Advances in Artificial Intelligence and Machine Learning; Research 3 (4) 1685-1698 Received 17-10-2023;; Accepted 26-11-2023;; Published 03-12-2023;

Benchmarking Differential Evolution on a Quantum Simulator

Parthasarathy Srinivasan parthasarathy.s.srinivasan@oracle.com
Oracle Corporation
USA

Corresponding Author: Parthasarathy Srinivasan

Abstract

The use of Evolutionary Algorithms (EA) for solving Mathematical/Computational Optimization Problems is inspired by the biological processes of Evolution. Few of the primitives involved in the Evolutionary process/paradigm are selection of 'Fit' individuals (from a population sample) for retention, cloning, mutation, discarding, breeding, crossover etc. In the Evolutionary Algorithm abstraction, the individuals are deemed to be solution candidates to an Optimization problem and additional solution(/sets) are built by applying analogies to the above primitives (cloning, mutation etc.) by means of evaluating a 'Fitness' function/criterion. One such algorithm is **Differential Evolution(DE)** which can be used to compute the minima of functions such as the rastrigin function and rosenbrock function. This work is an attempt to study the result of applying the **DE** method on these functions with candidate individuals generated on classical Turing modeled computation and comparing the same with those on state of the art Quantum computation.The study benchmarks the convergence of these functions by varying the parameters initialized and reports timing, convergence, and resource utilization results.

Keywords: Evolutionary algorithm, Differential evolution, Quantum computer, Qubit, Fitness function.

1. INTRODUCTION

Differential Evolution is a Genetic Algorithm technique to solve Combinatorial Optimization problems, by mutating random candidate solutions (population) and evolving the solution pool over several generations. The fitness functions considered in this work are the rastrigin and rosenbrock functions which are mathematically represented respectively as follows:

$$f\,(x) = \sum_{i=1}^{i=dim} ({x_i}^2 - 10 * cos\,(2 * \pi * x_i)\,) + 10 * n,$$

where dim ↔ number of dimensions in input
n ↔ input size

Citation: Parthasarathy Srinivasan. Benchmarking Differential Evolution on a Quantum Simulator. Advances in Artificial Intelligence and Machine Learning; Research. 2023;3(4):96.

$$f\,(x) = \sum_{i=1}^{i=n-1} \left(\left(100*(x_{i+1} - x_i^2)^2\right) + (x_{i-1})^2\right), -30 <= x_i <= 30,$$

where n ↔ input size

In this work the best solution to the above function is found by evolving a pool of random candidate solutions, by applying mutation (picking 3 random candidate solutions viz. a,b,c and computing the mutant with the following computation:

$$\mathbf{mutation[k]} = \mathbf{population[a][k]} + \mathbf{F} * (\mathbf{population[b][k]} - \mathbf{population[c][k]})$$

(k varying from 1 to 3 and F being a constant)

The solution pool is then evolved by computing the rastrigin error and thresholding the mutant. Input to the DE is sampled from Classical and Quantum Random number generators (RNGs) for comparison.

2. SCOPE OF WORK

The present work is pre-requisite for further research in the field of evolutionary algorithm implementation on Quantum machines involving Human Interaction because to study fitness functions with unknown form (which is essentially HCI), it is first necessary to fully understand known fitness function behavior on Quantum Systems such as the rastrigin function and rosenbrock function which is preliminarily investigated in this work.

3. SIGNIFICANCE OF THIS WORK/APPROACH IN THE CONTEXT OF EXISTING LITERATURE

The use of statistical methods, tools, and analyses, for the purpose of rigorous and objective, interpretation, and comparison, stems from several well-established works employing this approach. Following inspiring reference(s) [1, 2] exemplify to this end:

Computational Topology for Data Analysis Tamal Krishna Dey, Yusu Wang , 2016-2021, Cambridge University Press.

(section 13.3)

“these are important topics for the development of topological data analysis, e.g. leading to more rigorous quantification of uncertainty, noise, consistency and so on.”

A Riemannian Framework for Statistical Analysis of Topological Persistence Diagrams Rushil Anirudh, Vinay Venkataraman, Karthikeyan Natesan Ramamurthy, Pavan Turaga **(section 6. Discussion and Conclusion)** [2]

"Based on the theory and experiments presented so far, it is instructive to compare the space of PDs with respect to the different distance metrics. We will consider only the L2-Wasserstein metric (dL2) and the proposed Hilbert sphere metric dH . The interpretation of PDs with respect to these two metrics is very different. dL2 (X, Y) is the Earth Mover's Distance between the PDs X and Y considered as scaled discrete probability mass functions (pmfs). However, dH (ψX , ψY) is the geodesic distance between the square root of kernel density estimates of X and Y ."

[2] and many other related works (as can be obtained by traversing the references flower on platforms such as arXiv).

As mentioned above [2], although the use of statistics as a rigorous tool is known, to the best of research, knowledge and belief of the Author the approach has not been applied, prior to this work, to the comparative analysis, of Quality of Random number generation across Classical and Quantum simulation; and implication of the same in the context of applying the generated random numbers to the Differential Evolution algorithm which enables Benchmarking of known fitness functions viz. rosenbrock and rastrigin as done in this work.

The result discussion and conclusion sections of this work elaborate and illustrate the same by showing clearly and consistently the fact that Quantum RNG is higher in quality than classical RNG making the case for use of Quantum RNG in mission critical applications such as Human Computer Interaction (HCI) where an amalgamation of fields such as Evolutionary Algorithms and Quantum computing needs to be envisaged.

4. SPECIFICATION DETAILS OF EXPERIMENT APPARATUS (NUMPY PYTHON LIBRARY (CLASSICAL TURING COMPUTATION) AND MICROSOFT Q# (Q-SHARP) LOCAL QUANTUM SIMULATOR)

Turing Computation : python classical NumPy library :

Python 3.11.4 (tags/v3.11.4:d2340ef, Jun 7 2023, 05:45:37) [MSC v.1934 64 bit (AMD64)] on win32

Quantum Computation : Microsoft Q# (Q-sharp) local Quantum Simulator

qsharp 0.28.277227 pypi_0 pypi

Limitation of the quantum simulator: The simulator simulates[3, 4], the Level 1 Physical Qubit Quantum Computing Implementation Level on a classical AMD64 PC

https://cloudblogs.microsoft.com/quantum/2023/09/18/azure-quantum-learning-resources-enable-getting-ready-for-a-quantum-supercomputer/

and therefore, is only a very limited simulation of the computational power of a real Quantum computing backend.

5. DIFFERENTIAL EVOLUTION (DE) (ALGORITHM) PSEUDOCODE

Pseudocode of function invoking the rastrigin and rosenbrock functions (Fitness functions for thresholding mutant individuals(candidate solutions generated by mathematically combining parent individuals))[5–7].

- Set up the mutating and thresholding parameters:
- dim = 3 (Number of dimensions of the population individual's input parameter to the fitness function)
- pop_size = 50 (Number of individuals (candidate solutions) in the input)
- F = 0.5 # mutation (The multiplier used in the recombination of individuals)
- cr = 0.7 # crossover (The threshold meta heuristic which determines the inclusion of the new mutant individual by replacing an existing member of the population)
- max_gen = 200 (The number of generational iterations before which the best solution generated using the logic as above is accepted as final).
- Initialize the population of candidate solution individuals by sampling the probability from the quantum simulator(by means of executing H Gate on qubits) (for the quantum case [8, 9]) and from the numpy library(for the classical case).
  - (N.B.:- Certain nuances must be dealt for the rosenbrock function due to the constraint on its definition only between population values -30 to 30 and this influences some convergence results)
- Compute the population error vector by calling the rastrigin or rosenbrock error computation function which returns the error difference between the functions' known global minima and the functional value for individual members of the population.
- Randomly pick 3 population individuals and mathematically combine them to produce new mutant individual.
- Compute the error for the new mutant individual by calling the rastrigin or rosenbrock function on the mutant.
- Randomly generate another probability and threshold it with the crossover to accept the mutant by replacing a member of the population or rejecting the mutant.
- Repeat steps 4 to 6 for the number of maximum generations setup in step 1 keeping a record of the generational best solution (solution with minimum error).
- Obtain the final best solution by finding the solution with minimum error from the generational best solutions.

6. CONVERGENCE RESULTS

The results herein lend unequivocal credence to the factthat Quantum Random Number Generators generate a better quality of Random numbers which result in significantly better convergence [10], results as shown by the summary and illustrations below:

On the **X axis** is plotted the number of maximum generational iterations of the DE Algorithm (described above)

On the **Y axis** is plotted the index of the population individual where the best solution (minimum error) has been found

It is evident that the rastrigin and rosenbrock functions converge quicker (more favorably/optimally) when QRNG is utilized as opposed to Classical (Turing) model.

For rastrigin

QRNG convergence point: (6,10) (As shown in FIGURE 1)

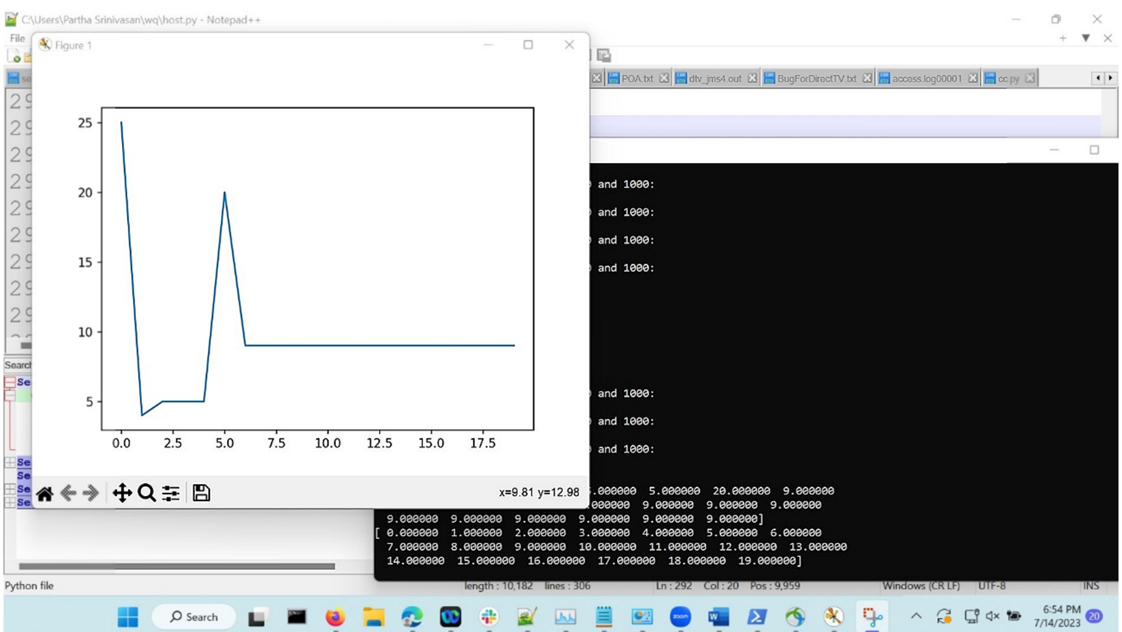

Figure 1: QRNG Convergence for rastrigin

Classical convergence point : (11,17) (As shown in FIGURE 2)

For rosenbrock

QRNG convergence point : (11, 50) (As shown in FIGURE 3)

Classical convergence point : (85,0) [Corrected per re-observation] (As shown in FIGURE 4)

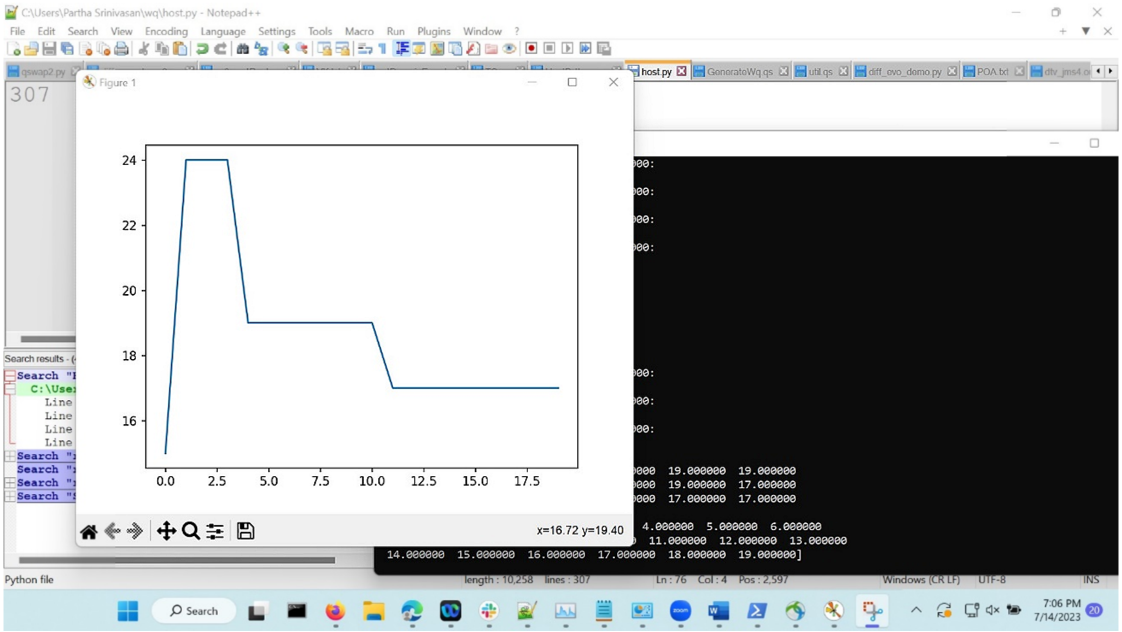

Figure 2: Classical Convergence for rastrigin

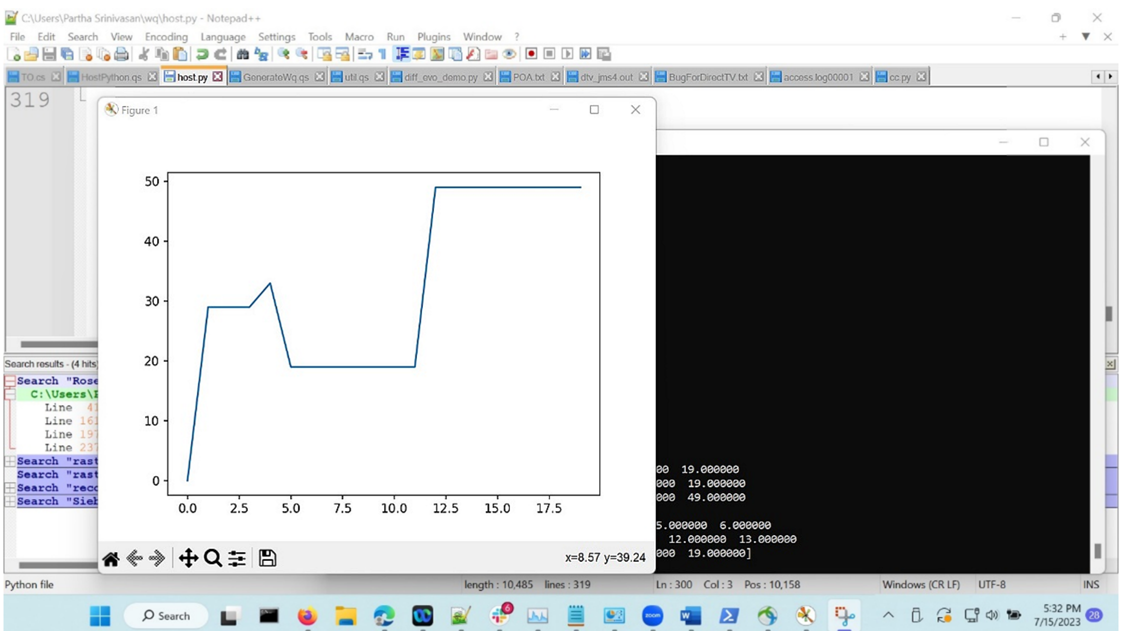

Figure 3: QRNG Convergence for rosenbrock)

Supplemental illustrations

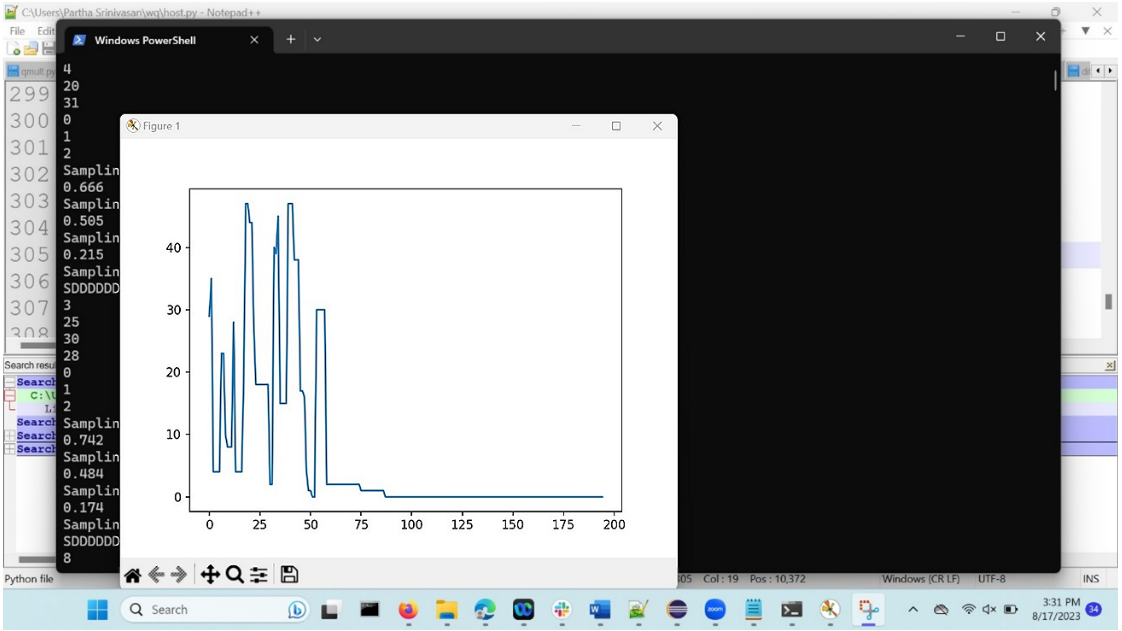

Figure 4: Classical convergence for rosenbrock

7. Statistical Result Analysis

7.1 Reason for Choice of Mann-Whitney U test for computing and comparing p-values of logical data groups.

p-values are a good indicator of the result of statistical comparison between groups of data similar in their classification and measurement and their statistical distance from normality/ideality reference. The Mann-Whitney test provides the most robust tool (in terms of accuracy and reliability) for obtaining and comparing p-values in cases such as this one in consideration since the test rigorously brings to fore the comparative Quality of the Random numbers generated, in terms of correlation and statistical separation (distance) of the value distributions.

7.2 Following is result of statistical comparison (using www.statskingdom.com) of : Group 1 : Quantum Rastrigin , Group 2 : Quantum Rosen (RNG quality derived)

Two sample Mann Whitney U test, using Normal distribution (two-tailed) (validation)

The **normal approximation** is used. The statistic's distribution is N$(24.5, 7.766^2)$.The data contains ties, identical values, it is recommended to use the **normal approximation** that uses the ties correction.

1. H_0 hypothesis

Since p-value > α, H_0 cannot be rejected.The randomly selected value of **Group1's** population is assumed to be **equal to** the randomly selected value of **Group2's** population.In other words, the difference between the randomly selected value of **Group1** and the **Group2** populations is not big enough to be statistically significant.

2. P-value

The p-value equals 0.07142, ($p(x \leq Z) = 0.03571$). It means that the chance of type I error, rejecting a correct H_0, is too high: 0.07142 (7.14%).The larger the p-value the more it supports H_0.

3. The statistics

The test statistic Z equals -1.8028, which is in the 95% region of acceptance: [-1.96 : 1.96].U=10, is in the 95% region of acceptance: [9.2793 : 39.7207].

4. Effect size

The observed **standardized effect size**, $Z/\sqrt{(n_1 + n_2)}$, is **medium** (0.48). That indicates that the magnitude of the difference between the value from **Group1** and the value from **Group2** is medium.The observed **common language effect size**, $U_1/(n_1 n_2)$, is **0.2**, this is the probability that a random value from **Group1** is greater than a random value from **Group2**.

7.3 Following is result of statistical comparison of: Group 1: Quantum Rastrigin, Group 2: Classical Rastrigin (RNG quality derived)

Two sample Mann Whitney U test, using Normal distribution (two-tailed) (validation)

The **normal approximation** is used. The statistic's distribution is $N(24.5, 7.783^2)$.The data contains ties, identical values, it is recommended to use the **normal approximation** that uses the ties correction.

1. H_0 hypothesis

Since p-value > α, H_0 cannot be rejected.The randomly selected value of **Group1's** population is assumed to be **equal to** the randomly selected value of **Group2's** population.In other words, the difference between the randomly selected value of **Group1** and the **Group2** populations is not big enough to be statistically significant.

2. P-value

The p-value equals 0.1231, ($p(x \leq Z) = 0.06156$). It means that the chance of type I error, rejecting a correct H_0, is too high: 0.1231 (12.31%).The larger the p-value the more it supports H_0.

3. The statistics

The test statistic Z equals -1.5418, which is in the 95% region of acceptance: [-1.96 : 1.96].U=12, is in the 95% region of acceptance: [9.2454 : 39.7546].

4. Effect size

The observed **standardized effect size**, $Z/\sqrt{(n_1 + n_2)}$, is **medium** (0.41). That indicates that the magnitude of the difference between the value from **Group1** and the value from **Group2** is medium.

The observed **common language effect size**, $U_1/(n_1 n_2)$, is **0.24**, this is the probability that a random value from **Group1** is greater than a random value from **Group2**.

7.4 Discussion on statistical analysis of convergence data

The interpretation of the above two statistical analysis clearly and consistently establishes the fact that Quantum RNG is higher in quality than classical RNG as viewed by a comparison of the p-value from the cases above : Case 1 -> p-value 0.07142 (comparison of the 2 quantum cases) ; Case 2 (comparison of 1 classical and the corresponding quantum case) -> p-value 0.1231 . This is substantiated by the derivation that in Case 1 the chance of Type I error is ~ half of that in Case 2 and consequently better convergence.

8. TIMING RESULTS (RASTRIGIN)

Table 1: (Timing values of probabilistic(classical) operation in seconds)

Sample size 10	Sample size 50	Sample size 100
0.0029854774475097656	0.00725	0.015338
0.00759577751159668	0.088405609130	0.013422727584838
0.013752460479736328	0.052	0.0591559410095214

Table 2: (Timing values of quantum (probability generation operation in seconds). (Number of generations 10, 50,100) on y axis.

Sample size 10	Sample size 50	Sample size 100
0.0029854774475097656	0.00725	0.00759577751159668
0.013752460479736328	0.088405609130	0.052
0.015338	0.013422727584838	0.0591559410095214

9. RESOURCE UTILIZATION (AS EXPECTED)

Classical (Turing Model)

CPU usage : 3% at 1.17 GHz.
Memory usage : 13.1 of 44.9 GB (As shown in FIGURE 5)

Quantum (RNG)

CPU usage : 15% at 1.07 GHz.
Memory usage : 12.9 of 44.9 GB (As shown in FIGURE 6)

Supplemental Illustrations

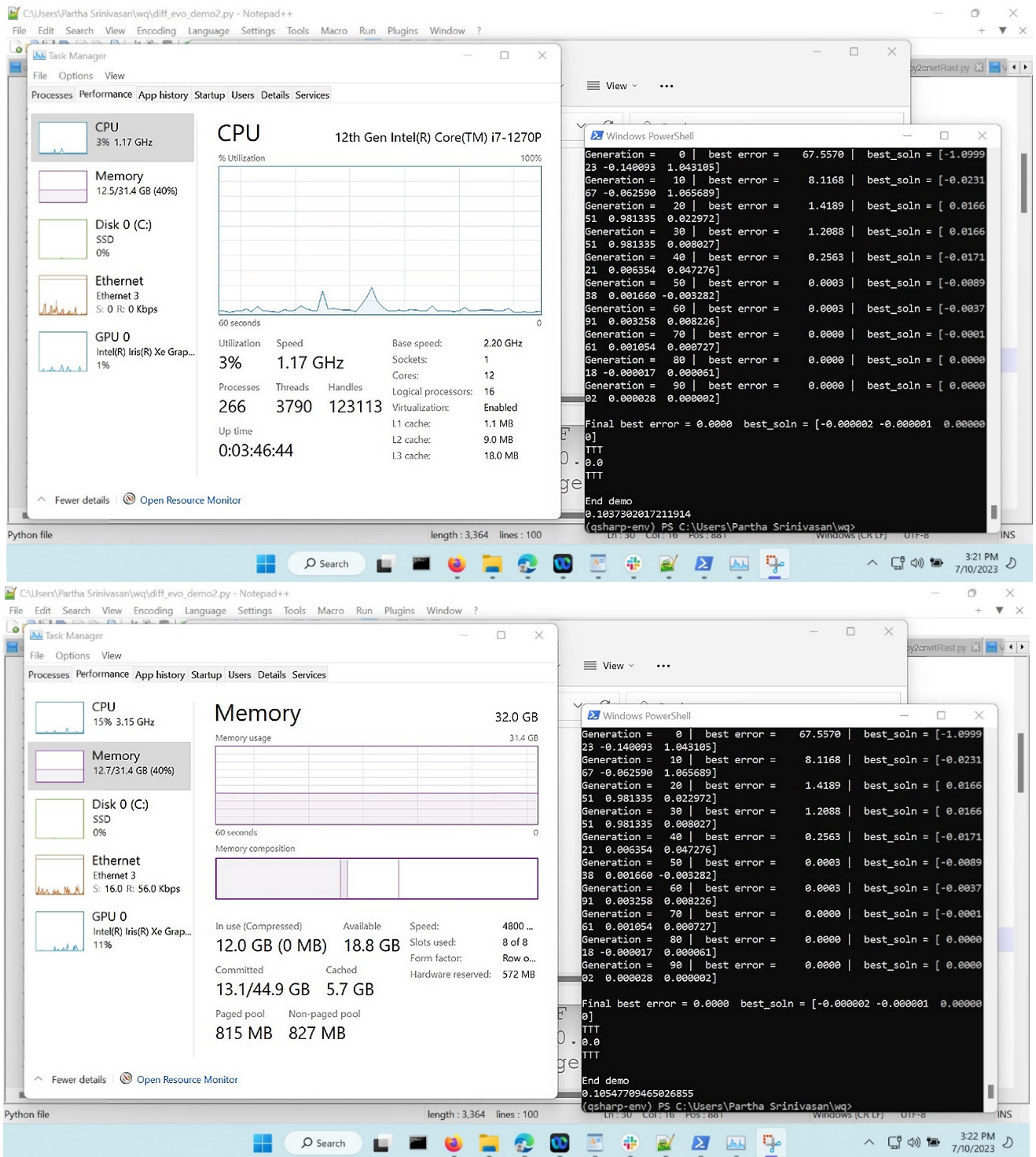

Figure 5: Resource utilization for Classical (Turing Model) RNG case

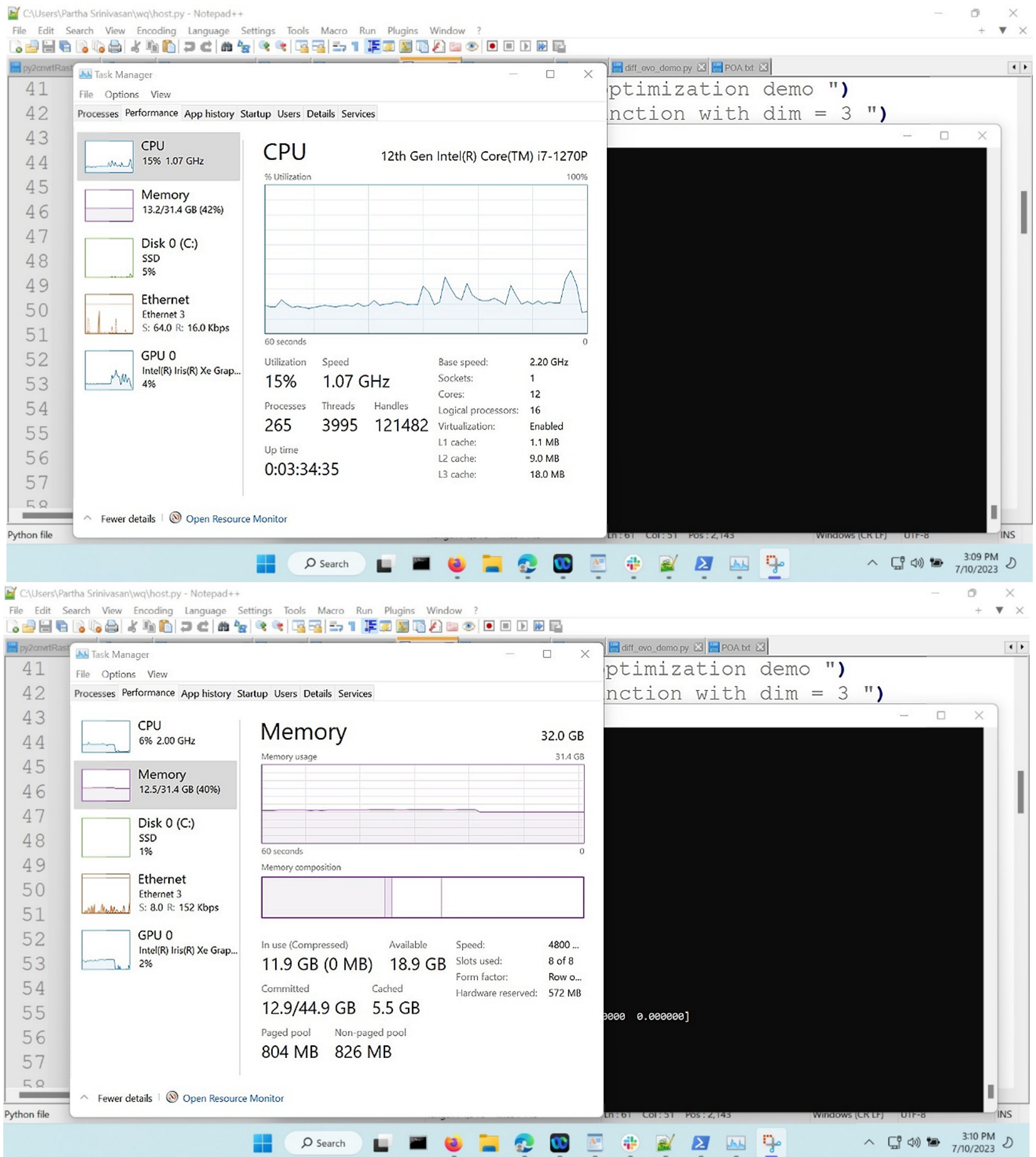

Figure 6: Resource utilization for Quantum (RNG) case

10. CONCLUSION

Contrary to intuition the quantum simulator does not live up to the timing measure when compared against the classical computer for the critical probability generation operation. (As shown in Table 1 and Table 2)

One should reasonably expect logically that working with an actual Quantum system (in place of the Quantum simulator) should amplify and yield consistent statistical results to those obtained and interpreted in this work with the simulator.

A brief digression [11], to the below excerpt helps to better elucidate the reason for the above expectation :

Quantum theory, the Church-Turing principle, and the universal quantum computer
DAVID DEUTSCH
Appeared in Proceedings of the Royal Society of London A 400, pp. 97-117 (1985)
(Communicated by R. Penrose, F.R.S. — Received 13 July 1984)

"**Perfect simulation of arbitrary finite physical systems:** The dynamics of quantum computers, though by construction 'finite', are still unphysical in one important respect: the evolution is strictly unitary. However, the third law of thermodynamics (1.3) implies that no realizable physical system can be prepared in a state uncorrelated with systems outside itself, because its entropy would then be zero. Therefore, every realizable physical system interacts with other systems, in certain states. But the effect of its dynamical coupling to systems outside itself cannot be reduced to zero by a finite process because the temperature of the correlation degrees of freedom would then have been reduced to zero. Therefore, there can be no realizable way of placing the system in states on which the components of the time evolution operator which mix internal and external degrees of freedom have no effect."

The above surmise automatically implies that in a Quantum Simulator running atop a classical Turing computer incurs overhead in setting up and emulating "pseudo" quantum states. Therefore, the above surmise offers the only logical explanation for the quantum simulator not meeting the timing expectations for RNG when compared to classical computation (NumPy library) on the same (classical) hardware. Due to this fundamental limitation with the Quantum simulation it **should be** reasonable to expect that the simulation performs (timing wise) not as good as real quantum backends regardless of the choice of Turing hardware/and or software for the simulation.

Perhaps a real quantum backend surpasses this shortcoming [12]. However due to integration and availability problems the Microsoft Azure quantum backends could not be utilized in this work but may be part of derivative works. Quantum machines have a little ways to go.

Having said so, the **quality** of random numbers produced by the simulator is evidently better and the implications such as convergence and timing have been discussed herewith in section 8. (Reproduced here for ready reference)

Discussion on statistical analysis of convergence data

The interpretation of the above two statistical analysis clearly and consistently establishes the fact that Quantum RNG is higher in quality than classical RNG as viewed by a comparison of the p-value from the cases above : Case 1 -> p-value 0.07142 (comparison of the 2 quantum cases) ; Case 2 (comparison of 1 classical and the corresponding quantum case) -> p-value 0.1231 . This is substantiated by the derivation that in Case 1 the chance of Type I error is ~ half of that in Case 2 and consequently better convergence.

The implication of lower Type I error is that the groups being compared bear more similarity.

This means the two Quantum cases resemble each other more (and also are closer to normality) as against the comparison between the classical and quantum case groups and this gets consistently reflected in the pattern of convergence for both the rastrigin and rosenbrock cases.

11. QRNG (PROVIDER) CURRENT LIMITATIONS AND SCOPE FOR FUTURE WORK

Integration Problem:

3 of the 4 quantum available backends have Integration Problems at present in that they are not capable of returning Integer values. This work serves to make this problem visible and public so that this gets addressed in future quantum offerings by quantum providers thus, enabling a smoother design and development facilitation for quantum computation consumers.

Availability Problem: Many of the high end quantum backends have limited availability at present again as found through this work and those that are available don't integrate suitably as above.

Future work should reasonably involve real quantum backend machines for obtaining results for comparison in place of the quantum simulator utilized in this work.

Also, it is suggested use other benchmark functions (few of which are mentioned by [Charilogis [13]) and cross verify results both with quantum simulators and real quantum backends.

Also, this work is a starting point for further studies on application of quantum systems to similar problems such as solution of randomized fractional differential equations.

12. ACKNOWLEDGMENT

Am grateful to the valuable comments of the reviewers.

References

[1] Dey TK, Wang Y. Computational Topology for Data Analysis: Notes From Book By. Cambridge University Press. 2016-2021.

[2] Anirudh R, Venkataraman V, Ramamurthy KN, Turaga P. A Riemannian Framework for Statistical Analysis of Topological Persistence Diagrams. In Proceedings of the IEEE conference on computer vision and pattern recognition workshops. 2016:68-76.

[3] https://cloudblogs.microsoft.com/quantum/2023/09/18/azure-quantum-learning-resources-enable-getting-ready-for-a-quantum-supercomputer/

[4] https://towardsdatascience.com/building-your-own-quantum-circuits-in-python-e9031b548fa7

[5] https://visualstudiomagazine.com/articles/2021/09/07/differential-evolution-optimization.aspx

[6] https://cse.engineering.nyu.edu/ mleung/CS909/s04/Storn95-012.pdf

[7] https://homepages.ecs.vuw.ac.nz/ mengjie/students/jasonPhd_thesis.pdf

[8] https://learn.microsoft.com/en-us/azure/quantum/tutorial-qdk-quantum-random-number-generator?view=qsharp-preview&tabs=tabid-copilot

[9] Ma X, Yuan X, Cao Z, Qi B, Zhang Z. Quantum random number generation. npj Quantum Information. 2016;2:1-9.

[10] https://medium.com/data-convergence/data-convergence-quantum-computing-algorithms-4545e1f10f33

[11] Deutsch D. Quantum Theory, the Church–Turing Principle and the Universal Quantum Computer. Proc. R. Soc. Lond. A. Mathematical and Physical Sciences. 1985;400:97-117.

[12] Jacak MM, Jóźwiak P, Niemczuk J, Jacak JE. Quantum Generators of Random Numbers. Scientific Reports. 2021;11:16108.

[13] Charilogis V, Tsoulos IG, Tzallas A, Karvounis E. Modifications for the Differential Evolution Algorithm. Symmetry. 2022;1:447.