A Custom Monkey Algorithm hybridized with GIS for location-allocation of COVID Centers

Manar Abu Talib
Dept. of Computer Science
University of Sharjah
Sharjah, United Arab Emirates
mtalib@sharjah.ac.ae

Sohail Abbas
Dept. of Computer Science
University of Sharjah
Sharjah, United Arab Emirates
sabbas@sharjah.ac.ae

Qassim Nasir
Dept. of Computer Engineering
University of Sharjah
Sharjah, United Arab Emirates
nasir@sharjah.ac.ae

Essa Basaeed
Dubai Police
Dubai, United Arab Emirates
ebasaeed@dubaipolice.gov.ae

Mohammed Al-Haidary
Dept. of Computer Engineering
University of Sharjah
Sharjah, United Arab Emirates
mjawad@sharjah.ac.ae

Abstract— **The unfortunate rise of a pandemic calls for an urgent deployment of medical virus testing and vaccination centers at strategically selected locations. Inefficient deployment causes a failure to optimally distribute resources within a country or city, which might result in the loss of lives due to a lack of medical availability in a certain region. In this paper, a custom discretizing Monkey algorithm (MA) is proposed and compared with an enhanced Genetic Algorithm (GA) to solve the single objective of a Capacitated Maximal Covering Location Problem (CMCLP) for COVID centers in the emirate of Dubai, United Arab Emirates. Five sites and 256 existing facilities will be selected. Based on the results, the MA has proved to be more performant and consistent than the genetic algorithm by achieving a 5.16% higher average fitness and a 21.03% run duration penalty compared to GA.**

Keywords— monkey algorithm, genetic algorithm, capacitated maximal covering location problem, ArcGIS, geoprocessing.

I. INTRODUCTION

Facility location-allocation is a decision-making strategy used to select the most appropriate location(s) for a given facility type and to allocate potential demand to that facility. It is used to minimize costs and maximize facility utilization when the size of the geospatial search domain is huge [1]. Location cost encompasses all the constraints the facility must satisfy at a given coordinate set to be well localized. Examples of location costs include population coverage, facility startup capital, and travel timing constraints, to name just a few. Zhong et al. [2] compute their system cost as the total of the following: the setup cost to open a distribution center, the fixed cost of the vehicles, the vehicle travel cost, and penalties for both lack of supply and oversupply. Another illustration of location costs is by Yu [3], who incorporates every customer's demand, their attraction to each facility, the distance between each customer and facility, and the opening cost of a potentially new facility at a given location. Yu also restricts the location costs to the budget set by the firm looking into opening a new facility.

Facility Location Problems (FLPs) can be generally categorized into covering-based and median-based problems and other less common problems such as the p-dispersion and maxisum dispersion [4]. One of the covering-based problem models is the Set Covering (SC) location problem, where the objective is to minimize the total location cost to cover all demand points [5]. The p-median location problem is an example of a median-based problem model, where the objective is to minimize the demand-weighted total distance or time between the facilities and the demand points [6]. The capacity constraint can be added to an FLP, whereby every facility can service a finite number of demand points [7]. A problem can be formulated by combining multiple objectives and criteria, as Farahani et al. explained[8]. Fuzzy logic, which involves working with uncertainty in data, can also be employed as a solution if it applies to the problem statement at hand and the datasets within reach of the developers [9]. One example of using fuzzy logic in this context is how Karasakal and Silav [10] combined the Non-Sorting Genetic Algorithm (NSGA-II) with the fuzzy logic of partial coverage.

In FLP, facility type limits the solution algorithm and formulation of the problem model. For instance, emergency facilities such as police stations, hospitals, and testing centers have strict timing as police and ambulances must reach destinations fast [11], modeled as a covering problem model [12], with an evolutionary algorithm typically used[13]. Meanwhile, energy facilities have a greater focus on fixed and variable costs [14]. This paper focuses on the COVID center as a facility, aiming to maximize the collective coverage for the city's population.

The rest of this paper is structured as follows: Section 2 analyzes the related literature and compares the design decisions made in this paper's methodology with that of other relevant papers in the literature. Section 3 presents the mathematical formulation of the problem model data preparation and objective function implementation. Section 4 presents and discusses the results of Genetic and Monkey algorithms and compares their performance. Section 5 concludes the paper and introduces potential future work.

II. LITERATURE REVIEW

In this section, we compare and contrast all aspects of our design and implementation with the literature and justify our decisions for each aspect.

A. FLP in Healthcare

Based on our literature review, 12 papers have discussed solving an FLP for a healthcare type of facility with an Artificial Intelligence approach, nine of which used a discrete search domain, similar to our case. Kamikawa and HASUIKE [15] modeled their problem as a hybrid between a set of regression equations and a Weber problem. Tavakkoli-

Moghaddama et al. [16] formulate the problem as a combination p-median problem where the total transportation costs are minimized and the social impact is maximized. Zhang et al. [17] designed three objectives around population coverage, whereas Zhang et al. [18] designed a bilevel objective. The papers by Elkady and Abdelsalam [19], ElKady and Abdelsalam [20], and Shariff et al. [21] each discussed how they model their problem around the capacitated maximal covering location problem. Similarly, Fernandes et al. [22] worked on a Two-Stage Capacitated Facility Location Problem. LU et al. [23] integrated fuzzy logic into their problem model and formulated it as a fuzzy queuing maximal covering problem. Berglund and Kwon [24] formulate their problem as a set covering problem in which they work at minimizing total cost. As the majority of the modeling approaches agree with our collected demographics dataset, we will model our problem as a capacitated maximal covering location problem.

B. GA and MA Algorithms

Moreover, we found 19 papers utilizing the GA standalone, two papers combining it in a hybrid solution approach, and only one paper utilizing the MA. Hernandez et al. [25] combined the GA with the Probabilistic Solution Discovery Algorithm (PSDA) to hedge against facility failure and build a more robust set of facilities. Mousavi et al. combine the GA with vibration damping optimization (VDO) to solve a capacitated multi-facility FLP [26]. Our hypothesis is that a hybrid approach is more effective than using any single algorithm on the search domain.

Four papers only were found to combine GIS and AI in one proposal. Rahman et al. [27] was mentioned before and is one of the latest papers to discuss the combination of GIS and AI. In addition, Shatnawi et al. [28] used particle swarm optimization (PSO) and the GA with GIS in 2020, showing that this paper is not the first to use the GA with GIS. Saeidian et al. [29] used the PSO and ant colony optimization with GIS. Lastly, Yeo and Yee [30] used an artificial neural network with GIS. To the authors' knowledge, this paper is the only paper that has implemented the MA with GIS. This implementation will be done using ArcGIS Pro software, with a Python-Django backend utilizing ArcPy – the library through which we used the ArcGIS geoprocessing tools and interacted with its geodatabase – and a React-based frontend.

C. GA Development

The foundation of our GA is based on the NSGA-II by Deb et al. [31]. While most papers do not consider the duplication of gene selection when creating a chromosome in the initial population or while performing mutation, we consider this our primary modification to our implementation of the GA. The duplication is necessary since our proposal is a multi-output application. Korać et al. [32] remove duplicates from the genetic operators, such as mutation, in each generation. We find it important to replace these with non-duplicates instead of simply eliminating them to ensure a predictable number of output genes in any given chromosome. The modification is done in the mating function, where if a gene is randomly selected and is already found in the chromosome, a new gene is selected. The implemented GA is discussed in the next section.

D. Our Contribution

In this paper, we develop a novel MA to resolve CMCLP for healthcare facilities within a geospatial environment. The solution is developed using ArcGIS software with Python backend and ArcPy for geoprocessing to model the problem in real-world data from the emirate of Dubai, demonstrating the feasibility of MA. A comparative analysis was performed to validate reliable performance by comparing the GA and MA with a custom-designed MA.

III. Methodology

A. Data Preparation

The primary dataset we needed was the population data for the city of Dubai, which we obtained from Esri's ArcGIS Pro software. It offers a polygon enrichment tool that calculates the population demographics within a given polygon according to 2020 data. Given the nature of our dataset, we do not need to use machine learning to preprocess our data.

Given that the datasets we found from Open Street Maps (OSM) for the healthcare and amenity facilities are discrete, given that the application's objective is to locate the best-existing amenity facilities to build a COVID center, our search domain will also be discrete.

B. Mathematical Formulation

This paper focuses on capacitating FLP and the maximizing covering location problem, named the capacitated maximal covering location problem (CMCLP). It maximizes the covered demand of each facility, which, in our case, is represented by population demographics. The mathematical formulation for our problem is as follows [21]:

$$Maximize \sum_{j \in J} \sum_{i \in I} c_{ij} a_j y_{ij} \quad (1)$$

With the following constraints:

$$\sum_{i \in I} x_i \leq p \quad (2)$$

$$\sum_{i \in I} y_{ij} = 1, \quad j \in J \quad (3)$$

$$y_{ij} \in \{0, 1\}, x_i \in \{0,1\}, \quad i \in I, j \in J \quad (4)$$

$$d_{ij} \leq s_i x_i, \quad \forall(i,j) \quad (5)$$

Where,

- i is a candidate facility in a set of candidate facilities I,
- j is a demand point in a set of demand points J,
- c_{ij} is a Boolean indicated whether customer j is within the service area of facility i,
- a_j is the demand volume at demand point j,
- y_{ij} is a Boolean indicating whether demand point j is allocated to facility i,
- x_i is a Boolean indicating whether a facility is established as candidate location i,
- p is the number of facilities to be established,
- d_{ij} is the demand of customer j corresponding to the facility i,

- and s_i is the demand capacity for facility I.

The objective function (1) maximizes the demand allocated to all facilities while respecting the service areas. Equation (2) requires a predefined number of facilities to be established. Equation (3) ensures that a demand point is only allocated to a facility if it is established. Equation (4) defines the binary values for the demand point allocation and the facility establishment variables, respectively. Finally, we have incorporated (5) into the list of constraints to consider each facility's capacity [33].

C. CMCLP Implementation

The search domain is subdivided into 1000 equal segments in the data preparation stage. Population demographics are then calculated for each segment to enrich the data, as shown in Fig. 1.

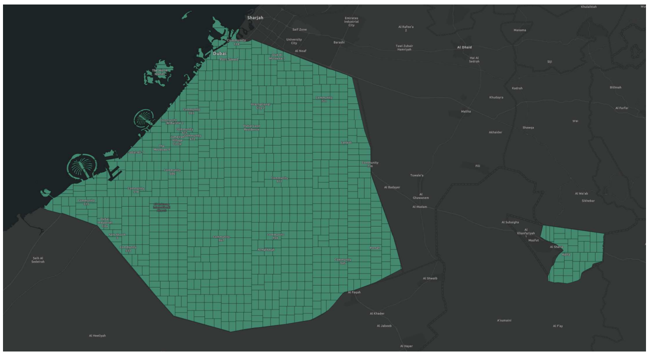


Fig. 1. Division of the Emirates of Dubai into 1,000 enriched segments.

The segmentation is run through two subsequent stages of processing. The first is base processing, where demand for 570 healthcare facilities is applied to the segments' demand values. The service areas of each healthcare facility are calculated based on traffic coverage within a one-mile radius. Traffic coverage refers to the area where traffic can flow and, consequently, residential units can exist. The service areas and demand segments are run through an intersection analysis that fragments the service areas according to their intersections with the segments. Fig. 2 shows the fragments of the service area for one of the healthcare facilities.

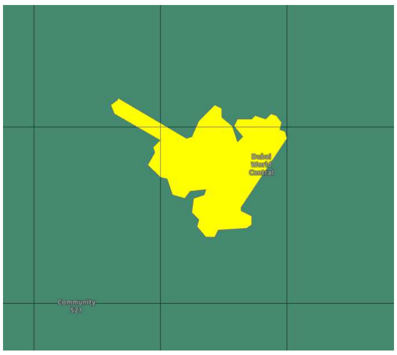


Fig. 2. Base intersection of the service area of a healthcare facility.

Each healthcare facility's impact is applied to each of its fragments. The facility impact is calculated by finding two demand values: the demand available within the fragment and the potential demand the facility can cover. Demand available within the fragment can be calculated with (6) below:

$$\textit{Demand within Fragment} = \frac{\text{Fragment Area}}{\text{Associated Segment Area}} * \text{Demand within Segment} \tag{6}$$

The potential demand that the facility can cover within the fragment can be calculated with (7) below:

$$\textit{Potential Demand Covered within Fragment} = \frac{\textit{Fragment Area}}{\textit{Service Area}} * \textit{Facility Capacity} \tag{7}$$

The maximum capacity for each healthcare facility is set at 5000 people/day, while the capacity for the candidate facilities is set at 3000 people/day. Both of these values are user-changeable. If there exists equal or more demand than the facility can cover within the fragment, then the full potential demand is deducted from the segment's demand; otherwise, the demand of the fragment is deducted from that of the segment.

The second stage of processing is candidate processing. It is almost identical to base processing, except that the demand from (6) is not calculated at this step, as the demand within the segment will be different at each point of metaheuristic algorithm execution. There are 256 candidate facilities. Fig. 3 shows the service areas in yellow for the candidate facilities.

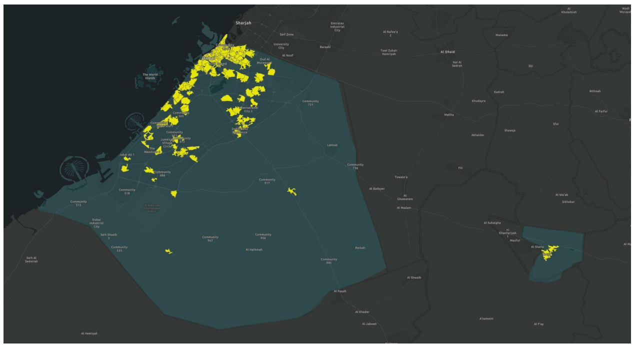


Fig. 3. Service areas for candidate facilities.

As discussed later, the metaheuristic algorithm is implemented to select five available candidate facilities. Since multiple outputs are expected, applying each candidate's demand to the demand segments will affect the following application. As such, every iteration of the algorithm will create a deep copy of the demand segments to simulate the effect of the candidates under consideration. The demand within each associated fragment will be calculated and deducted from the simulated segments. This process will ensure that the impact of every candidate facility is reflected. The steps used in implementing and using the CMCLP are:

- Search Domain Segmentation
- Base Service Areas Generation
- Base Intersection Analysis
- Base Fragments Processing
- Candidate Service Areas Generation
- Candidate Intersection Analysis
- Candidate Fragments Initial Processing
- Node Evaluation Per Algorithm Iteration

1) Genetic Algorithm Implementation

With a few minor improvements discussed in section II, our GA implementation is fundamentally based on the NSGA-II by Deb et al. [31]. The genetic algorithm is presented with a list of candidate facilities and mimics the behavior of genetic evolution to generate the expected output. This input list is ready after the candidate processing phase, as shown in Fig. 1. For this experiment, we set the desired output at five ordered candidate facilities, although this number is user-changeable. The order of the output solution set must be accounted for in the calculation of its fitness because applying demand implications on the segments would affect the next candidate facility in case their service areas overlap.

The first step in the algorithm is generating the initial population, which will be a random selection from the list of provided candidate facilities. This value is also user-changeable.

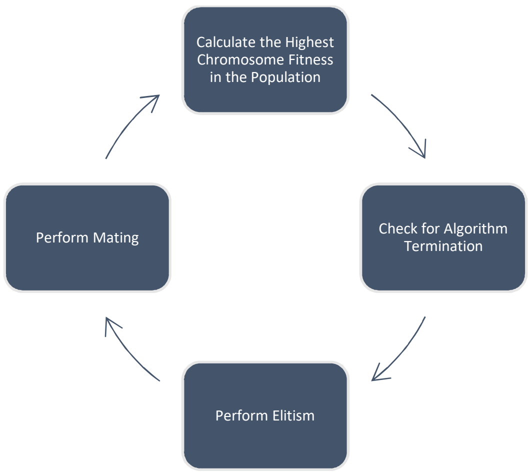


Fig. 4. Genetic algorithm main loop

The algorithm's main loop is then executed, as illustrated in Fig. 4. The standalone fitness of each gene in each chromosome is calculated in the initial population. The total standalone fitness for each gene is calculated and combined to become that of the parent chromosome. The chromosome with the highest total standalone fitness is then passed on to the CMCLP function. The returned fitness is considered the generation's fitness and is passed on to be checked against the algorithm termination mechanism. If the algorithm does not terminate, the population will undergo Elitism, where the fittest 10% of the population is passed on as is to the next generation. The other 90% start the process of Mating, where the remaining chromosomes will be split into two groups. Chromosomes from each group will enter the mating process. There is a 40% chance that the gene from the first chromosome will be passed on to the resulting chromosome, a 40% chance that the gene from the second chromosome will have its gene passed on, and a 20% chance for the resulting chromosome to be mutated. Mutation is the exploration mechanism for the genetic algorithm, where a new random candidate facility is selected and passed on as a gene. The mutation feature prevents the algorithm from resolving to a local maximum. At every step of the algorithm, duplicates are avoided. The main loop repeats until the termination mechanism breaks it.

D. Custom Monkey (MA) Algorithm Implementation

The MA replicates the principles of the behavior of a group of monkeys searching for food. Each monkey can perform three movements. First is a climb, where the monkey climbs to the nearest vantage point and looks for food near it. Second is a watch and jump, where the monkey scouts for a higher closest vantage point to move to and performs another climb. Third, the monkey can perform a somersault, taking a significant leap in a random direction before undergoing another climb. The somersault marks the exploration mechanism for the MA, just like mutation does in the genetic algorithm. This function prevents the algorithm from getting stuck at a local solution set. After all the monkeys have performed their movements, they will undergo a coordination phase, where they communicate their best solutions with each other.

For comparison purposes, the expected output is identical to that from the genetic algorithm: a solution set of five ordered candidate facilities. Since the MA is highly contextualized to its spatial search domain, it was implemented from scratch based on the aforementioned principles to suit the ArcGIS environment.

The algorithm begins by randomly spawning a preset number of monkeys within the boundaries of Dubai, with a minimum distance between each two monkeys of five miles, which was experimentally determined. This value is user changeable. Fig. 5 shows the spawned initial population from an example run, symbolized as yellow X's within the blue Dubai polygon.

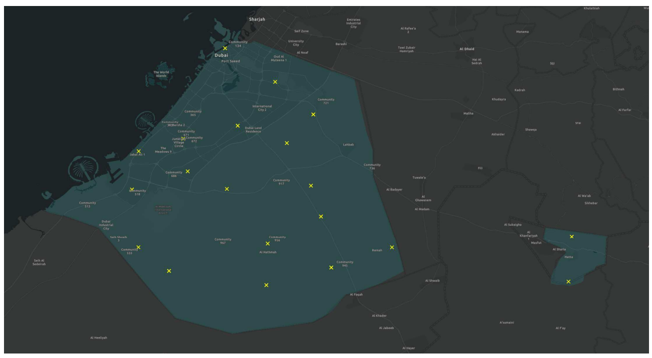


Fig. 5. Monkey algorithm initial population

The algorithm's main loop is then executed, as shown in Fig. 6. Each monkey will perform a climbing movement first. Each point representing a monkey within the Dubai boundaries will find the five closest candidate facilities and evaluate them using the implemented CMCLP solver. It will then nominate the best out of the five during the coordination phase. The best five solutions out of all nominated candidate solutions will move on to the next step. The five selected solutions form an ordered solution set, the fitness of which is then calculated. The result represents the fitness of that generation. If applicable, the algorithm checks that fitness against the termination mechanism and breaks the loop. If not, each monkey will run through a random movement. There is an 80% chance that the monkey will perform a watch-jump and a 20% chance of performing a somersault. The watch-jump movement will have the monkey analyze the population of its surrounding segments and move the monkey's representative point toward the highest population segment. The somersault will select a random direction that does not take the monkey outside the emirate boundaries and move the monkey's point a large distance towards it. At every step of the algorithm, duplicates are avoided. The main loop repeats until the termination mechanism breaks it.

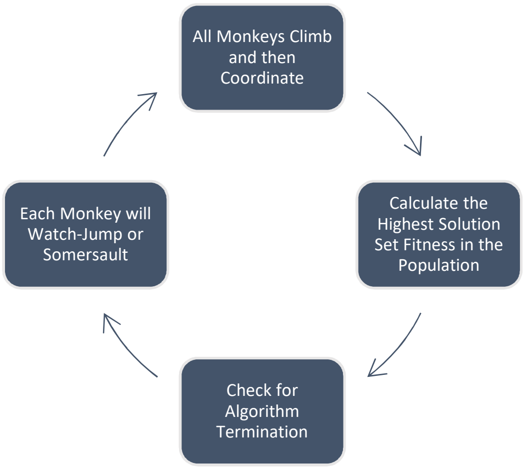


Fig. 6. Monkey algorithm main loop.

It is important to note that the code can be generalized to work with fitness and standalone fitness instead of incorporating population segments. The algorithm can then be used for any problem with a spatial search domain. We chose to use the population segments because it is computationally faster to specialize the algorithm programming to our application. It is also important to note that the algorithm can be reprogrammed to work with a continuous search domain by skipping the search for the closest candidate service areas to each monkey.

E. Fitness Function

The fitness optimal value (OPT) for any given run is calculated as per (7):

$$OPT = \text{Canidate Facility Capacity} \\ * \text{Size of Output Solution Set} \quad (7)$$

Since the default candidate facility capacity is set to 3000 people and the default size of the output solution set is set to 5 candidate facilities, the default OPT is 15,000.

The fitness function is then used to calculate fitness as per (8):

$$\mathit{Generation\ Fitness} \\ = (\text{Total Demand from the CMCLP function} \\ * \text{OPT}) * 100 \quad (8)$$

The fitness function is identical in both the Genetic and Monkey algorithms.

F. Termination Mechanism

The termination mechanism is identical for both implemented algorithms. The algorithm is terminated when fitness reaches 100 or no better solution set has been found within a given number of grace generations. The number of grace generations for each algorithm is tuned and determined in the next section to achieve the best fitness-to-run duration performance. This number is a user-changeable value.

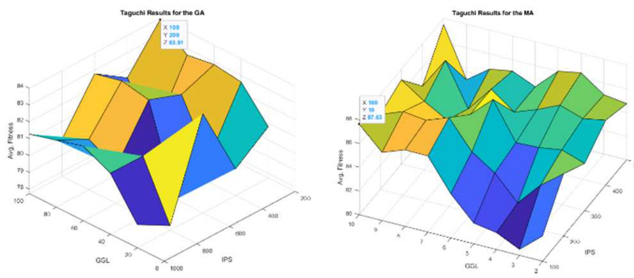
G. Tuning the Parameters

In order to tune the two algorithm parameters for each algorithm (i.e., initial population size and grace generations limit), we will use the Taguchi method. In the Taguchi method, we test for all combinations of values within a wide range of parameter values and determine the best values using the best fitness yield.

We set the healthcare and candidate capacities to a constant 10,000 persons/day. This value is a realistically large number for a COVID center that would challenge both algorithms and ensure that neither would reach a 100 fitness value. It would also ensure enough headroom for both algorithms to prove their performance. A fitness value of 100 indicates a fitness value equal to the optimal value. We chose parameters for the test based on run duration feasibility. We tested the GA with ranges of [200, 1000] and [20, 100] for the initial population size and grace generations limit, respectively. We then tested the MA with ranges [100, 500] and [2, 10] for the initial population size and grace generations limit, respectively.

IV. Results and Discussion

Fig. 7 shows the surf plots of the IPS, GGL, and fitness values for the GA and MA, respectively. The peak is selected in each plot, and the X (GGL) and Y (IPS) values are displayed. For the GA, it was determined that 200 units and 100 generations were the optimal parameter values for the IPS and GGL, respectively. For the MA, it was determined that 100 units and 10 generations were the optimal parameter values for the IPS and GGL, respectively.



(a) (b)

Fig. 7. Taguchi results for (a) GA and (b) MA

A. Statistical Analysis

Twenty runs were performed on each algorithm for statistical analysis. Table 1 shows the experimental results.

TABLE I. EXPERIMENTAL ALGORITHM RESULTS

Trial	GA Fitness	GA Duration	MA Fitness	MA Duration
1	85.86	83.00	85.44	165.00
2	80.54	87.00	85.40	151.00
3	78.67	83.00	85.82	120.00
4	85.79	81.00	85.64	71.00
5	81.88	83.00	85.64	102.00
6	80.88	83.00	87.67	102.00
7	87.26	80.00	87.55	65.00
8	80.28	82.00	87.67	108.00
9	70.64	81.00	85.64	70.00
10	81.83	81.00	88.09	65.00
11	86.20	82.00	85.93	83.00
12	83.95	84.00	85.64	71.00
13	82.89	84.00	85.64	107.00
14	79.44	80.00	87.80	199.00

Trial	GA Fitness	GA Duration	MA Fitness	MA Duration
15	84.54	81.00	87.13	65.00
16	81.94	82.00	85.64	71.00
17	83.47	82.00	87.67	152.00
18	83.90	81.00	87.80	71.00
19	83.13	80.00	87.44	83.00
20	83.02	85.00	85.88	70.00

Statistical analysis was performed on the data in Table 2 using Minitab. We first performed a Mann-Whitney test to confirm statistical differences before proceeding with the consistency analysis. Table 2 shows the test results for the fitness values. Both P-values are below the significance level of 0.05, indicating that the fitness values are statistically different. This significance allows us to differentiate the fitness performance of the two algorithms. Table 2 also shows the test results for the duration values. Both P-values this time are above the significance level, indicating that the run duration times are statistically indifferent. It concludes that the algorithms take a comparable amount of time to execute many runs.

TABLE II. MANN-WHITNEY TEST RESULTS.

Condition	Method	W-Value	P-Value
For fitness values	Not adjusted for ties	250.00	0.000
	Adjusted for ties	250.00	0.000
For run duration values	Not adjusted for ties	402.00	0.839
	Adjusted for ties	402.00	0.839

Table 3 shows the results of the descriptive analysis. The parameters for consistency are range, variance, and standard deviation. The GA has a fitness range of 16.620, while the MA has a fitness range of 2.682, which proves the greater consistency of the MA in comparison with the GA. The variance and standard deviation for fitness recorded by GA are 12.892 and 3.591, respectively, while the same values for fitness recorded by the MA are 1.067 and 1.033, reaffirming the statistical consistency of the fitness performance metric for the MA. Finally, after comparing fitness averages, we can conclude that the MA is more performant, with a 5.16% higher average fitness, a variant that is 11 times lower (indicating a more consistent fitness output), and a 21.03% run duration penalty.

TABLE III. DESCRIPTIVE STATISTICS FOR THE DATA

Variable	Mean	StDev	Variance	Minimum	Q1	Median	Q3	Maximum	Range
GA Fitness	82.306	3.591	12.892	70.636	80.628	82.957	84.392	87.256	16.620
GA Duration	82.250	1.803	3.250	80.000	81.000	82.000	83.000	87.000	7.000
MA Fitness	86.556	1.033	1.067	85.404	85.638	85.908	87.668	88.086	2.682
MA Duration	99.55	39.25	1540.26	65.00	70.25	83.00	117.00	199.00	134.00

Fig. 8 shows the fitness values of the genetic and the monkey algorithms. It demonstrates that the MA values are more stable than the GA.

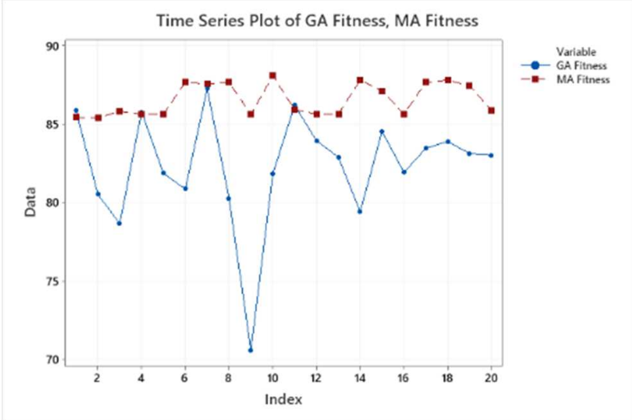


Fig. 8. Fitness values plot.

V. CONCLUSION

The effort to minimize the spread of COVID is of predominant importance. Increasing the availability of COVID vaccination and/or testing centers is part of that effort. This paper aims to develop an application with which a user can leverage evolutionary algorithms to locate optimal facilities where a COVID center can be built and to allocate demand to those potential facilities. The Genetic and Monkey algorithms were implemented and compared with each other in light of their optimal values. Both algorithms were tested using the same fitness function and termination mechanism for comparison. The MA proved more performant and was the better choice for this application.

For further research, the healthcare and candidate facilities can be updated to a timeframe well after the start of the COVID-19 pandemic, resulting in a more up-to-date solution. The problem formulation can also be expanded to include fixed location costs if the required datasets can be found. Moreover, the objective function can be solved with other choices of algorithms so long as they include an exploration mechanism. Subsequently, the application can be expanded to other emirates within the United Arab Emirates or other countries or cities worldwide.

ACKNOWLEDGMENT

We would like to express our sincere gratitude to the University of Sharjah in the UAE for funding this research. We thank our colleagues from OpenUAE Research and Development Group at the University of Sharjah and the support from the Dubai Police Scientists Council and the Department of Geographical Information System (GIS) Centre at Dubai Municipality. A special appreciation to the expertise and effort put into this project from Mohammed Alaa Ajlouni.

REFERENCES

[1] Q. Peng and H. Afshari, "Challenges and Solutions for Location of Healthcare Facilities," 2016, doi: 10.4172/2169-0316.1000127.
[2] S. Zhong, R. Cheng, Y. Jiang, Z. Wang, A. Larsen, and O. A. Nielsen, "Risk-averse optimization of disaster relief facility location and vehicle routing under stochastic demand," *Transp Res E Logist Transp Rev*, vol. 141, p. 102015, Sep. 2020, doi: 10.1016/j.tre.2020.102015.
[3] W. Yu, "Robust Model of Discrete Competitive Facility Location Problem with Partially Proportional Rule," *Math Probl Eng*, vol. 2020, 2020, doi: 10.1155/2020/3107431.

[4] A. Ahmadi-Javid, P. Seyedi, and S. S. Syam, "A survey of healthcare facility location," Mar. 01, 2017, *Elsevier Ltd*. doi: 10.1016/j.cor.2016.05.018.

[5] B. Crawford, R. Soto, E. Monfroy, G. Astorga, J. García, and E. Cortes, "A meta-optimization approach for covering problems in facility location," in *Communications in Computer and Information Science*, Springer Verlag, 2017, pp. 565–578. doi: 10.1007/978-3-319-66963-2_50.

[6] T. Levanova and A. Gnusarev, "Development of Ant Colony Optimization Algorithm for Competitive p-Median Facility Location Problem with Elastic Demand," in *Communications in Computer and Information Science*, Springer, Jul. 2019, pp. 68–78. doi: 10.1007/978-3-030-33394-2_6.

[7] E. D. Güneş, T. Melo, and S. Nickel, "Location Problems in Healthcare," in *Location Science*, Springer International Publishing, 2019, pp. 657–686. doi: 10.1007/978-3-030-32177-2_23.

[8] R. Z. Farahani, M. SteadieSeifi, and N. Asgari, "Multiple criteria facility location problems: A survey," Jul. 01, 2010, *Elsevier*. doi: 10.1016/j.apm.2009.10.005.

[9] C. Kahraman, S. Cebi, and F. Tuysuz, "Fuzzy location selection techniques," *Studies in Fuzziness and Soft Computing*, vol. 252, pp. 329–358, 2010, doi: 10.1007/978-3-642-12052-7_14.

[10] E. Karasakal and A. Silav, "A multi-objective genetic algorithm for a bi-objective facility location problem with partial coverage," *TOP*, vol. 24, no. 1, pp. 206–232, Apr. 2016, doi: 10.1007/s11750-015-0386-8.

[11] A. M. Caunhye, X. Nie, and S. Pokharel, "Optimization models in emergency logistics: A literature review," Mar. 01, 2012, *Pergamon*. doi: 10.1016/j.seps.2011.04.004.

[12] X. Li, Z. Zhao, X. Zhu, and T. Wyatt, "Covering models and optimization techniques for emergency response facility location and planning: A review," *Mathematical Methods of Operations Research*, vol. 74, no. 3, pp. 281–310, Dec. 2011, doi: 10.1007/s00186-011-0363-4.

[13] Y. J. Zheng, S. Y. Chen, and H. F. Ling, "Evolutionary optimization for disaster relief operations: A survey," *Appl Soft Comput*, vol. 27, pp. 553–566, Feb. 2015, doi: 10.1016/j.asoc.2014.09.041.

[14] C. Ou-Yang and R. Ansari, "Applying a hybrid particle swarm optimization_Tabu search algorithm to a facility location case in Jakarta," *Journal of Industrial and Production Engineering*, vol. 34, no. 3, pp. 199–212, Apr. 2017, doi: 10.1080/21681015.2016.1243167.

[15] T. Kamikawa and T. Hasuike, "Multi-objective optimization of allocations and locations of incineration facilities with Voronoi diagram and genetic algorithm: Case study of northwest bay area in Chiba prefecture," in *2019 IEEE 11th International Workshop on Computational Intelligence and Applications, IWCIA 2019 - Proceedings*, Institute of Electrical and Electronics Engineers Inc., Nov. 2019, pp. 35–40. doi: 10.1109/IWCIA47330.2019.8955065.

[16] R. Tavakkoli-Moghaddam, P. Pourreza, A. Bozorgi-Amiri, and N. Oladzad, "A Bi-objective Credibility-based Fuzzy Mathematical Programming Model for a Healthcare Facility Location-network Design Problem," in *IEEE International Conference on Industrial Engineering and Engineering Management*, IEEE Computer Society, Jan. 2019, pp. 1181–1185. doi: 10.1109/IEEM.2018.8607282.

[17] W. Zhang, K. Cao, S. Liu, and B. Huang, "A multi-objective optimization approach for health-care facility location-allocation problems in highly developed cities such as Hong Kong," *Computers, Environment and Urban Systems*, vol. 59, pp. 220–230, Sep. 2016, doi: 10.1016/j.compenvurbsys.2016.07.001.

[18] Y. Zhang, O. Berman, P. Marcotte, and V. Verter, "A bilevel model for preventive healthcare facility network design with congestion," *IIE Transactions (Institute of Industrial Engineers)*, vol. 42, no. 12, pp. 865–880, Dec. 2010, doi: 10.1080/0740817X.2010.491500.

[19] S. K. Elkady and H. M. Abdelsalam, "A modified multi-objective particle swarm optimisation algorithm for healthcare facility planning," *International Journal of Business and Systems Research*, vol. 10, no. 1, pp. 1–22, 2016, doi: 10.1504/IJBSR.2016.073685.

[20] S. K. Elkady and H. M. Abdelsalam, "A simulation-based optimization approach for healthcare facility location allocation decision," in *Proceedings of the 2015 Science and Information Conference, SAI 2015*, Institute of Electrical and Electronics Engineers Inc., Sep. 2015, pp. 500–505. doi: 10.1109/SAI.2015.7237189.

[21] S. S. R. Shariff, N. H. Moin, and M. Omar, "Location allocation modeling for healthcare facility planning in Malaysia," *Comput Ind Eng*, vol. 62, no. 4, pp. 1000–1010, May 2012, doi: 10.1016/j.cie.2011.12.026.

[22] D. R. M. Fernandes, C. Rocha, D. Aloise, G. M. Ribeiro, E. M. Santos, and A. Silva, "A simple and effective genetic algorithm for the two-stage capacitated facility location problem," *Computers and Industrial Engineering*, vol. 75, no. 1, pp. 200–208, Sep. 2014, doi: 10.1016/j.cie.2014.05.023.

[23] X. L. Lu, Y. X. Hou, and Q. Shen, "A fuzzy queuing facility location model with ant colony optimization algorithm for large-scale emergencies," in *2010 7th International Conference on Service Systems and Service Management, Proceedings of ICSSSM' 10*, 2010, pp. 55–60. doi: 10.1109/ICSSSM.2010.5530267.

[24] P. G. Berglund and C. Kwon, "Robust Facility Location Problem for Hazardous Waste Transportation," *Networks and Spatial Economics*, vol. 14, no. 1, pp. 91–116, Oct. 2014, doi: 10.1007/s11067-013-9208-4.

[25] I. Hernandez, J. Emmanuel Ramirez-Marquez, C. Rainwater, E. Pohl, and H. Medal, "Robust facility location: Hedging against failures," *Reliability Engineering and System Safety*, vol. 123, pp. 73–80, Mar. 2014, doi: 10.1016/j.ress.2013.10.006.

[26] S. M. Mousavi, S. T. A. Niaki, E. Mehdizadeh, and M. R. Tavarroth, "The capacitated multi-facility location-allocation problem with probabilistic customer location and demand: Two hybrid meta-heuristic algorithms," *International Journal of Systems Science*, vol. 44, no. 10, pp. 1897–1912, Oct. 2013, doi: 10.1080/00207721.2012.670301.

[27] M. Rahman *et al.*, "Location-allocation modeling for emergency evacuation planning with GIS and remote sensing: A case study of Northeast Bangladesh," *Geoscience Frontiers*, vol. 12, no. 3, p. 101095, May 2021, doi: 10.1016/j.gsf.2020.09.022.

[28] N. Shatnawi, A. A. Al-Omari, and H. Al-Qudah, "Optimization of Bus Stops Locations Using GIS Techniques and Artificial Intelligence," in *Procedia Manufacturing*, Elsevier B.V., Jan. 2020, pp. 52–59. doi: 10.1016/j.promfg.2020.02.204.

[29] B. Saeidian, M. S. Mesgari, B. Pradhan, and M. Ghodousi, "Optimized location-allocation of earthquake relief centers using PSO and ACO, complemented by GIS, clustering, and TOPSIS," *ISPRS International Journal of Geo-Information*, vol. 7, no. 8, p. 292, Aug. 2018, doi: 10.3390/ijgi7080292.

[30] I. A. Yeo and J. J. Yee, "A proposal for a site location planning model of environmentally friendly urban energy supply plants using an environment and energy geographical information system (E-GIS) database (DB) and an artificial neural network (ANN)," *Applied Energy*, vol. 119, pp. 99–117, Apr. 2014, doi: 10.1016/j.apenergy.2013.12.060.

[31] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 2, pp. 182–197, Apr. 2002, doi: 10.1109/4235.996017.

[32] V. Korać, J. Kratica, and A. Savić, "An improved genetic algorithm for the multi level uncapacitated facility location problem," *International Journal of Computers, Communications and Control*, vol. 8, no. 6, pp. 845–853, Nov. 2013, doi: 10.15837/ijccc.2013.6.134.

[33] A. Lodi, L. Mossina, and E. Rachelson, "Learning to handle parameter perturbations in Combinatorial Optimization: An application to facility location," *EURO Journal on Transportation and Logistics*, vol. 9, no. 4, p. 100023, Dec. 2020, doi: 10.1016/j.ejtl.2020.100023.