Google Summer of Code 2020

Title

QuTiP Project: Improve Quantum Circuits Efficiency and Portability

Ways to contact me

Name: Mateo Laguna Guantiva

* Postal Address: Carrera 26A #1D-55 Apt 402, Bogotá D.C., Colombia.
* Phone number/WhatsApp: (+57) 312 446 8728
* Institutional email: [m.laguna10@uniandes.edu.co](mailto:m.laguna10@uniandes.edu.co)
* Personal email: [mateolaguna2552@gmail.com](mailto:mateolaguna2552@gmail.com)
* Skype: [mateolajuna@hotmail.com](mailto:mateolajuna@hotmail.com)
* GitHub: <https://github.com/mlaguna10>
* LinkedIn: <https://www.linkedin.com/in/mateo-laguna-guantiva-67803517b/>
* Twitter: <https://twitter.com/mateolaguna9>
* Hangouts: [mateolaguna2552@gmail.com](mailto:mateolaguna2552@gmail.com)
* Microsoft Teams: [m.laguna10@uniandes.edu.co](mailto:m.laguna10@uniandes.edu.co)

Project

Which one is it?

Why it would be finished?

Objective

Overview

Quantum Computing has widely become an outstanding field catching the attention of many scientists. QuTiP understood some time ago the relevance of this emerging field and the needs of its community, that is why although QuTiP is well-known by its simulations of the dynamics of open quantum systems, it also developed a Quantum Information Processing (QIP) module.   
  
The high demands of QuTiP users for new features in the QIP module led to the GSoC 2019 project. This year, the GSoC 2020 project would allow the QIP module to achieve one of its greatest developments by integrating to its code the quantum libraries Cirq (Google) and Qiskit (IBM), two of the most developed companies around the world in this field that lead the quantum computing race. The project would also enhance quantum circuits in terms of speed to compete with other available solutions. Finally, the project would build a way to make possible communication between different implementations of quantum instructions, and it would further develop the GSoC 2019 code.

Expected Outcome

#1 : how to ? how long each one?

…

#6

Deliverables

1. *Faster quantum circuits:*

## A method under the qutip.QuantumCircuit module that makes the time measure possible.

## An IPython Notebook in GitHub by coding using libraries of other solutions to measure times for different scenarios against the QIP module.

## Functions to enhance the code in QuTiP taking actions\* to gradually reach the measures of other available solutions for quantum circuits. (*\*Actions like running quantum circuits in the background with C instead of Python or parallel computing)*

## An IPython Notebook in GitHub reporting the new speed capabilities of the QIP module.

1. *Import/export of quantum circuits to a standard format* *and other libraries:*

## Functions to export objects of the qutip.QuantumCircuit module in different formats such as JSON, HTML, JavaScript or XML to make them compatible with other implementations (e.g. Cirq, Qiskit, Yao, etc.)

## Functions written in Quantum Assembly Language (QASM) to create an intermediate representation for quantum instructions.

## Function to export/import circuits from Qiskit (IBM quantum circuit library) using its API. (It’s important to notice that IBM’s API is an API REST)

## Function to export/import circuits from Cirq (Google quantum circuit library) using its API. (It’s important to notice that Google’s API is a traditional API)

## An IPython Notebook in GitHub with the documentation of each of the above code products.

1. *Further development and extension of QuTiP:*

## A function to include stochastic dynamics to the quantum information noise simulation module developed in GSoC 2019.

## Methods to extend the quantum information processing module and the quantum circuit capabilities.

## An IPython Notebook in GitHub with the documentation of each of the above code products.

Background + emphasis on the skills of the project

Interest and motivation: this focus my career in QC.

I have completed my bachelor’s degree in Physics and I’m currently pursuing my second undergraduate degree in Computing and Systems Engineering. Due to these two fields of study, my background is related to computational physics which provides me with the knowledge to understand concepts as quantum circuits formalism, quantum information processing, and quantum computing; and which also provides me with the skills to work in the back-end and the front-end of a project, to code scientific-python scripts and to work in open-source projects through GitHub.

I have scientific-programming skills in python, java and, C. I handle libraries such as Scikit-Learn, Tensorflow and, Keras for Machine Learning and Artificial Intelligence. For data treatment, I use Numpy, Pandas, and Scipy. In data visualization, Matplotlib and Seaborn. Regarding the engineering-programming skills, I know to develop software in the back-end (Maven, Jenkins, GlassFish, Postman) and front-end (Bootstrap, Angular, TypeScript) using Java, CSS, HTML.

I joined the quantum optics research group at my college because I like applied quantum mechanics, and I started my double degree because I like computation and algorithms, but I was still missing the common topic between those fields. Then, I found quantum computing, and in a fraction of time, I knew it would be what I want to do for the rest of my life. I’d like to join this specific project because it’s my passion, I enjoy each of the tasks related to this topic. I also want to join this project because it would let me interact with people that are already doing what I want to do: their experience is priceless.

Section: projects which I developed

1. *Computational Search of The Lowest Bound of Phase Information for Plausible Digital Image Reconstruction*. I implemented artificial neural networks (using TensorFlow) to study the physics behind a digital image. The main purpose of the project was to know where was the critical information of an image for a reconstruction process.

GitHub repository: <https://github.com/mlaguna10/Codigo_neuronas>

1. *Natural Language Recognition to enhance the processes of medical services in Colombia.* I developed a code (using Tesseract) to recognize words relevant to medical services such as private and public hospitals, clinics, etc. This process was decisive for a patient when authorization for an immediate medical process was needed.

GitHub repository: <https://github.com/mlaguna10/Trabajo>

1. *Data Science applied to the industry: insights from purchases.* Given a set of data I was asked to find some useful insights for the company about the purchases of a product.

GitHub repository: <https://github.com/mlaguna10/rappi_test>

Time availability and duties

My classes end on 23rd May, and the next academic semester I won't have a heavy burden since I’ll take just two lectures. Given that academic activities would not interfere with the GSoC project, having the time availability, and a week-by-week program would allow me to accomplish each of the goals of this project. Moreover, I can keep working after the GSoC project as an open-source contributor to QuTiP.

Calendar

**Before the official coding time:**

* *Communication with the mentors of the project*: this task was completed on the 15th march with an email to Nathan Shammah, Eric Giguère and Alex Pitchford. Since that day we’ve shared emails and comments on GitHub until nowadays.
* *Communication with the community:* this task was completed on the 15th and 23rd march. On the 15th, I wrote an email to Boxi Li (graduate GSoC 2019) who kindly gave me helpful suggestions and provide me with his proposal as a guide. On the 23rd, I started helping a user in the QuTiP Help group.
* *Open a pull request in QuTiP:* this task was completed on the 20th of March. The first pull request [#1209](https://github.com/qutip/qutip/pull/1209) was made to solve issue #1208 asking to add more single-qubit gates to the QubitCircuit module. The second pull request [#106](https://github.com/qutip/qutip-notebooks/pull/106) was the documentation testing the new features added under issue #1208.
* *Learning the QuTiP structure of the code:* this task was made to make the pull request. However, this task would eventually come up again along with the entire project.
* *Setting up the tools to code and to manage the project:* download the QuTiP library and setting up a conda environment to be able to develop code as a contributor were tasks of this previous-stage of the project. This preliminary part of the project was managed using the tool TeamWork which allowed me to create tasks and milestones with defined dates.
* *Study of other solutions and set up of environments:* since this project would be related to other solutions such as Cirq and Qiskit, I’d need to understand better and practice more with those tools, so I won’t waste time during the project in other solutions but coding in mine.

1st June – 30th June

Milestone: faster quantum circuits

Week 1 and 2

* + 1. Design the scenarios that would be tested among the other quantum circuit solutions.
    2. Code the method that would measure the time of each previous quantum circuit scenario.
    3. Run the scenarios in QuTiP and measure times.

1.2.1. Set up environments to use other solutions.

1.2.2. Create the IPython Notebook importing the previous environments.

1.2.3. Test scenarios under the libraries of other solutions and measure those times.

Week 2 and 3

1.3.1. Set up the levels that QuTiP must overcome to be faster than other solutions.

1.3.2. Code calculations using C language.

1.3.3. Code calculations using parallel computing.

1.3.4. Seek for more alternatives to improve efficiency of the code.

1.4.1. Create the notebook with the report and the documentation of the work done in the month.

Important: there must be a reviewed commit by the end of this week.

Week 4 (First evaluation)

Fix bugs and solve setbacks to complete the milestone of the first month.

Note: Study QASM.

1st July – 31st July

Milestone: import/export of quantum circuits to standard format and to other libraries

Week 1 and 2

2.1.1. See the exportation/importation format of other solutions.

2.1.2. Code functions to export/import objects from QuTiP to other solutions.

2.2.1. Code functions in QASM.

2.3.1. Make requests to the IBM’s API.

2.3.2. Code function to export/import circuits from Qiskit using its API.

Week 2 and 3

2.4.1. Make requests to the IBM’s API (since it’s a non-REST API this would take more time)

2.4.2. Code function to export/import circuits from Cirq using its API.

2.5.1. Create IPython Notebook with the documentation of the code products.

Important: there must be a reviewed commit by the end of this week.

Week 4 (Second evaluation)

Fix bugs and solve setbacks to complete the milestone of the second month.

Note: Study GSoC 2019 deeper.

1st August – 31st August

Milestone: further development and extension of QuTiP – code submission

Week 1 and 2

3.1.1. Code a function to extend the GSoC 2019 project to stochastic dynamics.

3.2.1. Create a document with the new functions and methods added to the QIP module and the QuantumCircuit class along the past two months and half.

3.3.1. Create IPython Notebook with the documentation of the extension of GSoC 2019 and add the previous document.

Week 3 and 4 (Code submission)

Fix bugs and solve setbacks to complete the milestone of the third month.

Important: Start to prepare from week three the submission of the GSoC 2020 code (at least one commit must be in the development branch).

