# Equity Trading Server Allocation Using Chance Constrained Programming

Yi Sun*
*California State University, San Marcos, San Marcos, California, USA*

Khadijeh Hassanlou
*California State University, San Marcos, San Marcos, California, USA*

---

Trading firms have increasingly adopted the practice of colocation to equity exchange centers for faster data access and quicker transactions. Due to the fact that network latency, which is largely determined by physical distance, has begun to play a more important role in determining service response time. However, this may not be the optimal solution when firms trade in multiple exchange centers simultaneously, especially for arbitrage opportunity. Also, there are other factors that firms need to consider to determine server location and setup. In this study, a two-phased approach is recommended for assessing and optimizing the location-allocation problem driven from trading with multiple exchange centers. First, Analytic Hierarchy Process and the Brown-Gibson methods were adapted to assess server location and service qualities. Next, a stochastic Knapsack optimization model was proposed and solved for a numerical example using Genetic Algorithm.

---

* Corresponding Author. E-mail address: ysun@csusm.edu

## I. INTRODUCTION

Due to the move to high-speed internet communication and tremendous increases in computing power, network latency, which is mostly determined by physical distance, has begun to play a more important role in determining service response time (Johansson, 2000). Hence, the physical location of an application server will impact the performance of the services it hosts. As a result, in recent years, online trading firms have started renting servers near equity exchange centers so they can get access to trading statistics on their computers faster than competing investors. This practice is called colocation, and has been widely used to enable firms to execute high-frequency trading (HFT), which constitutes almost 75% of all buying and selling of US Equities (Buchanan, 2011). The physical proximity to the exchange center reduces the time from when a firm's buy or sell order is entered and when it's executed by a few milliseconds. With reduced network latency, their proprietary algorithms installed on these servers will be able to interpret the data and execute transactions with a speed advantage over competitors.

If a firm is to locate its application server(s) near one exchange center, that server will be far away from other centers when those centers are geographically distant from each other. This location choice creates a problem if the firm is trading simultaneously with multiple exchange centers in order to take advantage of arbitrage opportunities. Therefore, depending on trading practice and patterns, the server location may not necessarily be located near one center for optimal operations. Instead, the server should be geographically deployed in a location between those centers in which it trades (Wissner and Freer, 2010).

Still, apart from a physical distance or network latency, there are other criteria the firm needs to consider in selecting an optimal server location to host its trading applications. These criteria may include, among others, trading partners, trading patterns, network bandwidth, server facility cost, server cost and speed, system reliability, server security, local taxation, power usage and environmental impact. Currently, there is not much research devoted to equity trading server allocation problems when these factors are considered.

In this study, we will propose an assessment model based on Saaty's Analytic Hierarchy Process (Saaty, 1980, 1982) and the Brown-Gibson method (Brown and Gibson, 1972) for evaluating multiple criteria for selecting server locations. We will then create a linear programming model considering stochastic process parameters to help investment firms choose optimal server locations-allocations based upon multiple criteria and constraints including usage, cost, and capacity. The developed stochastic knapsack model will be reformulated using Chance Constrained Programming (Liu, 2009; Kosuch and Lisser, 2010). The model then will be solved with MATLAB R2017b Genetic Algorithm solver.

## II. LITERATURE REVIEW AND RATIONALS

High-frequency trading refers to trading that leverages powerful computers, sophisticated algorithms, and access to high-frequency financial data to conduct large amounts of orders at very fast speeds. Typically, traders with the fastest execution speeds are at an advantage over traders with slower execution speeds (Lewis 2014). A high-frequency trading algorithm can process trade orders in under 400 microseconds on average. To profit, HFT firms need to compete for the smallest speed advantages (Aswani, 2016).

The speed advantage depends partially on data and algorithm processing speed. On a distributed platform, sources of delay may include transmitting time, queuing delay, and network latency (Johansson, 2000). Server processing speed, transmit time and queuing delay have all been improving over time, thanks to the implementation of high-speed internet communications with broadband connections, the improvement in network equipment power and general computing powers. In comparison, network latency has not improved. Physical distance provides an effective lower bound in measuring network latency. As a result, being able to locate the servers that host HFT algorithms closer to a trading exchange center will help improve the speed of execution. This practice is known as colocation (Gary Shorter and Rena S. Miller 2014).

However, the business model of some HFT firms requires them to analyze multiple market conditions in order to make trading decisions. It has been argued that if an HFT firm engages in trading simultaneously in multiple exchanges in order to take advantage of arbitrage opportunities, then the server should be geographically deployed in a location between those centers it trades in (Wissner and Freer, 2010). For example, if a firm plans to purchase multiple shares of the same stock, it may do so across multiple stock exchanges in order to get the best possible price across the board. However, a purchase of stock on the Nasdaq may trigger a price increase of the same stock listed in another market such as the London Stock Exchange. However, there could only be a fraction of a second for the increase in the London Stock Exchange to take place after the purchase on Nasdaq. The arbitrage opportunity may only exist within a couple of milliseconds. A high-frequency trading firm using complex computer algorithms to perform its trading, when strategically located, may be able to complete the purchase before the window of opportunity closes.

When a firm selects a server or multiple servers to host its HFT algorithms, many criteria need to be considered. First, understanding the trading pattern of the firm is crucial for selection strategies. If a firm trades only in one center, then network latency will be just between the firm's server and the exchange center. The firm will be most likely to choose just one location near the exchange center. If the firm trades in multiple centers, regardless of whether the trades are dependent on each other or not (one trade in an exchange center will affect the trading decision in another center), it may choose one or multiple server locations to reduce overall network latencies, which includes all latencies of all related trade generated from the server(s) (Wissner and Freer, 2010).

Second, service performance also has a huge impact on the trading operation and profit margin. High-performance servers are required to process vast amounts of data and execute complicated trading algorithms. Service performance has multiple dimensions that include computing speed, availability, security, speed, information integrity, and so forth (Greenberg, A. Hamilton, J. Maltz, D. A. and Patel, P. 2009; Sun, He and Leu, 2007). Servers with better performance will generally cost more. Therefore, server selection affects the overall cost, including fixed cost and variable cost, and needs to be evaluated carefully.

We believe that we are among the first to study the optimal service location problems for equity trading servers with factors such as network latency, server performance and costs included. This study proposes a model to help HFT firms locate their servers optimally based on their requirement and trading patterns. A two-phased approach is recommended for assessing and optimizing the location-allocation problem driven from trading with multiple exchange centers. First, Analytic Hierarchy Process and the Brown-Gibson methods were adapted to assess server location and service quality. Next, a stochastic Knapsack optimization model was proposed and solved for a numerical example using Genetic Algorithm.

The AHP is a multi-criteria decision-making methodology particularly suitable for the situations where due regard for individual beliefs is critical. This methodology uses pair-wise comparisons and eigenvector to prioritize alternatives. Its theory and the underlining axioms were introduced by Saaty [Saaty 1980, 1982] and further developed by many other scholars [Arbel and Oren 1086, Harker and Miller 1990]. Since its introduction, the AHP has been applied to a wide spectrum of real-world problems. Shim (1989), and Vargas (1990) provided a comprehensive collection of industry and public-sector applications.

One of the AHP's criticisms is its pair-wise comparisons. These comparisons are labor intensive and could be unreliable under some circumstances. Therefore, there is an interest in incorporating other methodologies into the AHP to address potential problems. For some applications, a combination of AHP and the BG methods provides a better methodology for a firm to assess quality. The Brown-Gibson model uses actual costs to derive the relative merits of alternatives along criteria that can be measured objectively. It only relies on human judgments for intangible criteria. For a discussion of how AHP and Brown-Gibson work together and their applications, refer to Sun, He and Leu (2007) research.

As mentioned before, we will use a knapsack problem, as a widely studied NP-hard combinatorial optimization problem, for formulating the allocation model in the second phase. This classical problem deals with choosing a subset of items out of a given item, each with a weight and a reward. The selected subset cannot exceed a given limit such as the capacity of the knapsack, and total rewards proceeds from the subset maximized.

In real-world knapsack decisions, it is often the case that the reward parameters or the weight values (or both) are not deterministic and known exactly, so we are confronted with uncertainty in these kinds of problems. It is essential to address this uncertainty and detect it in our decisions. The way in this case would be stochastic programming.

Chance Constrained Programming (CCP) as an appropriate and approved method for these kinds of problems is proposed for dealing with uncertainty embedded in the developed model. Chance Constrained Programming as a type of stochastic programming, developed by Charnes and Cooper (1959), is a competitive method for solving optimization problems over uncertain constraints. The constraints and objective function, which contain random variables, are guaranteed to be satisfied and optimized with a certainly predetermined probability. CCP can be applied in various types of stochastic optimization problems, so there are numerous studies in this area. We will mention some related applications of CCP in solving stochastic knapsack problems.

Kosuch and Lisser (2010) studied and solved the two main types of stochastic knapsack problems with random weight parameters: the stochastic knapsack problem with probabilistic constraints and the problem with simple resources. Ilhan et al. (2011) considered an adaptive knapsack problem with deterministic weights and random normally distributed rewards. The model tries to maximize the probability of reaching the target reward level. The developed model is used for the resource allocation problem.

Cheng et al. (2014) developed a new version of the stochastic knapsack problem considering robust distribution for some parameters. They considered the classical knapsack problem where a set of constraints were satisfied with a particular probability and developed a robust knapsack problem that was reformulated as a semidefinite program (SDP).

Liu et al. (2016) proposed a methodology to solve the 0-1 multidimensional knapsack problem considering the Brownian motion for parameters variation. The multidimensional knapsack problem is known as NP-hard regarding computational complexity. The proposed method includes two main steps: generating a discrete solution and producing a feasible solution. The proposed method is comparable with other metaheuristics.

Range et al. (2018) considered the knapsack problem with stochastic weights. This stochastic knapsack problem (SKP) is formulated considering a probabilistic capacity constraint (CKP) and the SKP with simple recourse (SRKP). The formulated model as a network problem was solved using a Dynamic Programming approach for finding the shortest path.

In this research, we will develop a stochastic chance constrained fractional knapsack program with random weights and also random rewards, considering probabilistic and deterministic constraints together for optimizing the location-allocation of multiple such servers that their service quality is evaluated before using AHP/GP approach.

## III. DECISION MODEL

In this research two main stages are pursued: the first step includes finding the importance of different quality dimensions and also evaluating Quality of Service (QoS) for each provider based on predetermined weighted quality dimensions using AHP and GB methodologies. The second step is to concentrate on optimizing the allocation problem using the constrained fractional knapsack problem.

### 2.1. Quality Assessment using AHP/GB

In the first phase, we propose a combination of models for the firm to properly assess the quality of services (QoS) offered by high-frequency trading servers providers.

The HFT firm first develops a set of quality attributes for evaluating available servers and related services. Based on the evaluations of a group of experts, the AHP/BG method is used to derive the quality index of each service provider. The model also allows the firm to provide preferences or weights on a set of corresponding quality criteria to indicate their relative importance. The AHP approach can be applied to derive these weights. We then incorporate the composite indices into a mathematical model for optimal solutions of server location.

We used AHP approach to evaluate the relative merit of alternatives of qualitative/subjective measures which reflect a user's personal opinions or attitudes. The qualitative criteria include cybersecurity, physical security, information integrity/accuracy, server reliability, environmental impact, support from local regulation and policy, network latency, and customer service level. These criteria are important for selecting the right servers for HFT. We used a 1-9 ratio scheme for pair-wise comparisons. More specifically, a score of 1 means equally preferred, 3 moderately preferred, 5 strongly preferred, 7 very strongly preferred, and nine extremely preferred. We then calculated values of the eigenvector of this comparison matrix, normalized to between 0 and 1, and representing the relative merits of alternatives regarding an evaluation criterion.

When a firm evaluates criteria, it may have quality preferences or weights on a set of corresponding quality criteria to indicate their relative importance. In this AHP approach, we applied these weights by calculating the dot products of the user's preferences and the quality ratings of each server locations capable of fulfilling the request.

One of the criticisms of AHP is the laboriousness of the pair-wise comparisons. The AHP approach requires ${}^{1}/{}_{2}\, rn(n-2)$ comparisons, where r is the number of evaluation criteria and n is the number of alternatives. In a more dynamic situation, these comparisons must be reevaluated frequently. It is conceivable that a firm may consider switching between server locations often in order to accommodate market changes.

One can argue that the characteristics such as security, information integrity/accuracy, and speed are closely related to capital investment; therefore, they do not frequently change since capital investment is likely to be fixed for a longer period. On the contrary, cost does not share the same stable characteristics; it is likely instead to become a key competitive strategy exploited by price leaders and followers alike as the industry matures. In this situation, constantly conducting pair-wise comparisons over a large number of alternatives would become drudgery, if possible at all. Also, the cost can be measured in a more precise term than the AHP's notion of degree of preferences. Therefore, we integrate the AHP approach with the Brown-Gibson method (BG) so that it, in our opinion, works better in a volatile environment and provides an improved evaluation of cost-related criteria that can be measured in the monetary term. The two quantitative/objective criteria are fixed cost and variable cost.

The BG method classifies evaluation criteria into two categories: objective $O_i, (0 \leq O_i \leq 1)$ and subjective factors $S_i, (0 \leq S_i \leq 1)$, depending on whether or not they can be measured by the monetary term. Objective measures are derived by the AHP approach.

The BG method combines subjective and objective evaluations into a composite index, which is expressed as the linear combination of both indices and is defined as: $L_i = pO_i + (1-p)S_i$, where $p, (0 \leq p \leq 1)$ represents the relative emphasis on the objective evaluations (Maurino and Luxhøj, 2002). $L_i$'s, thus, are line segments between 0 and 1, and a decision frontier can be constructed based on these line segments.

The combination of AHP and the BG methods provides a methodology for assessing the quality of a server location. Regarding optimization, however, the methodology is myopic and greedy. It merely considers the best solution on a per usage basis, not the entire solution space in a planning horizon. Furthermore, it does not consider the budget implication for repetitive service requests. For example, an e-business may rely on Web Services such as credit checking, delivery tracking, and zip-code verification for every transaction. In such an environment, demand, budget, and other constraints may become critical. However, the AHP and the BG methods cannot accommodate any of these constraints. Therefore, we propose the following model that takes into account volume requests with a budget constraint.

## 2.2. Optimization Model

In the second phase, we then incorporate the composite indices derived from the first phase into a mathematical model for optimal solutions of server allocation.

## 2.3. Proposed Model Formulation

For convenience, we assume that there are M providers that offer high-frequency trading servers and related services. Also, we assume that the QoS of a provider is measured along r quality dimensions that may include network latency to each exchange center, security, speed, information integrity, cost, and so forth. The trading firm, using the AHP and BG approach, assesses QoS of providers and stores the evaluation results in a matrix whose entries are denoted as $q_{mr}$, where $0 \leq q_{mr} \leq 1$ with 1 being the highest evaluation and $\sum_m q_{mr} = 1, \forall r$. Among other things, $q_{mr}$ will include the network latencies incurred if provided m is selected. We further assume that a trading request is also accompanied by a set of QoS preferences, expressed as $w_r$, where $0 \leq w_r \leq 1$ with 1 being the highest preference and $\sum w_r = 1$. First, the deterministic version of the model is formulated as:

P(1): Maximize: $\sum_r w_r (\sum_m u_m q_{mr})$ ------(1)

Subject to:

$\sum_m u_m \geq d$ ---------------------------- (2)

$\sum_m u_m C_m + \sum_m FC_m g_m \leq T$ -------- (3)

$u_m \geq T_m g_m \ , \forall m$ -------------------- (4)

$u_m \leq \lambda g_m \ , \forall m$ ------------------------ (5)

$u_m \geq 0 \ \& \ g_m \in \{0,1\} \ , \forall m$

Table 1 summarizes the parameters and variables used in our model.

**Table1. Summary of Notation**

| | |
|---|---|
| m: | Number of providers. |
| r: | Number of criteria (quality dimensions). |
| $q_{mr}$: | QoS of provider m in quality dimension r. |
| wr: | Weight or importance of each quality criteria. |
| Sm: | Score of overall quality for Provider m. |
| d: | Total trading request. |
| TB: | Total budget. |
| $T_m$: | Minimum transaction that provider m will fulfil if it was selected. |
| $C_m$: | Variable Cost of using service offered by provider m. |
| $FC_m$: | Fixed cost of using service offered by provider m. |
| λ: | Arbitrarily large number. |
| **Decision Variables:** | |
| $u_m$: | Amount of service that is fulfilled by provider m. |
| $g_m$: | Binary variable (is 1 when provider m is selected and is 0 otherwise). |

The objective function is to maximize the overall quality level with the consideration of user's preferences. Constraint (2) is to ensure that all service requests are met; constraint (3) is to uphold the budget requirement, ensures that the required variable transaction cost and fixed setup cost doesn't exceed the total budget; and constraint (4) is to make sure that providers offering the services only fulfill requests and finally constraint (5) refers to the binary variable that takes 1 when provider m is selected and 0 otherwise.

For summarizing the objective function and reducing the parameters, using the Saati(1980) approach in decision making, the formula of the objective function can be aggregated as:

$$\text{Maximize: } \sum_m S_m u_m \text{ ------------- (6)}$$

Where:

$$\mathbf{S} = \mathbf{Q}_{[m\times r]} \cdot \mathbf{W}^{\mathbf{T}}_{[r\times 1]} \text{ ---------------- (7)}$$

**Q** and **W** respectively represent the $q_{mr}$ and $w_r$ in the matrix format.

The formulation is a constrained fractional knapsack problem and is strongly NP-hard.

In real-life problems, we often have to deal with uncertainty. Parameters cannot be predicted exactly but rather estimated probabilistically. Therefore, it is sometimes more desirable to model these parameters with random variables. In practice, it is often the case that at the time of making the knapsack decision either the reward parameters or the weights parameter (or both) are not exactly known.

In this research, there are some uncertain parameters that affect the whole model because both objective function and constraints need to deal with probabilistic parameters and address the amount of uncertainty that problem is suffered with. Since the objective function coefficients (Sm) resulted from AHP based on pairwise comparison matrix, they cannot be deterministic. The cost of service for each provider ($C_m$) is also considered as a stochastic parameter, and because of the inherent characteristic of demand in all kind of problems, d is stochastic too. Chance Constrained Programming as an appropriate and approved method for these kinds of problems is proposed for dealing with uncertainty embedded in the developed model.

## IV. CHANCE CONSTRAINED PROGRAMMING

In CCP, the objective function should be achieved with the stochastic constraints held at least α of time, where α is provided as an appropriate safety margin by the decision maker (Hassanlou, 2017).

Assume that x is a decision vector, ξ is a stochastic vector, and gj(x, ξ) are stochastic constraint functions, j= 1, 2… p. Since the stochastic constraints gj(x, ξ) ≤ 0, j= 1, 2… p does not define a deterministic feasible set, they need to be held with a confidence level α. Thus chance constraint is represented as follows (Liu, 2009):

$$\Pr\{ g_j(x, \xi) \le 0, j= 1, 2, \dots, p \} \ge \alpha \text{ ----- (8)}$$

Which is considered the same α for all stochastic constraints, and when we want to assume that they are different, it can be shown as follows:

$$\Pr\{ g_j(x, \xi) \le 0\} \ge \alpha_j, \; j= 1, 2, \dots, p \text{ ----- (9)}$$

**Theorem (1)**: Assume that the stochastic vector $\zeta = (a_1, a_2, \dots, a_n, b)$ and the function g(x, ξ) has the form g(x, ξ) $= a_1x_1 + a_2x_2 + \cdots + a_nx_n - b$. If $a_i$ and b are assumed to be independently normally distributed random variables, then Pr { g(x, ξ) ≤ 0} ≥ α if and only if

$$\sum_{i=1}^{n} E[a_i]x_i + \Phi^{-1}(\alpha)\sqrt{\sum_{i=1}^{n} Var[a_i]x_i^2 + V[b]} \le E[b];$$
(10)

Where Φ is the standardized normal distribution function. You can see proof of the above theorem in Liu (2009).

## Chance Constrained Knapsack Problem

In this paper, it is supposed that the Sm, Cm, and d, are stochastic and can be presented as following random variables: $\widetilde{S_m}$ , $\widetilde{C_m}$ and $\tilde{d}$ . Now the model can be reformulated as a Chance constrained Knapsack programming (CCKP). In the case mentioned above, the reward parameters (Sm) and also weights parameters (Cm) and right-hand side of the Knapsak problem are stochastic.

Stochastic parameters are considered as random variables and assume that they are independently normally distributed. Based on Kosuch and Lisser (2010) research, we can reformulate the P (1) model as:

P(2): Maximize: $E[\sum_m \tilde{S}_m u_m]$

Subject to:

$$\Pr\{ \textstyle\sum_m u_m \ge \tilde{d} \} \ge \alpha \text{ ------------------ (11)}$$

$$\Pr\{ \textstyle\sum_m u_m \tilde{C}_m + \sum_m FC_m g_m \le TB \} \ge \alpha$$
(12)

$$u_m \ge T_m g_m \;, \forall m \text{ ----------------------- (13)}$$
$$u_m \le \lambda g_m \;, \forall m \text{ ------------------------ (14)}$$
$$u_m \ge 0 \;\&\; g_m \in \{0,1\} \;, \forall m$$

Using Theorem (1), we can rewrite the stochastic constraints (11 and 12) in the form of equation (10) and the last version of CCKP problem can be formulated as:

P(3): Maximize: $E[\sum_m \tilde{S}_m u_m]$

Subject to:

$$-\textstyle\sum_m u_m + \Phi^{-1}(\alpha)\sqrt{Var(d)} \le E[d] \;\; (15)$$

$$\textstyle\sum_m E[C_m]u_m + \Phi^{-1}(\alpha)\sqrt{\sum_m Var[C_m]u_m^2} + \sum_m FC_m g_m \le TB \text{ ------------------------------------------ (16)}$$

$$u_m \ge T_m g_m \;, \forall m \text{ ----------------------- (17)}$$
$$u_m \le \lambda g_m \;, \forall m \text{ ------------------------- (18)}$$
$$u_m \ge 0 \;\&\; g_m \in \{0,1\} \;, \forall m$$

The proposed stochastic knapsack problem is a nonlinear NP-hard problem. In other words, optimal solutions can be obtained within a reasonable amount of time only for small-sized problems. However, problems of large size need heuristics and also metaheuristics that take advantage of the structures of the problem. In this research, Genetic Algorithm as a popular valid and appropriate metaheuristic method for solving the problem in NP-hard and NP-complete complexities level, is used for optimizing developed stochastic Knapsack Problem. GA is an evolutionary algorithm developed originally by Holland (1975). GA, based on the mechanism of genetics and natural selection, is capable of efficiently locating near optimal or even optimal solutions for many combinatorial optimization problems. The proposed CCKP model is solved using the MATLAB R2017b Genetic Algorithm solver.

## V. NUMERICAL EXAMPLE

As mentioned before, this research includes two main parts, first for evaluating the quality of service of providers using AHP/GB based on multiple criteria and the next for optimizing the allocation of demanded service to providers subjected to stochastic constraints. The following hypothetical numerical example is considered to demonstrate the agility and robustness of the proposed model.

### a) AHP/ GB

For the experiment, there will be eight server location providers and eight qualitative and two quantitative criteria. We repeat the experiment 30 times to generate mean and standard deviation. Table 2. And Table 3. respectively represent the result of AHP/GB method for evaluating the servers in term of subjective and objective factors.

As mentioned before in detail, the results for objective and subjective factors are combined linearly with a given p. The combined values are shown in Table 4.

### b) CCKP

In the hypothetical example, the following values are considered for the total budget, distribution of demand, distribution of cost, fixed setup cost and minimum required transaction for each provider:

$\tilde{d} \sim N\,(140\,,5)$

TB= 6000

**TABLE2. SUBJECTIVE FACTORS**

| Provider | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|---|---|---|---|---|---|---|---|---|
| E[$\tilde{S}_m$] | 0.112 | 0.135 | 0.127 | 0.124 | 0.123 | 0.127 | 0.128 | 0.125 |
| Var[$\tilde{S}_m$]] | 0.019 | 0.022 | 0.022 | 0.020 | 0.023 | 0.028 | 0.022 | 0.022 |

**TABLE 3. OBJECTIVE FACTORS**

| Provider | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|---|---|---|---|---|---|---|---|---|
| E[$\tilde{S}_m$] | 0.112 | 0.135 | 0.127 | 0.124 | 0.123 | 0.127 | 0.128 | 0.125 |
| Var[$\tilde{S}_m$]] | 0.019 | 0.022 | 0.022 | 0.020 | 0.023 | 0.028 | 0.022 | 0.022 |

**TABLE 4. COMBINED WITH P =.062**

| Provider | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|---|---|---|---|---|---|---|---|---|
| E[$\tilde{S}_m$] | 0.111 | 0.129 | 0.118 | 0.122 | 0.124 | 0.137 | 0.133 | 0.126 |
| Var[$\tilde{S}_m$] | 0.024 | 0.038 | 0.029 | 0.034 | 0.043 | 0.049 | 0.039 | 0.044 |

**TABLE 5. DISTRIBUTION OF COST, FIXED COST AND MINIMUM REQUIRED TRANSACTION**

| Provider | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|---|---|---|---|---|---|---|---|---|
| $E[\tilde{C}_m]$ | 20 | 25 | 21 | 23 | 24 | 28 | 26 | 25 |
| $Var[\tilde{C}_m]$ | 3 | 5 | 2 | 7 | 2 | 3 | 6 | 1 |
| $FC_m$ | 20 | 50 | 100 | 50 | 80 | 60 | 100 | 50 |
| $T_m$ | 5 | 25 | 5 | 10 | 15 | 5 | 20 | 5 |

**TABLE 6. RESULTS INCLUDING DECISION VARIABLES AND OBJECTIVE FUNCTION**

| m | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|---|---|---|---|---|---|---|---|---|
| $u_m$ | 17.82 | 0 | 26.37 | 0 | 25.84 | 17.57 | 0 | 52.39 |
| $g_m$ | 1 | 0 | 1 | 0 | 1 | 1 | 0 | 1 |
| ***Objective Function*** | | | | ***17,304*** | | | | |

The proposed stochastic fractional knapsack problem is solved using the MATLAB R2017b Genetic Algorithm solver for the supposed numerical example to find the best allocation of providers. The results including the decision variables and the optimized objective function for α=0.9 are represented in Table 6.

### c) Sensitivity Analysis

In this part, we want to assess the results produced with the model to show the validity and reliability of the results and also their sensitivity to some parameters. First, the result of the Genetic Algorithm solver is compared with the MATLAB nonlinear solver based on Hessian matrix calculations, which needs to be fed the starting point (u0). It Table 7 you can see the results for different sample starting points.

The results show that the nonlinear solver can produce the better objective function, but there are two main reasons that we chose the GA. First, an appropriate metaheuristic algorithm is preferred for such an NP-hard developed model and next, depending on starting point could be a weakness for the reliability of the nonlinear solver in this case because you can see in table 6 that the model doesn't work with zero u0.

**TABLE 7. COMPARING GA SOLVER WITH NONLINEAR SOLVER USING HESSIAN MATRIX WITH DIFFERENT U0**

| | **Starting point solution (u0)** | | | | | |
|---|---|---|---|---|---|---|
| Objective Function | [1 1 1 1 1 1 1 1] | [0 0 1 1 1 1 1 1] | [1 0 1 0 1 0 1 0] | [0 0 0 0 0 0 1 1] | [0 0 0 0 0 0 0 1] | [0 0 0 0 0 0 0 0] |
| Nonl. Solver | 17,413 | 17,609 | 17,990 | 17,561 | 17,417 | 0 |
| GA Solver | ***17,304*** | | | | | |

Tables 8. Represents the objective function sensitivity to different satisfaction levels for probabilistic constraints in the model. It is that obvious the larger probability of satisfaction means more limitation in feasible solution space and can lead to worsening the objective function value. Obviously, it can be concluded that if we consider deterministic model with 100 percent satisfaction for constraints the objective function will be worse that's the good evidence for good performance of developed CCKP in this research.

**TABLE 8. DIFFERENT SATISFACTION LEVEL (A) FOR PROBABILISTIC CONSTRAINTS**

| α | 0.50 | 0.60 | 0.90 | 0.95 | 0.99 |
|---|---|---|---|---|---|
| Objective Function | 22,864 | 21,003 | 17,304 | 17,002 | 16,869 |

Tables 9. Represents the objective function sensitivity to different values of the total budget. Results indicate that increasing the total budget doesn't necessarily lead to improvement in the objective function. In Table 9. There are some district values considered for sensitivity analysis, but the results indicate that the model is not so sensitive on budget level between 5000 to 15000 but decreasing total budget to less than 5000 could affect the objective function considerably. This result could be applied for decision maker to find indifference and sensitive intervals on budget level.

**TABLE 9. DIFFERENT LEVELS FOR TOTAL BUDGET WHEN THE A KEEPS CONSTANT EQUAL 0.90**

| TB | 4000 | 5000 | 6000 | 7000 | 10000 | 15000 |
|---|---|---|---|---|---|---|
| Objective Function | 9,699 | 17,304 | 17,304 | 17,325 | 17,325 | 16,727 |

## VI. CONCLUSIONS

Location and service quality of servers has become a prominent issue for HFT firms' business model. In particular, those who trade in multiple markets seeking arbitrage opportunities are sensitive to the server locations about exchange centers. In this paper, we proposed a two-phased approach to help firms select server locations. First, Analytic Hierarchy Process and the Brown-Gibson methods were adapted to assess server location and service qualities. Next, a stochastic Knapsack optimization model was proposed for optimizing the allocation of required service to multiple servers. To address the uncertain parameters embedded in the model, we reformulated it using chance a constrained knapsack program which requires a heuristic algorithm to be developed to obtain optimal or near-optimal solutions. The proposed CCKP was solved for a numerical example using Genetic Algorithm.

The application of the AHP and the BG methods made it possible to evaluate the QoS of high-frequency trading servers and match a customer's quality preferences. MATLAB R2017b GA solver is used for solving the NP-hard developed model to find the best location-allocation of multiple servers. Robustness of results is tested using sensitivity analysis on some parameters and also comparing the GA solver results with nonlinear solver result.

Since this is a preliminary study of the selecting of HFT servers based on quality criteria, further development and refinement of the modeling and solution methodologies is required. While being a mathematically proven methodology, AHP entails a time-consuming evaluation process, and the whole evaluation process must take place again whenever new services and service providers are added. We will explore other multi-criteria decision-making methodologies to make QoS assessments less labor-intensive and more effective. Also, the study can certainly benefit from using real-life data to validate the model. Empirical research can help establish with what quality dimensions trading firms will be most concerned. It can also help with better understanding of the qualities of the available services.

Finally, it is needed to mention that the authors used the developed CCKP for trading server allocation problem, but it is not limited to this application, can be easily adjusted for various demanding applications such as cloud computing capacity management, Wireless Sensor Networks (WSN's) location allocation planning, Supply Chain Management, networks caches planning and etc.

## REFERENCES

Arbel, A. and Oren, S.S., "Generating Search Directions in Multiobjective Linear Programming Using The Hierarchy Process", Socioeconomic Planning Sciences, 20, 1986, 369–373.

Aswani, S., "Leveraging Co-location for Competitive Advantage in HFT, By Senior Manager", Solutions Marketing, Hewlett Packard Enterprise, May 31, 2016, https://www.hpcwire.com/solution_content/hpe/financial-services/leveraging-co-location-competitive-advantage-hft/ (accessed June 23, 2018).

Brown, P.A. and Gibson, D. F., "A Quantified Model for Facility Site Selection: An Application To A Multi-Facility Location Problem", AIIE Transactions, Vol. 4, 1972, 1–10.

Buchanan, M., "With High-Speed Trading, the Market Cannot Hold, Bloomberg.com", October 4, 2011, http://www.bloomberg.com/news/2011-10-05/with-high-speed-trading-market-cannot-hold-commentary-by-mark-buchanan.html (accessed June 23, 2018).

Charnes, A. and Cooper, W.W., "Chance-constrained Programming", Management Science, 6, 1959, 73–79.

Cheng J., Delage E., Lisser A., "Distributionally Robust Stochastic Knapsack Problem", Society for Industrial and Applied Mathematics, Vol. 24, No. 3, 2014, 1485–1506.

Greenberg, A. Hamilton, J. Maltz, D. A. and Patel, P., "The Cost of a Cloud: Research Problems in Data Center Networks", ACM SIGCOMM Computer Communication Review, 39(1), 2009, 68-73.

Harker P.T. and Miller, I., "Global Effective Questioning in the Analytic Hierarchy Process", European Journal of Operational Research, 48, 1990, 88–97.

Hassanlou Kh., "A Multi Period Portfolio Selection Using Chance Constrained Programming", Decision Science Letter, 6, 2016, 221–232.

Ilhan T., Iravani M., Daskin M., The Adaptive Knapsack Problem with Stochastic Rewards, Operations Research Volume 59 (1), 2011, 242-248.

Johansson, J. M. "On the Impact of Network Latency on Distributed Systems Design", Information Technology Management, (1), 2000, 183-194.

Kosuch, S., Lisser, A., "Upper Bounds for The 0-1 Stochastic Knapsack Problem and A B&B Algorithm", Annals of Operations Research, 176(1), 2010, 77–93.

Lewis, M., Flash boys: A Wall Street Revolt. W. W. Norton, 2014.

Liu, B., Theory and Practice of Uncertain Programming, 3rd ed., UTLAB, 2009.

Liu J., Wu Ch., Cao J., Wang X., Kok L., "A Binary Differential Search Algorithm for the 0–1 Multidimensional Knapsack Problem", Applied Mathematical Modelling, 40 (23–24), 2016, 9788-9805.

M. A. Maurino, J. T. Luxhøj, Analysis of a Group Decision Support System (GDSS) for Aviation Safety Risk Evaluation, Rutgers University, New Brunswick, NJ, 2002.

Range T., Kozlowski D., Petersen N., "A Shortest-Path-Based Approach for The Stochastic Knapsack Problem with Non-Decreasing Expected Overfilling Costs", Computers & Operations Research, 97, 2018, 111-124.

Saaty, T.L. The Analytic Hierarchy Process, New York: McGraw Hill, Pittsburgh: RWS Publications, 1980.

Saaty, T.L. Decision Making for Leaders; The Analytical Hierarchy Process for Decisions in a Complex World, Pittsburgh: RWS Publications, 1982.

Saaty, T.L., "Axiomatic Foundation of the Analytic Hierarchy Process", Management Science, 32, 1986, 841-855.

Shim, J.P., "Bibliographical research on the Analytic Hierarchy process", Socio-economic Planning Sciences, 23, 1989, 161–167.

Shorter, G. and Miller, R.S., "High-Frequency Trading: Background, Concerns, and Regulatory Developments", Congressional Research Service, June 19, 2014, https://fas.org/sgp/crs/misc/R43608.pdf (accessed June 23, 2018).

Sun, Y. He, S.Y. and Leu, J. Y., "Syndicating Web Services: A QoS and User-driven Approach", Decision Support Systems, 43(1), 2007, 243–255.

Vargas, L.G., "An Overview of The Analytic Hierarchy Process", European Journal of Operational Research, 48, 1990, 2–8.

Wissner-Gross, A. D., and Freer, C. E. "Relativistic Statistical Arbitrage." Physical Review E 82.5 2010, The American Physical Society.