In this Scilab-Octave interface we are passing input to Octave through one temporary file(text) from Scilab. After that Octave gives corresponding outputs(except graphs) to another temporary file. It(Octave) will displays graphs directly. Finally Scilab reads output file and converts output to corresponding types and displays on Scilab console.

It executes one command at a time, so we will get problem in executing 3rd party packages. For that this interface uses another file(pkgs) to store recently loaded packages and includes them in input file(octFile) every time.

3.2 Interface Structure :

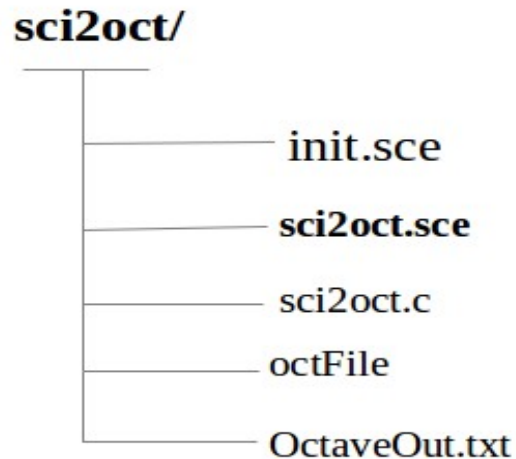


Fig 1:Structure of sci2oct tool

init.sce :

It is an initialization program. It initializes whole set up by linking C routines and Scilab functions to Scilab.

sci2oct.sce :

It is the main program in this interface. It contains **sci2oct(-,-)** and other functions to handle Octave inputs and outputs. First it takes Octave command and arguments and writes in **octFile**. While writing it checks arguments data types and adds corresponding symbols('{','}',',','\n'), and “”) at the beginning and ending.

After reading output from file it converts to corresponding type(matrix, sparse matrix or string matrix) and displays on Scilab console.

Its takes care loading already loaded packages and etc.

sci2oct.c :

This C program is called by **sci2oct** function from **sci2oct.sce** program. It activates Octave to take input. And it removes unwanted letters(ex : **ans = etc**) in output file.

octFile :

It stores Octave command temporarily.

OctaveOutput.txt :

This OctaveOutput.txt file used to store Octave output temporarily.

3.3 User interaction diagram :

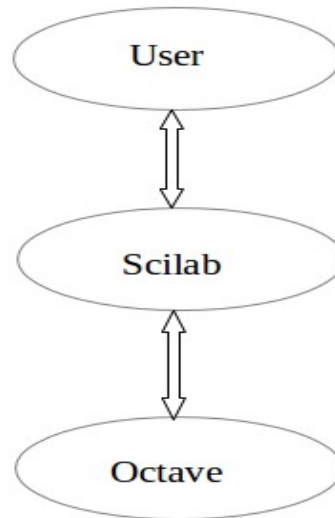


Fig 2 : User interaction diagram

3.4Data flow diagram

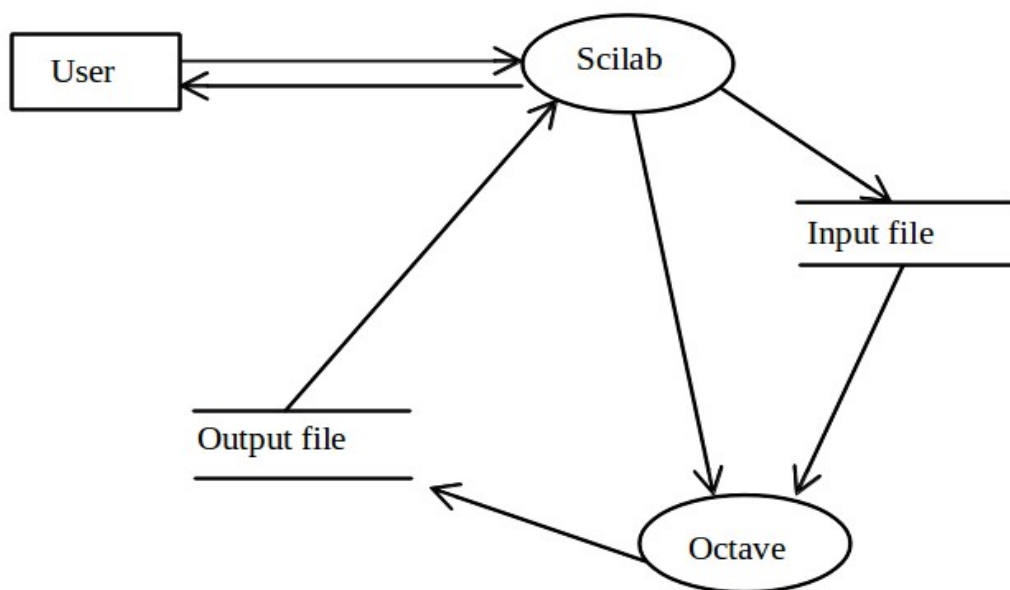
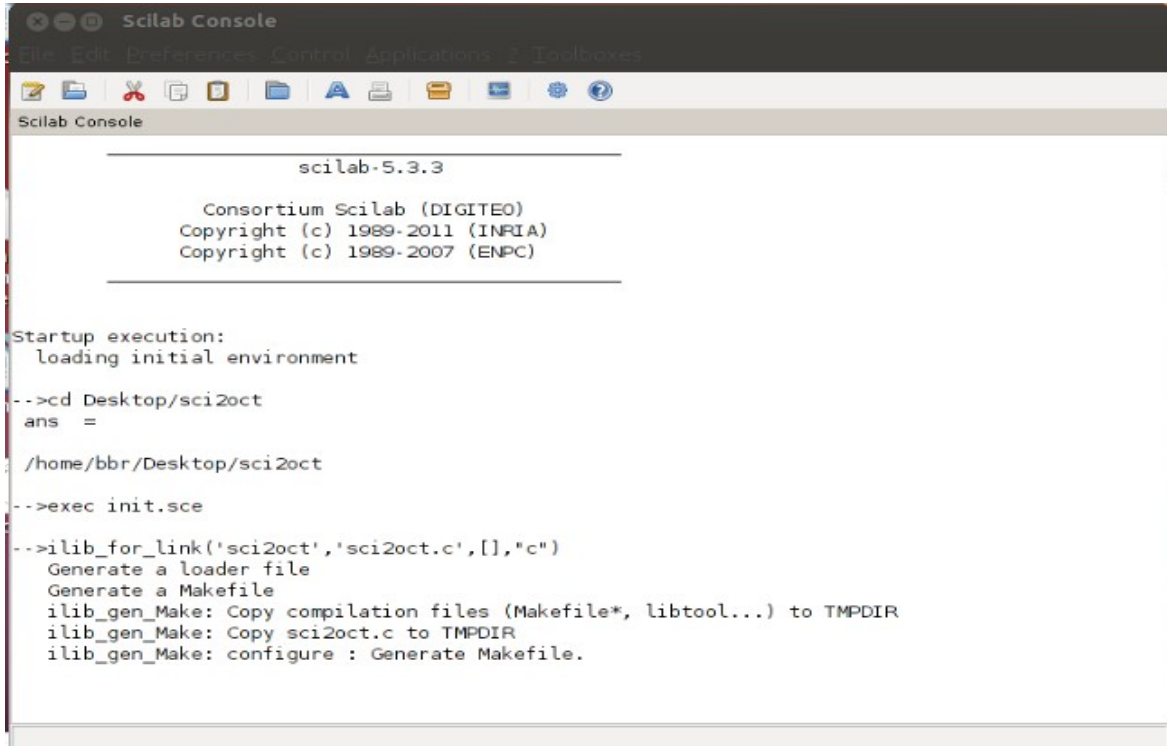


Fig 3 : Dataflow diagram

4 User manual :

4.1 Initialization :

Make sci2oct directory as present working directory in Scilab console. And execute init.sce(exec init.sce) in Scilab.



```
Scilab Console
File Edit Preferences Control Applications & Toolboxes

Scilab Console

scilab-5.3.3

Consortium Scilab (DIGITEO)
Copyright (c) 1989-2011 (INRIA)
Copyright (c) 1989-2007 (ENPC)

Startup execution:
loading initial environment

-->cd Desktop/sci2oct
ans =

/home/bbr/Desktop/sci2oct

-->exec init.sce

-->ilib_for_link('sci2oct','sci2oct.c',[],"c")
Generate a loader file
Generate a Makefile
ilib_gen_Make: Copy compilation files (Makefile*, libtool...) to TMPDIR
ilib_gen_Make: Copy sci2oct.c to TMPDIR
ilib_gen_Make: configure : Generate Makefile.
```

Fig 4 : Initialization of sci2oct tool

4.2 Using sci2oct function

Arguments : Octave command, arguments(optional)

Return value : Output of Octave command or error message

Description : This is the main function in Scilab-Octave interface. It takes Octave commands as valid Scilab string and its(Octave command) arguments as arguments. Then it writes that Octave commands in octFile. Octave puts it's output in OctaveOutput.txt. Scilab reads output from OctaveOutput.txt and displays on it's own console.

Example : 1 Here matrix1 is complex matrix. “ishermitian” is an Octave command to check given matrix is Hermitian matrix or not. We can give any command in two ways. But second one is most preferable(Reason is explained in coming pages).

```
-->matrix1=[2,2+%i,4;2-%i,3,%i;4,-%i,1];  
  
-->sci2oct('ishermitian(matrix1)') ← First way  
ans =  
  
    1.  
  
-->sci2oct('ishermitian',matrix1) ← Second Way  
ans =  
  
    1.  
  
-->
```

Fig 5 : Scilab-Octave interface example 1

Example 2: Using Packages(Here control system toolbox is used)

```
-->sci2oct('pkg load control');  
  
-->sci2oct('tf','s');  
  
-->ans  
ans =  
  
Transfer function 'ans' from input 'u1' to output ...  
  
yl: s  
  
Continuous-time model.
```

Fig 6 : Example 2(Using other packages)

4.3 Some guidelines :

- Install latest versions of Scilab and Octave.
- User should initialize this interface for every new Scilab console.
- In this interface we can pass Octave commands in two ways. First one is as entire expression(ex : `sci2oct('plot(x,y)')`). And Second one is passing function name and arguments differently(ex : `sci2oct('plot',x,y)`). Second one most entertained because first one won't work for some functions(which are in scilab with same name).
- User should have read and write permissions(file) in sci2oct directory.