

Academy of Science - Independent Researcher

Using a Quantum Genetic Algorithm to Optimize the Spending of a Cyber Security Budget

Author:
Ayush Hariharan

Research Mentor:
Duke Writer

Abstract

Although computers have increased in computational ability, one category of problems, the NP-Complete problem set, still alludes modern technology. However, quantum computing, a computational method that utilizes quantum mechanics to overcome the limitation of a two-state system, has emerged as a potential solution. The uniqueness of this researcher's current work starts with reducing the time complexity of NP-Complete problems, specifically the Knapsack Problem. By taking advantage of the Qiskit Toolkit and the IBM Qasm Simulator, an evolving qubit population was modified using a quantum-inspired genetic algorithm. With a two-tailed p-value of 0.0054 when compared to previous research, the quantum algorithm provided a better optimization of the knapsack problem. However, one caveat of all genetic algorithms is their ability to fall into local maxima. To avoid this issue, the current research implements a disaster algorithm. After implementing this algorithm, without tuned hyperparameters, the optimization of the knapsack problem increased by 3.75%. Since the proposed new algorithm had significant results, it can be utilized for various applications of the knapsack problem. One novel and concrete application explored by this research deals with the budget allocation for a Small-Medium Enterprise (SME) to minimize the impact of hacker exploits. Using a capture-the-flag model, the known vulnerabilities as well as the confidentiality of the data can be used as inputs to calculate explicit and implicit costs. On top of that, specific controls implemented by SME can mitigate the impact of the exploit, thereby lowering explicit and implicit costs. The new proposed algorithm, and respective application of the knapsack problem, can optimize the mitigation of a given exploit via the implementation of specific controls while remaining under a budget.

1 Background Information

Information Security

With the advancing technology present in modern society, corporations and individuals alike are shifting their focus towards an online world. From cloud storage services such as Heroku and Amazon Web Services to web applications such as Facebook, the internet connects people around the world and makes life easier for most societies. Simply put, the internet represents convenience in the eyes of individuals and its human nature to abuse that convenience. So, most private information, from credit card numbers to social security pins, are now being shared online. In the United States alone, Internet traffic for all United States banks grew by 77.6% in a single year, highlighting the imminent shift towards online banking [12]. And with this influx of private information on the internet, the security of that information gets called into question. Especially with the recent hacks by Russia and China onto the United States, it is apparent that databases are not strong enough to keep this information secure. Breaches in security happen way to often and the information of 44 million consumers is dangled in the air [16].

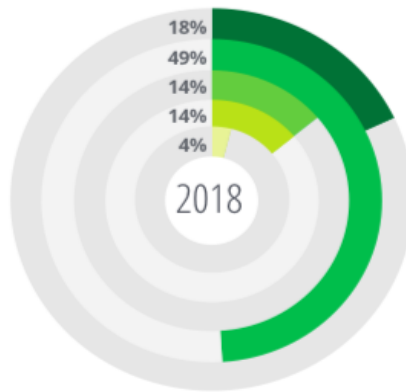
With this increasing pressure on companies to keep online information secure, the field of cyber security had blossomed. Since most systems are open to security breaches by worldwide hackers, the importance of cyber security in this modern era has been emphasized. cyber security is the practice of defending private and public information from potential hackers who are accessing the system through an unauthorized manner. However, the field is generally composed of firms who are contracted by banks and other corporations to keep the information secure. ~~Now, the information that is being targeted by hackers and exploitation is in the hands of cyber security firms, either small or big.~~

Small-Medium Enterprises (SME)

When looking at the field of cyber security, there are two major firm-types present: bigger firms and small-medium enterprises. Bigger firms are firms that have a constant influx of money from the government and have substantial changes to their budget each year. On the other hand, small-medium enterprises, like their name suggests, are smaller firms that are heavily-restricted with the available funding for cyber security, generally working with a fixed budget [5]. This smaller budget forces trade-offs within the company where the chief technical officer has to make important decisions. Regarding information security, the bigger firms have layers of security protecting the information making it more secure when compared to a smaller firm that doesn't have the resources to implement that security. For this reason, approximately 72% of cyber breaches occur at Small-Medium enterprises making the information more insecure when compared to bigger firms [5].

However, SME's do provide substantial benefit to their customers despite their decrease in size. With a smaller company, there is more flexibility for research and

development. This means that cutting edge technology, such as new threat detection systems and blue-team defenses, are being developed. On top of that, an SME can experiment with different ways to protect classified information or find patterns in data that haven't been found before. Due to the research and development nature of small-medium enterprises, they likely have more classified data when compared to bigger firms, who are trusted with more public data. Since the data is more classified, if there is a breach in a small-medium enterprise, the risks are significantly higher when compared to a breach in a larger company. Data that is vital to the survival of the country could be put in risk. So, it is important for SME's to implement controls that mitigate all of the vulnerabilities that are present in the system. Despite the importance of these controls, most Chief Information Security Officers are not confident in patching up all of the vulnerabilities under the budget constraint that is present in the company. For example, in a report published by Deloitte and NAISCO, 75.5% of CISO's cited a lack of sufficient budget as the top challenge [5]. Also, the study finds that 49% of cybersecurity firms only have 6 to 15 specialists working to protect the data.



Top Restriction for SME's [1]

Each color signifies a range of cybersecurity specialists. From the top (18%) to the bottom (4%), the first bar (18%) represents 1-5 full-time employees, the second bar (49%) represents 6-15 full-time employees, the third bar (14%) represents 16-25 full-time employees, the fourth and last bar (4%) represents 26-50 full-time employees. Since there is an abundant shortage of specialists in this field, action needs to be taken. For this reason, it is important to develop a system that will analyze the trade-offs of each control and advise the implementation of the controls that offer the greatest mitigation against an attack.

SME Investment Model

To deal with this advisement problem, Fielder et al. developed a cyber security model that utilizes game theory and existing data sets to model the decision making of these small-medium enterprises. When making decisions regarding the defense of a network, these SME's generally tend to consider two critical factors: the cost of implementing a particular defense and the impact that the defense has on the business [5]. By modeling the relationship between these two factors, an outside advisor can

tell the CISO how to appropriately spend the budget. In this game theory analysis, there are two individuals who are being modeled, the SME, which will be called the defender from now on, and the hacker who is trying to exploit the data, which will be called the attacker from now on. Since a company can only implement defenses that are within their limited budget, each defender has a budget B associated with it. In general, the budget will be used to protect against attacks on known vulnerabilities in the system as to improve the security of the data.

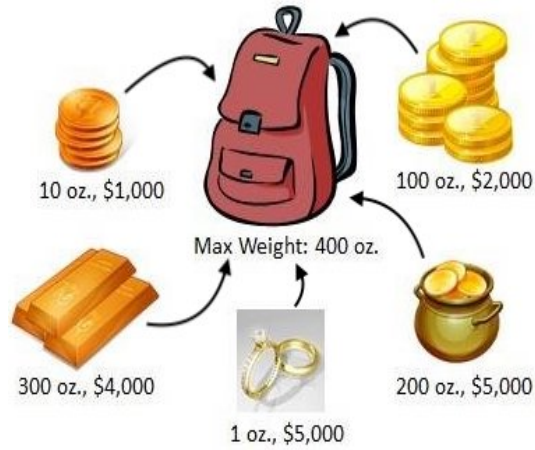
Now, like was mentioned before, the model itself describes the relationships between the attacker and the defender in a mathematical context. The defender is an SME manager that has to defend the organization's assets from cyber threats. In order to mitigate any potential business disruption and maintain the organization's reputation [5]. To protect against commodity attacks, the defender will implement control strategies and each control can be implemented at each level. On the other hand, the primary objective of an attacker is to collect data from his targets, which are defined as vulnerability and depth pairs. The vulnerability describes a known exploit for that target and the depth describes the confidentiality of the data that will be lost using that known exploit. Using these vulnerability and depth pairs, the total damage on the defender, both explicit and implicit, can be calculated. Depending on the depth of the data, impact factors, such as data loss, business disruption, and reputation, will change based upon a random variable. On the other hand, depending on the vulnerability of the target, threat factors, such as frequency of attacks, will change based upon a random variable. Depending on the number of impact and threat factors, the loss will vary proportionally. So, Fielder et al [5] has proposed a mathematical method to calculate the cyber security loss of a certain attack.

However, a defender is not going to wait for an attack to happen but will try and implement controls to mitigate the effect of an attack. Each one of these controls has a degree of vulnerability mitigation onto the attack and each control has a probability associated with it. In this case, the probability represents the likelihood that the defender will actually implement that control. Using those two measures, efficacy of the control can be calculated. Using the efficacy calculated with the control as well as the proposed cyber security loss, the implicit cyber security cost shall be calculated.

Now, there is a cost function, represented by the implicit cyber security cost, and the expected mitigation of the control. Using these two functions and the budget, B , this investment problem turns into a classic NP-Complete problem called the Knapsack problem.

Knapsack Problem

The knapsack problem is a NP-Complete optimization problem with the goal of finding, in a set of items with given values and weights, the subset of items with the highest total value, under a weight restriction [3].



Knapsack Problem [15]

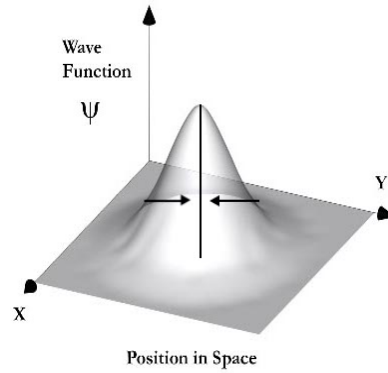
In this problem, a value has to be optimized while remaining under a constraint. A common interpretation of the knapsack problem is to maximize the fuel available to use in a rocket while remaining within a certain budget. Other interpretations include designing a system that can handle large an influx of data while remaining under corporate budget or determining which parts to purchase in order to increase the fuel efficiency of a car. In the case of cyber security investment, the mitigation of an attack has to be maximized while staying under the budget provided to the defender. However, the biggest application of the Knapsack function, is that the solution can be used to solve other NP problems through NP-Complete reduction. Scientists have tried to solve the knapsack problem classically and failed; however, with quantum computers providing a drastic increase in computational ability, new approaches have been designed to take advantage of a quantum computer's speed and general reliability.

Quantum Systems

With the increasing complexity of mathematical problems, quantum computing has become a popular solution to avoid long periods of computation. A quantum computer uses fundamental principles of quantum mechanics, including superposition and entanglement, to maximize computing power. In a traditional computer, data is encoded in bits, 0 and 1, which represent the on and off positions of a transistor; however, a quantum computer uses quantum bits, or qubits, to represent any of the infinite states between on and off. Each qubit is mathematically notated as a vector, with a horizontal component that represents the probability of being a 0 and a vertical component that represents the probability of being a 1. From this point onward, the horizontal component will be referred to as alpha, and the vertical component will be referred to as beta. Although it is impossible to determine the exact state of a qubit, due to uncertainty and superposition, vector notation allows scientists to approximate this value using the direction and magnitude of the qubit. In a quantum system, or a space with multiple qubits, a major characteristic is that the magnitude of all of the probability vectors in the system must equal 1. In this experiment, Dirac notation will be used where a bra vector represents a row of qubit vectors and a ket vector represents a column.

Another important aspect of any quantum system is the idea of measurement, which allows for a transformation of quantum states into classical states. A quantum entity, or system, **only comes into existence** and acquires definite properties when an observation or measurement is made [6]. An important part of quantum mechanics are wave functions which gives the mathematical probabilities of the state of a quantum entity before an observation has made [6]. Each possible probability in the wave function is commonly referred to as an Eigen function.

In order to obtain a measurable value from the system, the wave function must be collapsed into a specific Eigen function, **as can be seen in the image above**. For example, when a function is anti-differentiated, the antiderivative obtained could have various vertical translations. The abstract notation, using C to represent the translation, is an example of a wave function with a specific antiderivative being an Eigen function. However, to collapse this wave function a value for C must be determined.



Collapse of Wave Function – Modified from [18]

In quantum mechanics, a collapse of the wave function cannot be done by substituting a value; instead, energy must be added to a system. Energy is usually represented by a Hamiltonian quantum operator, which is a parameter used to translate quantum wave functions into classical Eigen functions. By adding energy to the system, the possibilities of a wave function are localized into a single particle and this localization results in measurement. However, the results from a measurement of the wave function is not random. According to the **Copenhagen interpretation**, every Eigen function has a certain probability of occurring during the measurement process. In terms of quantum computing, the separate values of 0 and 1 can be represented by a discrete particle while a qubit, or the infinite values between 0 and 1, can be represented by a wave function.

Therefore, during **measurement**, in the $\alpha^2 + \beta^2$ term, **if α^2 is higher than β^2 the measurement of the system returns a 0; however, if β^2 is higher than α^2 , the measurement of the system returns a 1**. In certain extreme cases, the value of $2\alpha\beta$ overpowers the values of $\alpha^2 + \beta^2$. In these cases, the value of the system might not collapse to either a 0 or a 1. However, since these cases are extremely rare, it is assumed in this research that the value of $\alpha^2 + \beta^2$ significantly overpowers the value of $2\alpha\beta$. In the future, the researcher plans to augment the uniqueness of this work by incorporating the edge cases where $2\alpha\beta$ overpower the value of $\alpha^2 + \beta^2$.

Quantum-Inspired Genetic Algorithm

An efficient and quick approach to this problem uses quantum-inspired genetic algorithms (QIGA). A standard genetic algorithm is a population-based search method which consists of 5 crucial components: initialization of a population, evaluation, parent selection mechanism, variation operators and survivor selection [8]. Once the population is initialized and the first evaluation is completed, a genetic algorithm takes advantage of the concepts of evolution and modifies a selected part of the population. After the selected portion is determined using a parent selection mechanism, those individuals are modified using variation operators and the algorithm is run again. Eventually, the population should “evolve” towards an optimal solution. Although genetic algorithms were a good way to optimize problems, classical computers did not have the computational ability to cycle through enough generations for the algorithm to be effective.

When applied to quantum mechanics, however, a quantum genetic algorithm can be used based upon the concepts of quantum bits and quantum superposition states. Using qubit encoding, where each qubit uses a vector to represent binary code, a quantum algorithm can use a quantum system with multiple qubit vectors as an initial population [7]. Specifically, each quantum gene is represented by a qubit vector, each quantum chromosome is represented by a bra vector, and each quantum population is represented by a column of chromosomes. When the quantum population is first created, each vector is initialized with a linear superposition of states, essentially setting both alpha and beta to $\sqrt{2}/2$, so that the probability of getting a 0 or a 1 during measurement is the same.

Once the initialization of the population has been completed, the entire quantum population is measured, using the Copenhagen interpretation, with each chromosome considered a single binary string, and each string evaluated with respect to the knapsack problem. The string which produces the highest profit while remaining under the constricting value is stored as best individual in that generation. The fitness of the best population is then plotted on a curve for future use. In order to create a new generation from the existing quantum population, each quantum gene, represented by a qubit vector, is rotated in the search space and this rotation results in new values for alpha and beta. When a new generation is created, the inherent randomness of quantum computing is abused since this randomness creates natural variability in the population. The probability amplitudes (alpha and beta) are plotted and the algorithm is repeated for multiple generations. The quantum population will eventually start to reach a local maximum and when the population reaches the maximum, the evolutionary algorithm is terminated [11]. Thus, the theory of the genetic algorithm remains the same but a QIGA manipulates qubit vectors, instead of classical values, to create increase optimization.

Higher-Order Quantum-Inspired Genetic Algorithm

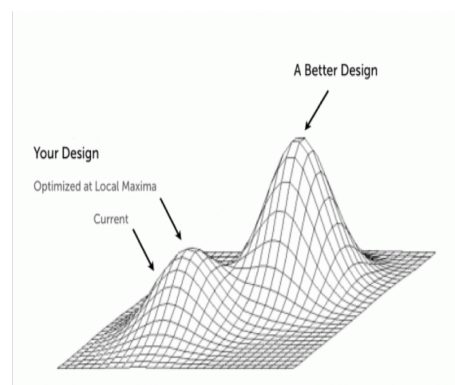
Previously, the QIGA had countless flaws, including the use of single qubit vectors for optimization and the minimal measurements of the quantum population. Re-

searchers from Poland fixed many of these issues by developing a higher-order quantum-inspired genetic algorithm, that uses quantum registers to handle the encoding instead of isolated qubit vectors [11]. A quantum register is essentially an operation that transposes the vectors of two different qubits onto each other. The benefit of using quantum registers instead of single qubits is that error in the algorithm is drastically minimized. QIGA algorithms with Order-r can give a more accurate representation of a solution to the knapsack problem since the ability to model relations between separate genes increases [11].



Apart from the use of a quantum register instead of isolated qubits, the QIGA-2 also has other features that help streamline the optimization process. After the initialization of the population, essentially the same process as the QIGA-1 except the genes are grouped into 2-bit quantum registers, the population is evaluated using a couple of methods. The measurement function returns one of the binary strings 00, 01, 10 and 11 with a probability of $\alpha_0^2, \alpha_1^2, \alpha_2^2, \alpha_3^2$ respectively [11]. Simply put, instead of measuring each individual vector, as in most quantum algorithms, the QIGA-2 assigns a two-bit value, from 0 – 3 in binary, to each register in the chromosome. The creation of a new generation is also handled differently. If the probability amplitude, of any of the four coefficient values in the quantum register, does not correspond to a specific pair of bits currently in the best individual, the probability amplitude will be decreased (amplitude contraction) according to the rule: $\alpha_{new} = \alpha_{old}$ where α is a parameter of the algorithm [11]. Once the manipulation is completed, the transposed vectors in each register end up rotated and the rotation value is noted.

After testing the new algorithm with different mathematical problems, including the knapsack problem, the polish team concluded that the stabilization, or optimization, value for their algorithm was significantly higher than previous genetic algorithms proposed to solve similar problems [11]. However, this algorithm still has countless flaws. The current setup of the code does not account for the stabilization that occurs when an environment reaches a local optimum. A local optimum is when the genetic algorithm prematurely converges onto a single value for a certain period of time giving the illusion that the absolute maxima, or extremity point, has been determined [11].



Local vs Global Maxima [2]

Disaster Algorithm

In recent years, the standard QIGA has been used as a foundation for newer algorithms that improve optimization and efficiency. A major research breakthrough in this field decreased the time of convergence in later stages of the evolutionary algorithm and improved local search performance by implementing a self-adaptive rotating angle strategy and a disaster algorithm [7]. The rotating strategy exploited different concepts of linear algebra to decrease the number of parents with each call of the algorithm. The disaster condition, however, imposed a self-made disaster once the genetic algorithm reached a premature convergence value. For example, in some functions, there are various relative maxima and the genetic algorithm might mistake a relative maximum for an absolute maximum. In a quantum system, the disaster was usually resetting the probability amplitudes of some vectors to a linear superposition of states. This prevents premature convergence and ensures that the highest value for the knapsack function is returned.

Although these researchers, from the College of Field Engineering in Nanjing, China, did not test their algorithm using the knapsack problem, their method required similar computational abilities. After testing, researchers concluded that the addition of the rotating technique and the disaster condition improved the times of convergence of the quantum-inspired genetic algorithm, and returned a higher optimization value [7]. However, the research that was performed by this team used isolated qubit vectors to encode qubit strings, as common with QIGA 1 algorithms, and the simplicity of the design could have caused a lower optimization value than initially expected.

Uniqueness of this Researcher's Current Work

The new research performed addresses the flaws in previous research and allows for greater theoretical optimization of the knapsack problem. In the new algorithm, the higher-order quantum-inspired genetic algorithm was replicated on the IBM Quantum simulator, using the POWER 9 chip, as opposed to the CUDA architecture. Theoretically, this better simulation of quantum mechanical concepts should allow for a better optimization of the knapsack problem. In addition to that, although the researchers provided a range for the amount of probability amplitude manipulation, they didn't provide a certain value. So, this research will tune that hyper parameter and find a value for for which the optimization of the knapsack problem is the greatest. Finally, In the new algorithm, a disaster condition will be added to the higher-order QIGA to prevent premature convergence. In fact, social disasters technique, which applies a catastrophic operator once a local maximum is reached is one of the best methods to obtain the global extremity point instead of the local extrema [14].

In the algorithm that will be created, the disaster condition will implement a simple quantum catastrophe operator. Therefore, by tuning the higher-ordered QIGA, the probability of reaching a global maximum will increase drastically, and the optimization value for the knapsack problem will be much higher. Finally, once the algorithm has been completed, it will be applied to the cyber security investment scenario and a spending plan for the small-medium enterprise will be determined. In previous research, the cyber security investment model was based upon simulated values and game theory. However, in this research, the threat factors and mitigation values will be obtained from real-life case studies. Specifically, the threat factors are compiled from 2018 Threat

Impact and Endpoint Protection Report. On top of that, the control mitigation values will be calculated based upon the matrix proposed by Rakes et al. Researching this topic will allow the scientific and engineering community to accomplish countless tasks that have a limitation factor and an optimization value.

Experimental Design



In this research problem, the extent to which the tuned QIGA-2 algorithm can optimize the spending of a cyber security budget is investigated. Using the model defined by Fielder et al. as well as some modifications to that model, the QIGA algorithm will be implemented and an optimal investment strategy can be determined. The project will consist of two phases where the first phase involves tuning the QIGA algorithm with the optimal probability amplitude manipulation and implementation of the disaster algorithm. In this phase, if the probability amplitude manipulation along with the disaster algorithm is implemented on a POWER 9 architecture as opposed to the CUDA architecture, then the QIGA-2 output, or the optimum profit for the Knapsack function, will be significantly higher than that of previous research. Since the POWER architecture is more sophisticated than the CUDA architecture, specifically relating to quantum computing concepts, running the newly created algorithm on the IBM Q32 should increase optimization of the knapsack problem. For the first phase, the independent variable is the manipulation of the probability amplitudes and the implementation of the disaster algorithm while the dependent variable is the optimization of the Knapsack problem.

The second phase of the project involves applying the tuned QIGA algorithm to the cyber security modeled previously defined and see if this algorithm can provide an optimal investment strategy for CISO's. In this phase, if the QIGA algorithm is implemented in the cyber security investment problem, then the investment strategy that is obtained will be better when compared to previous case studies. This is because the QIGA algorithm has a better optimization of the knapsack problem when compared to that of previous research; in this phase, the independent variable is the implementation of the QIGA algorithm with a cyber security purpose while the dependent variable is the investment strategy that is obtained after implementing the QIGA algorithm.

2 Method

A Higher-Order Quantum-Inspired Genetic Algorithm, or a QIGA-2, consists of six major steps and these individual steps are performed multiple times throughout the evolution process. Every time these steps are performed is called a generation and through multiple generations, the population should evolve to produce a greater optimization of the knapsack function. The steps in each generation derive from Darwin's theory of evolution which states that evolution is a cause of overpopulation, competition between individuals of the species, and eventual survival of the fittest. However, before implementing the QIGA-2 algorithm on the IBM Q32 computer, various libraries including Qiskit by IBM and NumPy must be imported in order to connect with the

quantum computer.

Defining the Population

The first section of the quantum genetic algorithm is reliant on the idea of over-population and natural variability in genes. As Darwin concluded, evolution is spurred when there are too little resources for too many individuals and random variability's in the population cause some individuals to be better than others. In the first step, the quantum population is created and initialized. In this case, the population is an array of 32 qubits, the maximum number of qubits available on the IBM Q32, and each quantum chromosome represents a row in that array (Fig. 1).

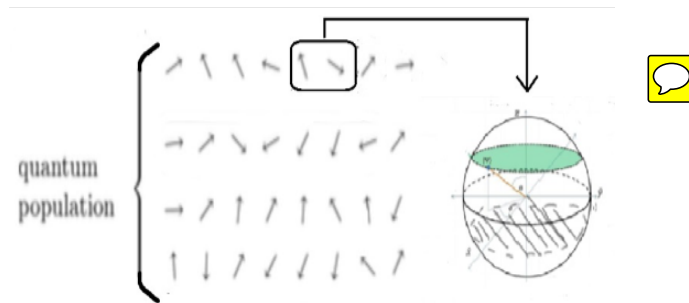


Figure 1: Quantum Population on Q32 [11]

Finally, every two qubits in that row are grouped together in qubit registers to represent different genes. The purpose of having multiple chromosomes is that each chromosome represents a different individual trying to pack the knapsack. By having more individuals attempting to find a solution to the problem presented, the chance of a non-obvious solution emerging is higher, and the optimization of the knapsack function increases drastically. In addition to that, the presence of multiple genes, or the implementation of qubit registers as opposed to individual qubits, allows for a more accurate interaction between different individuals. In terms of the cyber security application, each quantum chromosome represents an individual attack on the small-medium enterprise. And for each attack, the defender will implement certain controls and achieve a certain level of mitigation against the attack. Using another model proposed by Rakes et al., which considers the proportion of threats that survive when the countermeasure is implemented, the level of mitigation for the control will be considered. So, there will be four different defenders that are implementing four different investment strategies each with a different level of mitigation.

After the quantum population is created, each qubit register in that population is initialized with a linear superposition of states. Essentially, a linear superposition means that the probability of getting one and the probability of getting zero is the same for each qubit. By starting out with an equal chance of getting one and zero, there is natural variability in the population and the genetic algorithm can be more effective. In the case of the cyber security application, the linear superposition of states that the quantum population is initialized with represents an equal probability of a defender implementing a certain control and not implementing that control. Therefore, there is natural variability in the population and the genetic algorithm is effective.

Finding an Optimal Plan

Now that the population has been created and has been initialized with a linear superposition, the second section of the quantum genetic algorithm can begin. This section the individuals with the best random attributes are chosen and the fittest individuals in the population, or those who pack the knapsack the best, will remain for the next generation. After the initialization of the quantum population, it will be classically measured by collapsing the existing wave function. Depending on the alpha and beta values present in each qubit register, a binary string of size two, either 00, 01, 10 or 11, will be obtained from each register (Fig. 2).

$$|\psi\rangle = \cos\left(\frac{\theta}{2}\right)|0\rangle + \sin\left(\frac{\theta}{2}\right)|1\rangle$$

$$|\psi\rangle = \frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle$$

Figure 2: Quantum Measurement of Qubits [11]

During each fitness evaluation, or the conversion between quantum and classical states, the qubit register is measured 1024 times and the most consistent binary string was returned. The binary strings from every four registers will then be combined until four classical strings are obtained, one for each quantum chromosome in the population. After quantum measurement, using the Copenhagen interpretation, is complete, the binary strings are evaluated in order to determine the best individual. Depending on the binary string, which essentially represents the controls that are implemented by the defender, a certain mitigation will be received. In this case, if there is a 1 that means that the defender chose to implement a specific control. On the other hand, if there is a zero, then the defender didn't implement a certain control. Apart from the profit value, a total weight for the knapsack was also obtained, or the total cost of implementing all of the controls, and that total cost was compared to the cost cap, or the budget, set before the first generation. After each binary string is evaluated, the binary string, and respective quantum chromosome, with the highest profit value is stored as the best chromosome. With the application to a cyber security investment problem, the best chromosome will be the defender that implemented the controls that resulted in the greatest mitigation of the attack.

Since the best strategy to mitigate an attack on the defender's small-medium enterprise has been determined, the probability amplitudes of each qubit register will be manipulated in the following manner. Initially, the first binary pair in the best individual, b , will be obtained and depending on the value of that pair, the probability amplitudes will be manipulated accordingly. For example, if the pair is 10, then the probability amplitude for 10 will increase in a way such that the sum of the probability amplitudes in the quantum system remain under 1.0 (Fig. 3).

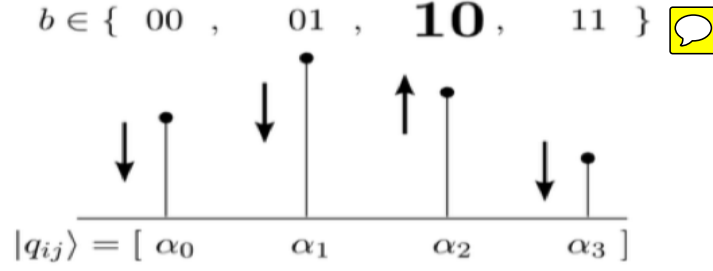


Figure 3: Manipulation of Probability Amplitudes [11]

On the other hand, all of the other probability amplitudes will decrease by a factor of γ . In this case, the probability of a defender implementing the successful controls increases while the probability of a defender implementing the unsuccessful controls decreases. The factor of probability manipulation, γ , is a hyper parameter of this algorithm and this hyper parameter was tuned by testing out different values for γ and observing the optimization of the knapsack problem. In the end, the controls that resulted in higher mitigation of an attack were encouraged while controls that resulted in lower mitigation of an attack were discouraged.

At this point, the algorithm has completed one generation of fitness evaluations. In order to obtain an accurate optimization of the knapsack problem, this process needs to be repeated for 10 generations in order to get a higher optimization of the knapsack problem. At the end of these ten generations, the data of final profit outputted by the QIGA-2 algorithm on the POWER architecture will be obtained and that data can be compared to that of previous research. However, now that the probability amplitude manipulation is complete and the optimal value has been determined, a disaster algorithm will be implemented. In this case, the disaster algorithm will simply reset the population to a linear superposition of states in the middle of the QIGA-2 algorithm. However, the generation that the disaster algorithm is implemented, is a hyper parameter of this algorithm and this hyper parameter was tuned by testing out different generations. In the end, the location of the disaster algorithm was chosen by the overall mitigation of an attack by the defender.

After one instance of the algorithm, or 10 generations, has completed its run-time, the data of final profit outputted by the QIGA-2 algorithm will be obtained and that data will be compared to that of previous research. On top of that, the different hyper parameters including the γ value and the generation for the disaster algorithm will be tuned.

Unique Application - Cybersecurity Investment

Once the hyper parameters are tuned using a common knapsack problem, the implementation of the knapsack problem in cyber security investment shall be determined. Essentially, instead of using profit and weight values to optimize the traditional knapsack problem, mitigation and cost values will be used. However, the transition from a traditional knapsack problem to a cybersecurity investment problem is not as simple as switching variables. In theoretical models, as proposed by previous researchers including Fielder et al., it is simple to adapt the model, that originally relied on game

theory and Nash equilibrium, to rely on the Knapsack problem. However, when using real-data collected by polling CISO's of small cybersecurity firms, the transition to the knapsack problem becomes much more difficult.

The uniqueness of this researchers current work emphasizes the importance of real-data when developing the investment plan. In order for this algorithm to impact CISO's in the industry, the plan must incorporate strategies from various case studies. When using a theoretical model, certain aspects about society are assumed, such as worker efficiency and management strategies. However, in the real-world, these aspects about cybersecurity firms are often completely different than originally assumed. For this case, a theoretical model will not provide a meaningful investment plan that is executable. On the other hand, in this research, the 2018 Threat Impact and Protection report, which has case studies of Small Medium Enterprise's spending plans and the respective mitigation on an attack, will be used to provide profit-weight pairs for the knapsack problem. Instead of generating these profit-weight pairs based upon a theoretical model, real case studies and survival probabilities are used to provide an executable investment plan for Chief Information Security Officers.


However, inputting real-life data into the knapsack problem is significantly more difficult than using a theoretical model. With the case study above, the 2018 Threat Impact and Protection report provides a matrix corresponding to 8 different threats on a cybersecurity firm and 8 different countermeasures that corresponds to each threat. However, the organizations of these threats and countermeasures make it difficult to translate to a single-choice knapsack problem. Each countermeasure has a vector with 8 components and each component corresponds to the countermeasure's effectiveness against each of the 8 attacks. So, with countermeasure 1, the first component of the vector corresponded to its effectiveness against the first attack. In addition to that, the second component of the vector corresponded to its effectiveness against the second attack. This process is then repeated for all of the countermeasures and all of the different threat values. However, an immediate problem with the interpretation of the report is that each countermeasure corresponds to a vector instead of a scalar value. Therefore, this research developed a method to translate that vector quantity into a scalar value. In this way, the dimensionality of the multi-choice Knapsack problem is reduced down to a single-choice Knapsack problem.

The methodology to reduce the dimensionality of the Knapsack problem relies on the idea of the dot product. Essentially, in the same 2018 Threat Impact and Protection report, there is another vector corresponding to the amount of cyber security loss with each threat attacked on the Small-Medium enterprise. By taking the dot product of this vector and the mitigation vector corresponding to each countermeasure, the amount of cybersecurity loss reduced by the implementation of the counter-measure is determined. For example, with the first countermeasure, the dot product will describe how much money is saved by the small-medium enterprise. Now, instead of a vector, the dot product reduces the dimensionality to a scalar quantity. In this way, the QIGA-2 algorithm can be used to advise a defender on how to optimally spend his or her money to receive the highest mitigation against an attack. The optimal investment plan that was determined by this algorithm will then be compared to the 2018 Threat Impact and Protection report which has case studies of Small Medium Enterprise's spending plans and the respective mitigation on an attack.

3 Results

The first phase of data collection involves tuning the actual hyper parameters of the quantum genetic algorithm. One hyper parameter that needs to be tuned is the value that describes how much the probability amplitudes are manipulated when the quantum population is evolving. This hyper parameter is important because it is fundamental when describing the probability amplitude manipulation that occurs in each generation of the genetic algorithm. It helps determine which cyber security controls are optimal and successful in mitigating attacks. Depending on the value for , the optimization of the knapsack problem can either increase or decrease. In this research, four different values for were tested including 0.8, 0.9, 0.95, and 0.99 and the final optimization of the knapsack problem was recorded.

Table 1: ~~Researcher's Simulation and Tuning~~ Results of QIGA Hyper parameters

| Statistics | Probability Manipulation | | | | | Disaster Algorithm | | | |
|------------|--------------------------|--------|--------|--------|--------|--------------------|--------|--------|---|
| Trial | 0% | 80% | 90% | 95% | 99% | 0 | 5 | 6 | 7 |
| 1 | 5662 | 8957 | 9528 | 9416 | 7004 | 9528 | 7503 | 7689 | 8853 |
| 2 | 6406 | 7509 | 7723 | 8039 | 5885 | 7723 | 7017 | 7117 | 843  |
| 3 | 6920 | 8631 | 10436 | 7108 | 7261 | 10436 | 8642 | 7277 | 9290 |
| 4 | 5980 | 8005 | 9352 | 7040 | 6135 | 9352 | 9920 | 7793 | 9696 |
| 5 | 9678 | 8149 | 7566 | 7462 | 8179 | 7566 | 8390 | 8398 | 8722 |
| 6 | 8775 | 8299 | 7324 | 6941 | 7030 | 7324 | 9323 | 8031 | 8154 |
| 7 | 9258 | 7798 | 8368 | 9753 | 7076 | 8368 | 8762 | 8338 | 8459 |
| 8 | 7070 | 9303 | 8504 | 7077 | 8418 | 8504 | 7945 | 8484 | 6486 |
| 9 | 6954 | 7799 | 6020 | 7161 | 8649 | 6020 | 7044 | 8232 | 7910 |
| 10 | 4097 | 8230 | 7008 | 9684 | 9064 | 7008 | 8448 | 8008 | 7736 |
| 11 | 8009 | 8658 | 8965 | 8453 | 6598 | 8965 | 9321 | 8696 | 7323 |
| 12 | 5388 | 7851 | 7508 | 8023 | 8261 | 7508 | 9138 | 8148 | 8272 |
| 13 | 7225 | 7807 | 8168 | 7885 | 9538 | 8168 | 9311 | 10983 | 8643 |
| 14 | 7786 | 7114 | 9071 | 7619 | 7887 | 9071 | 8493 | 8106 | 8120 |
| AVG | 7086 | 8150 | 8252 | 7975 | 7641 | 8252 | 8518 | 8235 | 8292 |
| SE | 414.21 | 156.51 | 310.67 | 266.80 | 293.33 | 310.67 | 236.19 | 241.81 | 214.09 |

*** Simulation design and results were generated by the researcher through defining and measuring the quantum population with a linear superposition of states, then manipulating the probability amplitudes of the qubits until a maximum knapsack optimization was achieved. To prevent convergence to a local maxima, researcher implemented a disaster algorithm at various generations (gen). See Figures 1-3 for detailed background and Figures 4-5 for summary of researcher's analysis of the simulations.

These specific values were chosen for a reason when dealing with probability amplitude manipulation. At first, it was determined that a higher probability manipulation value, such as 99%, would be the optimal manipulation value since that would drastically increase the probability of implementing a successful control. However, after running the algorithm, it was realized that an excessively high value would result in over fitting, where each defender was too reliant on the patterns of the best individual. In this case, the quantum genetic algorithm would again get stuck at a local maximum and the optimum investment plan would never be found. So, the value for was reduced to 80%. After running the algorithm at this manipulation, it was determined that the value was too low. With this value being too low, the defenders in this generation aren't as reliant on the patterns discovered in previous generations and a lower optimization was occurring. So, as a compromise, the value of was raised to 95% and then 90% to get a variety of probability amplitude manipulation coefficients.

After the data was collected regarding the hyper parameter manipulation, another vital hyper parameter was the generation which to implement the disaster algorithm. Depending on the generation that the disaster algorithm was implemented at, the manipulation of the knapsack function changed drastically. For example, if the disaster algorithm was implemented at a later date, then the behaviors of the defenders would change too late and the company would get stuck at the local maximum of mitigation. On the other hand, if the disaster algorithm was implemented at an earlier date, then the behaviors of the defender would change too early and the company would not receive the higher level of mitigation that the previous generation had gotten. Either way, there is a trade-off between getting stuck at a local maximum or not even reaching that local maximum. To resolve this trade-off, it is important to tune the disaster algorithm hyper parameter which was done in the data above. In this specific case, the data was collected starting at a disaster step of 7. It was initially assumed that at later generations of the quantum-genetic algorithm, the defender would likely get stuck in a local maximum and wouldn't be able to accurately optimize the investment problem that was presented earlier. However, after trying that out, it was determined that too little time was given to the defender for adaptation purposes. He or she wasn't able to adapt his or her strategy before the 10th generation had completed. So, the disaster step was slowly decreased from 7 to 6 and eventually to 5. For each of these disaster steps, the entire QIGA algorithm was run, with all 10 generations, and the final output was recorded.

Now that the hyper parameters have been tuned, the QIGA algorithm can be implemented in the context of the cyber security investment problem where a defender has to implement controls that result in the highest mitigation. Currently, the data is being collected regarding the application of the QIGA algorithm to the cyber security investment problem. However, when that data is collected, it will be similar to the data above except the final output would be a mitigation value instead of a dollar amount.

The total mitigation across 10 generations will be recorded and the QIGA algorithm will be run 14 times in order to determine whether the mitigation is better than that of existing case studies.

4 Analysis

QIGA Algorithm for Optimization of Knapsack Problem

After the initial data for comparison purposes was collected, without the manipulation of amplitudes (also referred to as 0% probability manipulation in Table 1), the evolution of the quantum-genetic algorithm was compared to that of previous research. In previous research, the general trend observed by Kucharski and Nowotniak was that the optimization of the knapsack problem increased via a logarithmic trend, or in some extreme cases in an exponential trend. Through past research, it had already been verified that an appropriate QIGA algorithm would evolve in a logarithmic pattern – making greater strides towards packing the knapsack problem earlier while getting slower and slower towards the end. Also, since the fit is logarithmic as opposed to exponential, there is no upper limit meaning that there is no definite solution to the knapsack problem. In order to verify the validity of the proposed algorithm, the maximum profit over 10 generations was checked for a logarithmic trend.

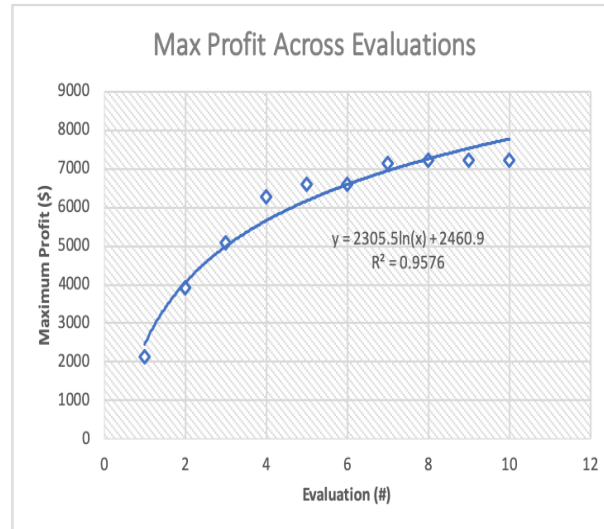



Figure 4: Researcher's Simulation of QIGA Evolution

The data, which showed the maximum profit in the knapsack over multiple generations, was fitted with a logarithmic trend line and an r-squared value of 0.9576 was obtained. This high r-squared value suggests that 95.76% of the variation in profit value was explained by the difference in generations during the quantum-inspired genetic algorithm. This logarithmic trend suggests that the profit in the knapsack increases rapidly during the first couple of generations but slows down in the higher generations. A pattern likely due to the higher influence of the weight restriction as more items are added to the knapsack. Initially, the large items are added to the sack since the influence of the weight restriction is not as prominent. However, in the later generations,

the smaller stacks of money are put into the plastic bag and the change in maximum profit varies minutely during this latter half. Finally, the error bars on the data points are fairly small suggesting that the variation from the trend line is minimal. Though these three trials, it was demonstrated that the quantum population was evolving as expected on the POWER architecture.

In order to establish the significance of the proposed algorithm in the context of a knapsack problem, the optimization that was obtained without any probability amplitude manipulation (or 0% probability amplitude manipulation as can be seen in Table 1) was compared to the highest value obtained by previous researchers \$5709. In order to compare the data that was collected to that of previous research, a 1-sample t-test with a significance level of 0.05 was run on the data and the results of the test were used to determine whether the QIGA-2 algorithm on the IBM Q32 simulator was significantly better than the algorithm which was run on the CUDA model. Since both of the p-values (one-tailed and two-tailed) obtained from the 1-sample t-test, 0.003 and 0.005 respectively, were less than the established significance level of 0.05, the null hypothesis could be confidently rejected. If the null hypothesis was true, as the proposed algorithm didn't perform any better than that of previous research, there would only be a 0.274 or 0.548 percent chance of obtaining a profit value this high or higher based upon random sampling variability alone. Therefore, there is enough statistical evidence to suggest that the proposed algorithm on the POWER architecture performed significantly better than that of previous research. In the end, the proposed QIGA-2 algorithm, on the IBM quantum simulators, performed better due to the simulation of more qubits and the existence of four quantum chromosomes in the population as opposed to three. These factors allowed for more genetic variability and increased the chance of finding the non-obvious solutions to the knapsack problem. Different routes were taken and ultimately, a higher optimization of the knapsack function was achieved.

However, this was just the basic QI  algorithm on the POWER architecture. In the finalized algorithm, multiple hyper parameters were tuned including the value of which described the amount of probability amplitude manipulation and the generation for which to implement the disaster algorithm. First, when regarding the manipulation of probability amplitudes, an analysis of variance test, or an ANOVA test, was run to determine whether the change in probability amplitudes did in fact achieve a statistically significant change in the optimization of the knapsack problem. The significance level of the ANOVA test was 0.10 and that represented the alpha power of this statistical test. When implementing the probability amplitude manipulation, the goal was to simply increase the optimization of the knapsack problem. So, the alpha power of the test didn't need to be extremely high so a value of 0.1 was chosen. Since the p-value of .052 is less than the significance level of 0.1, the null hypothesis could be confidently rejected. If the null hypothesis was true, as the proposed algorithm with manipulation of amplitudes didn't perform any better than the algorithm without the manipulation of amplitudes, there would only be a 5.27% chance of obtaining a profit value this high or higher based upon random sampling variability alone. Therefore, there is enough statistical evidence, within the power of this ANOVA test, to suggest that the manipulation of probability amplitudes increased the optimization of the knapsack problem. Since the ANOVA test doesn't provide the greatest value for , that had to be calculated

manually. When looking at the data table, it could be observed that a value of 0.9 or a 90% manipulation of amplitudes resulted in the highest optimization of the knapsack problem. In order to confirm the values obtained from the ANOVA test, the maximum profit over the probability amplitudes were observed.

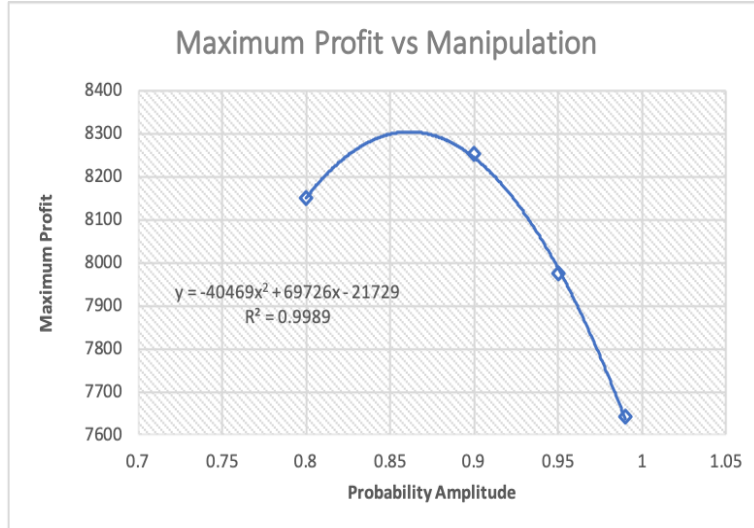


Figure 5: Researcher's Simulation of QIGA Probability Manipulation

From Figure 5, it can be seen that the maximum profit obtained has a quadratic relationship with the probability amplitudes. This is likely due to the fact that in any quantum system, the square of the probability amplitudes are capped at a certain value, 1. Therefore, it makes sense that the evolution of maximum profit values would also be capped at a certain value. So, when taking this into consideration and observing the regression above, the researcher's determined that there was a maximizer for the manipulation function and it was verified that the maximum value for would be around 0.9 and this value was used in the rest of the research.

Now that the optimal probability amplitude was determined, the generation to which to implement the disaster algorithm had to be tuned. Depending on the generation, the defender would either have too little time to implement the new strategy (and two little time for the disaster algorithm to work), or too little time to manipulate the probability amplitudes. In the finalized algorithm, the generation for which to apply the disaster algorithm was tuned, using steps of 5, 6 and 7. To determine whether the disaster algorithm achieved a statistically significant change in the optimization of the knapsack problem, an analysis of variance test, or an ANOVA test, was run. Again, like with the manipulation of probability amplitudes, the significance level of the ANOVA test was 0.10 and that represented the alpha power of this statistical test. When implementing the probability amplitude manipulation, the goal was to simply increase the optimization of the knapsack problem. So, the alpha power of the test didn't need to be extremely high so a value of 0.1 was chosen. Since the p-value of .848064 is greater than the significance level of 0.1, the null hypothesis couldn't be rejected. If the null hypothesis was true, as the proposed algorithm didn't perform any better with the disaster algorithm, there would be an 84.8% chance of obtaining a profit value this high or higher based upon random sampling variability alone. Therefore, there isn't enough statistical evidence, within the power of this ANOVA test, to suggest that the

disaster algorithm increased the optimization of the knapsack problem. However, in Table 1, it can be observed that a disaster step of 6 does increase the optimization of the knapsack problem by around 3.22%. Although this isn't a statistically significant change, there is some improvement, which is all that matters. The disaster algorithm will help when the QIGA reaches a local maxima, which might not occur in every single trial. So, with any improvement in the knapsack optimization, the disaster algorithm is considered successful. Another way to determine the success of the disaster algorithm is to look at standard error. Also, when looking at Table 1, it can be observed that the standard error with the disaster algorithm is much lower than the standard error without the disaster algorithm. This suggests that the disaster algorithm is preventing the QIGA algorithm from getting caught on those lower local maxima and is successfully increasing the optimization of the knapsack problem. Before, it was up to chance whether the algorithm remained at the lower local maxima or whether it found a non-obvious solution. Now, the probability of finding that non-obvious solution increases drastically.

Application of QIGA to Cyber Security

Now that a working QIGA algorithm has been developed, that algorithm can be applied to the cyber security investment model proposed by Fielder et al. Although the data is still being collected regarding the cyber security application, once the data has been collected a 1-sample t-test will be run comparing the collected data to that of existing case studies. When looking at case studies, most of them only post the highest mitigation of a small-medium enterprise in thwarting in attack. So, the 1-sample t-test will be run comparing the data that was collected to the highest mitigation of previous case studies. In this t-test the significance level was 0.05 and that represented the alpha power of the statistical test. If the p-value turns out to be less than 0.05, then the null hypothesis can be reject. In this case, the null hypothesis states that the QIGA algorithm didn't achieve higher mitigation than that of existing case studies. By rejecting the null and confirming the statistical significance of the QIGA algorithm, an effective investment plan will be proposed to the company and the small-medium enterprise will be better protected against an attack.

5 Conclusion

The first part of this research was concentrated on developing an effective algorithm to optimize the knapsack problem. However, beyond that, I have applied that algorithm to the real-world scenario of cybersecurity investment. From this research, an effective QIGA algorithm was developed on the IBM Q32 quantum simulator that increases the optimization of the knapsack problem. Due to the success of this algorithm in optimizing the knapsack problem, the improved QIGA algorithm can be used for multiple practical applications. Since the Knapsack problem is a simple NP-Complete problem, the proposed QIGA algorithm should be able to optimize the solution to other NP-Complete problems through the process of NP-Complete reduction. Another major application for the QIGA-2 algorithm is towards system design, which is a knapsack

problem that can have major impact on the world as a whole. Essentially, system design process translates the customers' needs into a buildable system design that selects subsystems from an allowable set while remaining under a corporate budget. Chapman et al. in an analysis of the system design process explain how system design problem is NP-complete by reduction from the Knapsack problem (Chapman, Rozenblit Bahill, 2001). Both of these problems are attempting to maximize a certain value, either the required system performance or the amount of money in the knapsack, while remaining under a certain restraint, including a weight cap and a certain budget. Essentially, by utilizing the concept of NP-complete reduction, the knapsack problem is a useful tool to solve a variety of problems plaguing the real-world. With this proposed algorithm, and the future modifications to this algorithm, a significantly better solution to the knapsack problem could be found and this solution could be used to solve a variety of important computer science problems in the future. Computer science is an ever-evolving field of study and finding the optimal solution to these fundamental problems, such as the knapsack problem, could be extremely beneficial especially as technology continues to grow. With the emergence of new technologies, most prominently in the field of robotics, solutions to these old problems could have significant impacts on the field.

Apart from the actual optimization of the knapsack problem, an accurate model for cyber security investment was proposed. Using the proposed QIGA algorithm, an optimal investment plan will be provided to the CISO so that his or her small-medium enterprise will be protected against the malicious attacks implemented by a hacker. The known vulnerabilities in the system will be handled and the company can perform more research and development without worrying that their database will be compromised. In this online network that society has woven in the modern world, the importance of keeping information safe cannot be emphasized enough. If a company has a bad security system implemented and their data is vulnerable to hackers from all around the world, the government and other business contractors will not want to do business. The reputation of these small-medium enterprises relies on the idea that their backend is secure enough so that the confidential data which will be entrusted to them doesn't get lost. With this cyber security investment plan, small-medium enterprises will have the means of strengthening their backend and increasing their reputation within society. In a modern world where everyone's personal information is vulnerable, it is important for companies to invest in protecting that information and helping every person in the American society.

References

- [1] 2018 Deloitte NAISCO Cybersecurity Study (2014) www.nascio.org/Portals/0/Publications/Documents/2018/2018DeloitteNASCIOCybersecurityStudyfinal.pdf
- [2] Birkett, A. (2015). How to Get Over Local Maximum? Everyone Hits It Eventually. CXL. Retrieved 16 October 2019, from <https://conversionxl.com/blog/local-maximum/>
- [3] Bossaerts, P., Murawski, C. (2016). How Humans Solve Complex Problems: The Case of the Knapsack Problem. *Nature Partner Journals*. doi: 10.1038/srep34851
- [4] Chapman, W. L., Rozenblit, J., Bahill, A. T. (2001). System design is an NP-complete problem. *Systems Engineering*, 4(3), 222-229.
- [5] Fielder, A., Panaousis, E., Malacaria, P., Hankin, C., Smeraldi, F. (2016). Decision support approaches for cyber security investment. *Decision Support Systems*, 86, 13-23.
- [6] Forrester, R., (2017, Jan 20). The Quantum Measurement Problem: Collapse of the Wave Function Explained. Social Science Research Network. <http://dx.doi.org/10.2139/ssrn.2901820>
- [7] Fu, C., Liu, J., Wang, H., Zhi J. (2013, Mar 9). The Improvement of Quantum Genetic Algorithm and Its Application on Function Optimization. *Hindawi*, 2013. dx.doi.org/10.1155/2013/730749
- [8] Garg A., Goyal S. (2017). Vector Sparse Representation of Color Image Using Quaternion Matrix Analysis Based on Genetic Algorithm. *Imperial Journal of Interdisciplinary Research*, 3(7). www.imperialjournals.com/index.php/IJIR/article/view/5429
- [9] IBM Research Editorial Staff. (2018). An Open High-Performance Simulator for Quantum Circuits. IBM Research Blog. Retrieved 5 June 2019, from <https://www.ibm.com/blogs/research/2018/05/quantum-circuits/>
- [10] Kucharski, J., Nowotniak, R. (2012). GPU-based tuning of Quantum-Inspired Genetic Algorithm for a Combinatorial Optimization Problem. *Bulletin of the Polish Academy of Science*, 3(7). doi: 10.2478/v10175-012-0043-4
- [11] Kucharski, J., Nowotniak, R. (2014, Sept). Higher-Order Quantum-Inspired Genetic Algorithms. *IEEE*. doi: 10.15439/2014F99
- [12] Mia, M. A. H., Rahman, M. A., Uddin, M. (2007). E-Banking: Evolution, Status and Prospect. *The Cost and Management*, 35(1).
- [13] Montanaro, A. (2016). Quantum Algorithms: An Overview. *Nature Partner Journals*. doi: 10.1038/npjqi.2015.23
- [14] Neves, J., Miguel, R. (1999). Preventing Premature Convergence to Local Optima

in Genetic Algorithms via Random Offspring Generation. Lecture Notes in Computer Science, 1611, 127-136. https://doi.org/10.1007/978-3-540-48765-4_16

[15] Pan, Xiaohui Zhang, Tao. (2018). Comparison and Analysis of Algorithms for the 0/1 Knapsack Problem. Journal of Physics: Conference Series. 1069. 012024. [10.1088/1742-6596/1069/1/012024](https://doi.org/10.1088/1742-6596/1069/1/012024).

[16] Rachwald, R. (2008). Is banking online safer than banking on the corner? Computer Fraud Security, 2008(3), 11–12. doi:10.1016/s1361-3723(08)70045-9

[17] Rakes, T. R., Deane, J. K., Paul Rees, L. (2012). IT security planning under uncertainty for high-impact events. Omega, 40(1), 79–88. doi:10.1016/j.omega.2011.03.008

[18] The Copenhagen and Many Worlds Interpretations of Quantum Mechanics. (2019). Afriedman.org. Retrieved 16 October 2019, from <http://afriedman.org/AndysWebPage/BSJ/CopenhagenManyWorlds.html>