Abstract—Background: Quantum computing is a new field of computer science that seeks to introduce a new paradigm for computing faster than classical computers. Research into algorithms that are uniquely efficient on quantum computers has been ongoing since the introduction of Shor’s algorithm for integer factorization in 1994. In particular, the intractable algorithms in NP are desperate for a more efficient solution that could be provided by a quantum computer – so much so that the P = NP problem is currently the most important question in theoretical computer science. One such NP problem to be examined is the Travelling Salesman Problem. Precise solutions of the TSP can be calculated in no better than O(n22n), becoming intractable at impractically small values of n; even the best approximation algorithms are limited to values of n in the thousands – requiring potentionally hundreds of CPU days to complete. Thus is it critical to explore the capabilities of a quantum algorithm in solving the TSP

Methods: Such algorithms can currently be ~~simulated~~ implemented? using the Qiskit ~~Aqua library~~, an open source project that provides applied science researchers the ability to utilize these quantum algorithms to advance their work. It is the intent of this project to contribute to the Qiskit library with the code and documentation of a TSP algorithm based on the algorithm proposed by Srinivasan et al (Srinivasan, Satyajit, Behera, & Panigrahi, 2018).

Anticipated results:

# I. Background

## A. Quantum computing

Quantum computing is a new field of computer science that seeks to introduce a new paradigm for computing faster than classical computers.

(National Academies of Sciences, 2018), (Deutsch, Ekert, & Lupacchini, 1999)(C. H. Bennett, 1973)(Charles H. Bennett, Bernstein, Brassard, & Vazirani, 1997), (feynman…)

*Current models for quantum computing*. Annealing/adiabatic vs gated (Albash & Lidar, 2018), (Gottlieb, 2018) , Noisy Quantum Computers (NQC) (NISQ?)

*Current state and projections for quantum computing.*(National Academies of Sciences, 2018), (Cervera-Lierta, 2018)

*Public access Quantum Platforms* (Microsoft, 2017), IBM Qiskit (<https://medium.com/qiskit>), (Qiskit, 2017/2019),(Qiskit, n.d.), Project Q,

## B. Quantum Algorithms

Research into algorithms that are uniquely efficient on quantum computers has been ongoing since the introduction of Shor’s algorithm for integer factorization in 1994.

*Qubits, Entanglement, and Unitary operators in gated quantum computers.* (Zhi-Yu & Shang-Wu, 2009), (Preskill, 2013)

## C. Algorithm complexity and combinatorial problems

In particular, the intractable algorithms in NP are desperate for a more efficient solution that could be provided by a quantum computer – so much so that the P = NP problem is currently the most important question in theoretical computer science.

## D. Traveling salesman problem

One such NP problem to be examined is the Travelling Salesman Problem. Precise solutions of the TSP can be calculated in no better than O(n22n), becoming intractable at impractically small values of n; even the best approximation algorithms are limited to values of n in the thousands – requiring potentionally hundreds of CPU days to complete.

*Significance of the TSP.*

*Classical computing approaches.*

*Quantum TSP algorithms for Gated QC.* (Goswami, Karnick, Jain, & Maji, 2004), (Stechly, n.d.), (Roy, Chawdhury, & Nayek, 2009), (Bang et al., 2012), (Srinivasan et al., 2018)

## E. Present purpose

Such algorithms can currently be ~~simulated~~ implemented? using the Qiskit Aqua library, an open source project that provides applied science researchers the ability to utilize these quantum algorithms to advance their work. It is the intent of this project to contribute to the Qiskit library with the code and documentation of a TSP algorithm based on the algorithm proposed by Srinivasan et al .

# II. Methods

## A. Software tools

(Stechly, n.d.), (Qiskit, 2018/2019)

(Pistoia & Gambetta, 2018), (Qiskit, 2017/2019)

# III. Anticipated results

# IV. Abbreviations

# References