Scientific Research Publishing

Journal of Computer and Communications, 2023, 11, 24-31
https://www.scirp.org/journal/jcc
ISSN Online: 2327-5227
ISSN Print: 2327-5219

Use the Power of a Genetic Algorithm to Maximize and Minimize Cases to Solve Capacity Supplying Optimization and Travelling Salesman in Nested Problems

Ali Abdulhafidh Ibrahim, Hajar Araz Qader, Nour Ai-Huda Akram Latif

Al-Nahrain University, Baghdad, Iraq
Email: dr_ali9@yahoo.com, hageraraz@gmail.com, Homelander18@gmail.com

How to cite this paper: Ibrahim, A.A., Qader, H.A. and Latif, N.A.-H.A. (2023) Use the Power of a Genetic Algorithm to Maximize and Minimize Cases to Solve Capacity Supplying Optimization and Travelling Salesman in Nested Problems. *Journal of Computer and Communications*, **11**, 24-31.
https://doi.org/10.4236/jcc.2023.113003

Received: January 20, 2023
Accepted: March 26, 2023
Published: March 29, 2023

Abstract

Using Genetic Algorithms (GAs) is a powerful tool to get solution to large scale design optimization problems. This paper used GA to solve complicated design optimization problems in two different applications. The aims are to implement the genetic algorithm to solve these two different (nested) problems, and to get the best or optimization solutions.

Keywords

Genetic Algorithm, Capacity Supplying Optimization, Traveling Salesman Problem, Nested Problems

1. Introduction

Numerous research studies have been carried out to discover optimal or best trading rules based on technical indicators in different applications for problems using individual approaches, such as Genetic Algorithm [1] [2] [3].

Genetic Algorithm is one main area of Evolutionary Algorithms. The main idea of EA is to reproduce population depending on individuals who meet chosen criteria, and cancel the other individuals that not meet the criteria.

An implementation of a genetic algorithm begins with a population of chromosomes (random process). One then evaluates these structures and allocates reproductive opportunities in such a way that those chromosomes which represent a better solution to the target problem are given more chances to "reproduce" (and to appear) than those chromosomes which are poorer solutions. The genetic algorithm refers to a model introduced and investigated by John Holland (1970). In a broader usage of the term, a genetic algorithm is any population-based

model that uses selection and recombination operators to generate new sample points in a search space. Many genetic algorithm models have been introduced by researchers from an experimental perspective [4] [5].

This study paper used the GA to solve two nested problems: the first was TSP, while the second one was Capacity Supplying Optimization (CSO).

We implement GA for CSO to find a good solution to allocate specific equipment's or items in a fix space or Truck, in other words is how to distribute equipment's in a limited storage to maximize the profits.

One of the popular problems was the traveling salesman problem (TSP) [6]. TSP was studied first by William Rowam Hamilton and Thomas Penyngton Kirkman in the 18th century. TSP aims to determine the cost function, which is travel (that includes the distance) between each possible pair, the TSP is to find the best possible way of visiting all the cities and returning to the starting point that minimizes the travel cost (or travel distance) to minimize the costs. The second section contains data sets, the third section concerned represents the theory of this paper, while the forth section contains the results and discussion for this paper RESULTS AND DISCUSSION, The final section contains the conclusions that were derived from the results of the present research.

2. Data Sets [7]-[12]

This paper consists of two data sets, as follows:

1) First Data Set: contains four local companies in Iraq counter in the field of selling electrical devices and selling household goods and products, which they are:

- LG Company,
- Al-Hafiz Company,
- Royal Company, and
- Samsung Company.

These electrical equipment's, which are prices, and size of each electrical equipment's, in Iraq market. **Figure 1** reads data using python program.

Where these companies manufacture devices a variety of electric vehicles in different sizes and prices through a truck with a limited capacity, as shown in **Table 1** [7] [8] [9] [10].

2) Second Data Set contains Distance costs associated with time of each journey, that consists of starting and ending each journey between two cities in Iraq. which Represented by the transport problem of quantities of goods represented by devices Electricity requires transportation from Baghdad city to other cities and vice versa, where transportation prices are from Baghdad To the other cities are different according to the times, as well as the case for transportation from the cities to Baghdad, **Figure 2**.

3. Formalating the Problem of Genetic Algorithm

Genetic Algorithm

Genetic algorithms are optimization methods that search for best solution to the

```
products_list = []
products_list.append(Product("Refrigerator LG", 0.50, 865000))
products_list.append(Product("Refrigerator SUMSUNG", 0.55, 1200000))
products_list.append(Product("FREEZER ROYAL", 0.65, 296000))
products_list.append(Product("FREEZER ALHATH ", 0.75, 305000))
products_list.append(Product("AIR CONDITION LG ", 0.45, 900000))
products_list.append(Product("AIR CONDITION ALHATH", 0.35, 700000))
products_list.append(Product("Microwave Oven ALHATH", 0.35, 100000))
products_list.append(Product("Microwave  Oven SUMSUNG", 0.45, 110000))
products_list.append(Product("TELEVISION SUMSUNG", 1.245, 738520))
products_list.append(Product("TELEVISION LG", 1.09, 460000))
products_list.append(Product("electricmixer ALHATH", 0.14, 123000))
products_list.append(Product("electricmixer ROYAL", 0.15, 43000))
products_list.append(Product("cloth dryer ALHATH", 0.60, 658500))
products_list.append(Product("cloth dryer LG", 0.65, 700000))
spaces = []
prices = []
names = []
for product in products_list:
  spaces.append(product.space)
  prices.append(product.price)
  names.append(product.name)
limit = 3
population_size = 20
mutation_probability = 0.01
number_of_generations = 100
```

Figure 1. Reading first data type using python code, which contains name, sizes, and prices of devices for different companies.

Table 1. Iraqi companies manufacture devices a variety of electric vehicles in different sizes and prices through a truck with a limited capacity for different companies.

No.	Item	Prices/ID	Item No.	Size/cm³
1	Refrigerator LG	865,000	400	50
2	Refrigerator SUMSUNG	1,200,000	250	55
3	freezer royal	296,000	308	65
4	freezer Alhfth	305,000	500	75
5	Air condition LG	900,000	325	45
6	Air condition Alhfth	700,000	270	35
7	Microwave Oven Alhfth	100,000	450	35
8	Microwave Oven SUMSUNG	110,000	444	45
9	Television Sumsung	738,520	350	49
10	Television LG	460,000	500	43

problem under consideration until a particular ending condition is met. The solution to a problem is called a chromosome. The parameters to be optimized are called genes, which are the contents of the chromosome. The basic parts that are prevalent to almost every genetic algorithm are shown in **Figure 3** [5] [13] [14]:

1) Initial population: which it is set of chromosomes that randomly chosen in the beginning of the algorithm and this collection will be serving as the first

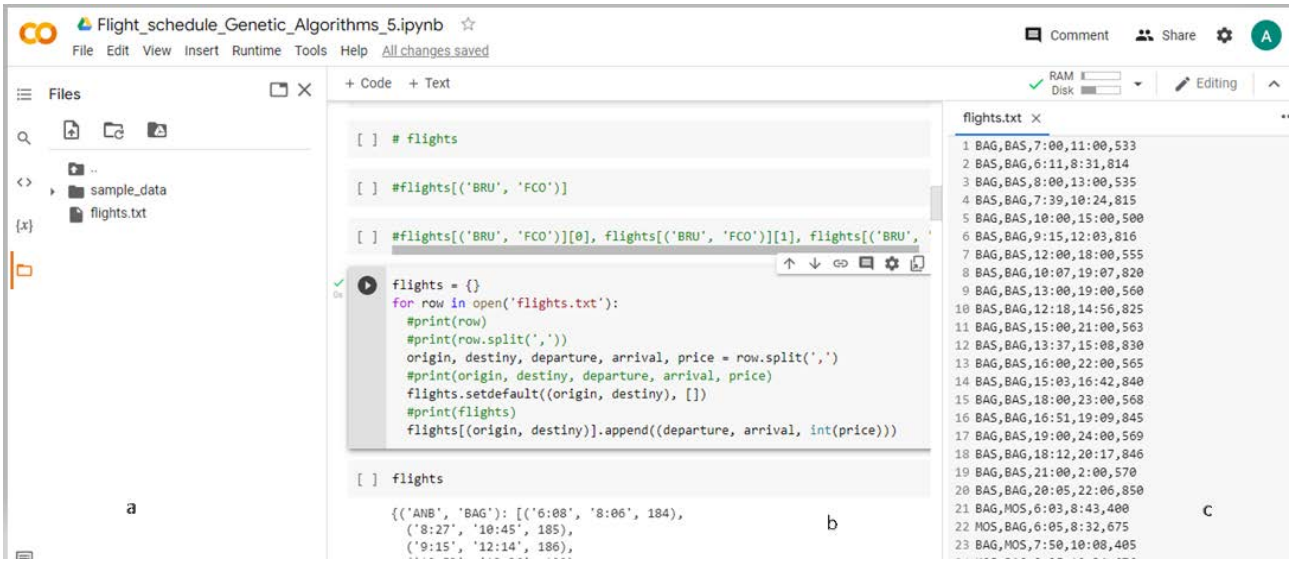

Figure 2. (a) represent the python code to read the second type of data, while (b) represent the data file which consist of transport data of quantities of goods represented by devices electricity requires transportation from Baghdad city to other cities and vice versa.

generation.

2) Fitness or evaluation function for optimization: this function is the most important component of the algorithm because it tests and quantifies how to fit each potential solution. The chromosome refers to a numeric value that symbolizes a candidate solution to the problem.

3) Selection operator: To reach an acceptable solution, a population must run through several generations. At each generation selection operator selects some of the chromosomes for reproduction based on a probability distribution defined by the user. The more fitter a chromosome is, it is likely to be selected [2].

a) Crossing: Recombination merges or crossed over parts of two or more parent's solutions to create new generation, perhaps better solutions (*i.e.* offspring). Cross over operator exchange a part of the sequence of the two selected chromosomes to produce two offspring.

b) Mutation: Mutation used to maintain genetic diversity of the chromosomes of a population of a genetic. It is analogous to biological mutation [8].

4. Results and Discussion

In this section, the implementation of GA to get the best solution for two problems under study, as follows:

1) Capacity Supplying Optimization

The implementation of GA to get the best solution for the first problem CSO by using the following Python code (see **Figure 4**).

The result (see **Figure 5**) after implementing Python code (**Figure 4**), as follows:

The optimization results of the prices (3,933,500) represent the maximum values that could we get using GA (see **Figure 6**).

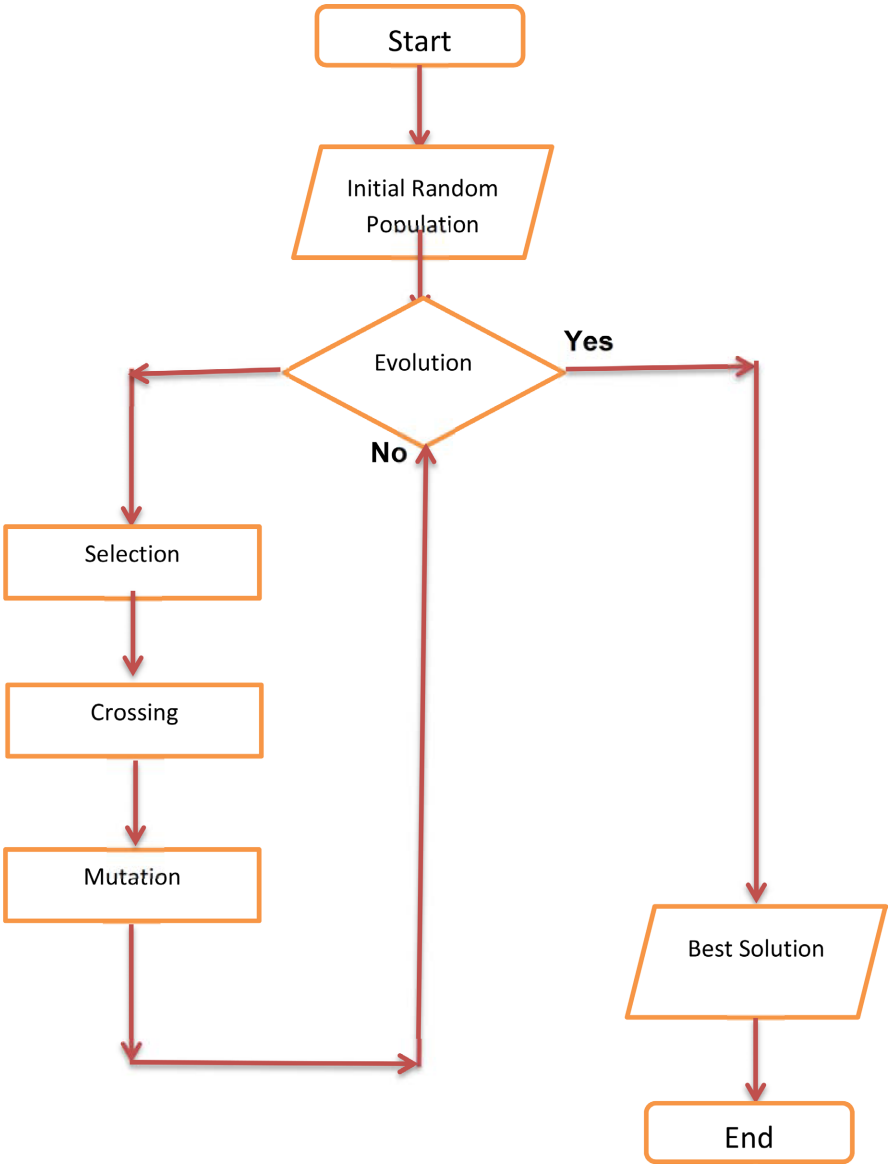

Figure 3. Genetic Algorithm flowchart [14].

```
best_solutions = tools.selBest(population, 1)
for individual in best_solutions:
  print(individual)
  print(individual.fitness)
  for i in range(len(individual)):
    if individual[i] == 1:
      print('Name: ', names[i], ' - Price: ', prices[i])
```

Figure 4. Python code used to print Capacity Supplying Optimization result.

2) Traveling Salesman Problem

The implement of GA to get the best solution for the second problem TSP by using Python code (see **Figure 7**).

The result (see **Figure 8**) after implementing Python code (**Figure 7**), as follows:

The optimization results of the total cost prices (5202) represent the minimum

```
[1, 0, 0, 0, 1, 1, 0, 1, 0, 0, 0, 0, 1, 1]
(3933500.0,)
Name:  Refrigerator LG  - Price:  865000
Name:  AIR CONDITION LG  - Price:  900000
Name:  AIR CONDITION ALHATH  - Price:  700000
Name:  Microwave  Oven SUMSUNG  - Price:  110000
Name:  cloth dryer ALHATH  - Price:  658500
Name:  cloth dryer LG  - Price:  700000
```

Figure 5. Display the result of Capacity Supplying Optimization, where the first line (vector) represent the operation of choosing (1) or not choosing (0) the devices, and the second line represent the total number of prices of each chosen device.

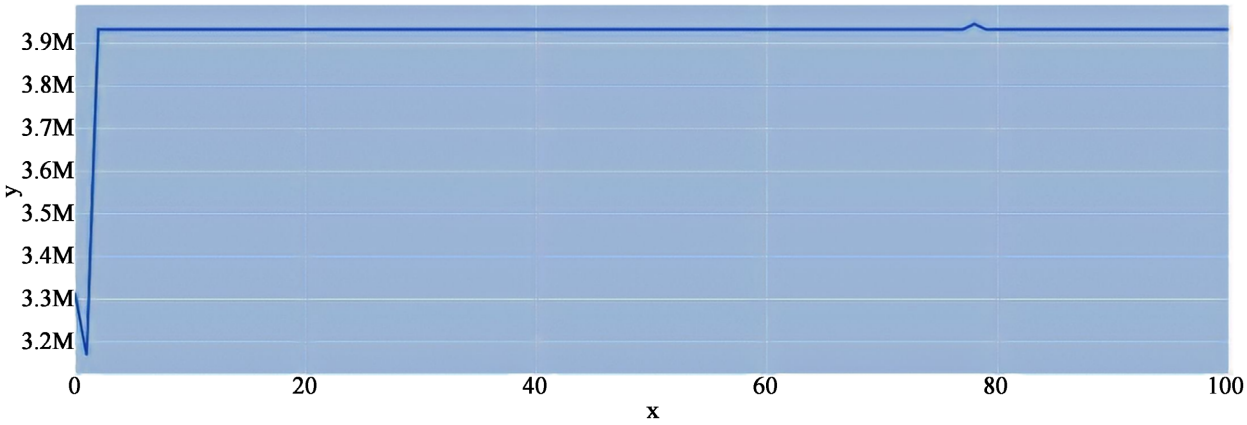

Figure 6. Represent the optimization results of the prices (3,933,500) represent the maximum values that could we get using GA.

```
best_solution = tools.selBest(population, 1)
for individual in best_solution:
  print(individual)
  print(individual.fitness)
```

```
print_schedule(individual)
```

Figure 7. Python code used to print Traveling Salesman Optimization result.

```
[ ]
        Anbar      ANB  6:08- 8:06 184  6:58- 9:01 122
        Basra      BAS  6:11- 8:31 814 10:00-15:00 500
        Mosul      MOS  6:05- 8:32 675  6:03- 8:43 400
        Erbil      ERB  6:25- 9:30 573  6:33- 9:14 395
       Ansari      ANS  6:17- 8:26 516  6:39- 8:09 345
       Kirkuk      KIR  6:12-10:22 408  6:09- 9:49 270
Total price: 5202
```

Figure 8. Display the result of Traveling Salesman Problem, which determine the total price (of each minimum prices of each traveling from Baghdad to other cities: Anbar, Basra, Mosul, Erbil, Ansari, and Kirkuk).

values that could we get using GA of TSP for each traveling from Baghdad to other cities: Anbar, Basra, Mosul, Erbil, Ansari, and Kirkuk (see **Figure 9**).

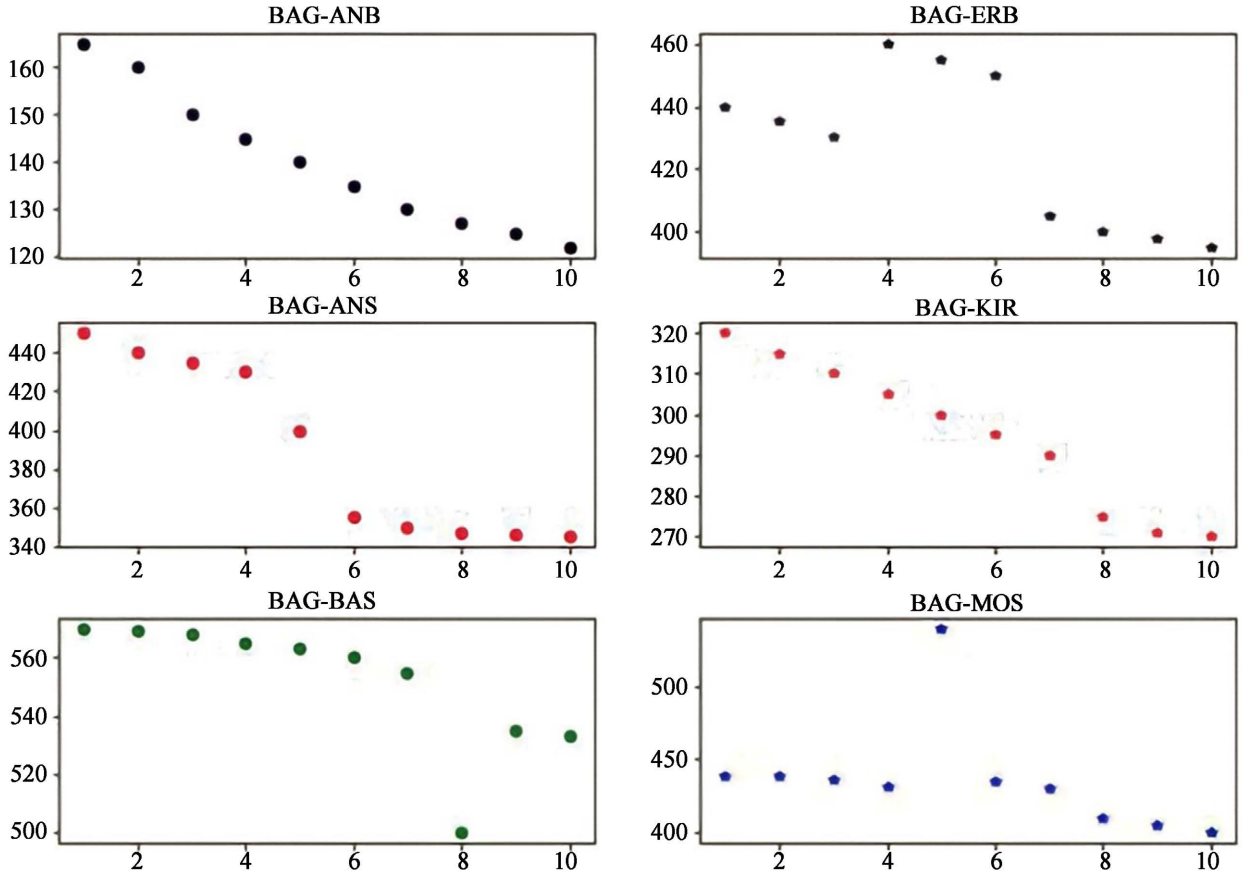

Figure 9. Display the optimization results of the total cost prices (5202) represent the minimum values that could we get using GA of TSP for each traveling from Baghdad (BAG) to other cities: Anbar (ANB), Basra (BAS), Mosul (MOS), Erbil (ERB), Ansari (ANS), and Kirkuk (KIR), where the x-axes represent the number of iterations and y-axes represent the cost for traveling.

5. Conclusions

From the above results we can conclude the following:

1) Genetic Algorithms proved best areas of solutions for real world problems.

2) Genetic Algorithms are adaptive to their environments, as this type of method is a platform appearing in the changing environment.

3) GA used to solve two nested problems.

Conflicts of Interest

The authors declare no conflicts of interest regarding the publication of this paper.

References

[1] Sourabh, K., Singh, C.S. and Vijay, K. (2021) A Review on Genetic Algorithm: Past, Present, and Future. **80**, 8091-8126. https://doi.org/10.1007/s11042-020-10139-6

[2] Sefiane, S. and Benbouziane, M. (2012) Portfolio Selection Using Genetic Algorithm. *Journal of Applied Finance & Banking*, **2**, 143-154. https://www.scienpress.com/Upload/JAFB/Vol%202_4_9.pdf

[3] Zhang, J.Y., Qi, P.Y. and Wang, J.K. (2023) Multi-Objective Genetic Algorithm for Synchrotron Radiation Beamline Optimization. *Journal of Synchrotron Radiation*, **30**, 51-56. https://doi.org/10.1107/S1600577522010050

[4] Haldurai, L., Madhubala, T. and Rajalakshmi, R. (2016) A Study on Genetic Algorithm and Its Applications. *International Journal of Computer Sciences and Engineering*, **4**, 139-143. https://www.academia.edu/29744734/A_Study_on_Genetic_Algorithm_and_its_Applications

[5] Abdulhafidh, I.A. and Anmar, D.L. (2019) Design & Implementation of an Optimization Loading System in Electric by Using Genetic Algorithm. *Journal of Computer and Communications*, **7**, 135-146. https://doi.org/10.4236/jcc.2019.77013

[6] Michael, H. and Kurt, H. (2007) TSP-Infrastructure for the Traveling Salesperson Problem. *Journal of Statistical Software*, **23**, 1-21. https://doi.org/10.18637/jss.v023.i02

[7] Darrell, W. (1994) A Genetic Algorithm Tutorial. *Statistics and Computing*, **4**, 65-85. https://doi.org/10.1007/BF00175354 https://sci2s.ugr.es/sites/default/files/files/linksInterest/Tutorials/Whitley94.pdf

[8] https://lg-iq.com

[9] https://www.google.com/search?q=alhafidh&oq=alhafidh&aqs=chrome..69i57j0i512j0i10i512j0i512j0i30l5.1397j0j15&sourceid=chrome&ie=UTF-8

[10] http://www.royalcoegypt.com/

[11] https://www.samsung.com/us/

[12] https://maps.app.goo.gl/MkTUxhTBcuxGbNaX8?g_st=it

[13] García-Altamirano, Juan Carlos, Mika Olsen, Jorge Cervantes-Ojeda (2023) How to Construct the Symmetric Cycle of Length 5 Using Hajós Construction with an Adapted Rank Genetic Algorithm. https://doi.org/10.46298/dmtcs.10189

[14] Ivan, G. (2021) Learning Genetic Algorithms with Python. 1st Edition, BPB Publications, India. https://ebin.pub/learning-genetic-algorithms-with-python-empower-the-performance-of-machine-learning-and-ai-models-with-the-capabilities-of-a-powerful-search-algorithm-english-edition-8194837758-9788194837756.html