Since January 2020 Elsevier has created a COVID-19 resource centre with free information in English and Mandarin on the novel coronavirus COVID-19. The COVID-19 resource centre is hosted on Elsevier Connect, the company's public news and information website.

Elsevier hereby grants permission to make all its COVID-19-related research that is available on the COVID-19 resource centre - including this research content - immediately available in PubMed Central and other publicly funded repositories, such as the WHO COVID database with rights for unrestricted research re-use and analyses in any form or by any means with acknowledgement of the original source. These permissions are granted for free by Elsevier for as long as the COVID-19 resource centre remains active.
Scheduling shuttle ambulance vehicles for COVID-19 quarantine cases, a multi-objective multiple 0–1 knapsack model with a novel Discrete Binary Gaining-Sharing knowledge-based optimization algorithm

Data Science for COVID-19. https://doi.org/10.1016/B978-0-12-824536-1.00034-4
Copyright © 2021 Elsevier Inc. All rights reserved.
The problem of the large number of people who feel symptoms of coronavirus infection has worsened in the recent period, which necessitates appropriate planning for the speedy transfer of these cases to quarantine hospitals. This is to be done with the best possible efficiency in the utilization of time while considering also the health and morbidity factor of those people. Patient Transport Service have a key role in ensuring patients who are frail or vulnerable or have no other means of transport get to their health appointments safely and on time. Patient transport is a service that transfers patients to and from medical facilities, and ambulances are used to respond to emergency medical services. They can rapidly transport paramedics and other first responders to the scene, carry equipment for administering emergency care, and transport patients to hospital or other definitive care [5].

The optimal planning of the ambulance facility and the scheduling of its movements back and forth to transport suspects with the new Coronavirus are of great importance. This should be considered at the level of specialists and researchers for the optimal utilization of this vital facility.

The optimal scheduling of shuttle trips to transport COVID-19 patients to isolation hospitals when receiving calls from people at risk of infection with the virus, especially those in contact with patients whose positive tests have been confirmed can save lives. It can prevent contacting of those who might be carrying the virus with their families and relatives where the infection can transfer to more people.

This research is concerned to the optimal scheduling of ambulances vehicle trips so that the optimal use of the available time is achieved, and the largest number of suspects patients is transported. At the same time, priority is given to prospected patients according to their health condition so that more weight is given to those who are in more dangerous status than others. All the prospected patients are transported to the isolation hospitals to conduct the necessary analyses and treatment.

The structure of the paper is as follows: the second section explains the problem of scheduling shuttle ambulance vehicles (SSAVs) for COVID-19 patients. It highlights the efforts done by the ambulance specialists as battlefield partners who are indifferent to the seriousness and obstacles they face. It reveals also the current plan of the ambulance facility to perform their job starting from receiving people calls, preparing and prioritize patient list, schedule ambulance vehicles, and then transport patients to the isolation hospitals. Finally, this section comes to the conclusion for the necessity of using scientific methods to ensure the best scheduling of the ambulances vehicle to do the work in the best manner, and that the statement of this problem resembles the well-known Multi-objective Multiple 0–1 Knapsack Problem.

Section 3 presents a concise review of the knapsack problem (KP). Important related versions of this famous problem are also mentioned like the 0–1 KP, Multi-objective Knapsack Problem (MOKP), and the Multi-objective Multiple Knapsack Problem (MOMKP).

The mathematical representation for scheduling the shuttle ambulance vehicles is designed in Section 4 including all needed formulations. The proposed model is a Binary Constrained MOMKP; the steps of the solution procedure are also explained.
An illustrative case study in Cairo, Egypt, with all relative data is explained in Section 5. Its complete mathematical model is formulated.

In Section 6, a novel Discrete Binary version of a recently developed gaining-sharing knowledge-based technique (GSK) is introduced for solving the (SSAV). GSK cannot solve the problem with discrete binary space; therefore, Discrete Binary-GSK optimization algorithm (DBGSK) is proposed with two new discrete binary junior and senior stages. These stages allow DBGSK to inspect the problem search space efficiently.

Section 7 represents the experimental results of (SSAV) obtained by DBGSK, and Sections 8 gives the conclusions and the suggested points for future researches.

2. Scheduling shuttle ambulance for COVID-19 patients

Suddenly, as a part of this world, we found ourselves face to face in front of a battle with COVID-19, the virus that necessitated tackling it, honing all medical efforts, and raising the level of preparedness in all sectors of health concerned. Doctors and nursing make extraordinary efforts, but we must not forget the link among them and patients, they are the ambulance specialists who take nonstop daily trips to and from homes to isolation hospitals. They are battlefield partners who are indifferent to the seriousness and obstacles they face.

The workers in the ambulance system make great sacrifices; they work in very difficult conditions and play an important pivotal role in facing the emerging crisis of Coronavirus to prevent the spread of this epidemic. All of them are heroes at the forefront of the first line in the face of this dreaded global epidemic that threatens all countries. The ambulance facility is the unknown soldier facing Coronavirus, where the personnel make a great effort and play a heroic role, and they deserve our support and honor.

The telephone communication rates at the ambulance facility have increased significantly during this period, as the number of phone calls that the ambulance agency receives is about 2500 calls per day from the governorates of Greater Cairo. The ambulance facility transmits 60 to 170 daily cases of suspicion from homes to hospitals and ambulance workers work around the clock to transfer suspected people with Coronavirus to places of isolation and medical examination.

The current plan of the ambulance facility is to maintain an enough ambulance equipped with their crew on standby. They are ready to deal with any Corona situation that appears in the province, whether positive or suspected. The authority has also an alternative scenario to prepare other cars for each governorate in case the conditions change for the worst.

There are enough drivers and paramedics, all of whom receive continuous training. It is not possible to use a paramedic for real-time interaction before he goes through entire practical training through simulated experiments.

The ambulance operations room receives the reports through a contracting company, which sends the communication in writing to appear on the screen with details of the
case, name, address, and all relevant data. The operations room employee immediately begins dealing with it and calls the ambulance to the location of the caller. He continues to follow the case to make sure it arrives to the hospital. Data are recorded on-screen to calculate the report receipt time, response time, and status arrival.

The ambulance specialists wear the uniforms assigned to them in such cases. During the isolation trip, they are busy with the patient following for oxygen, especially if he suffers from shortness of breath, as well as to the heartbeat through screens and devices. They are also in constant contact with the ambulance facility and recipients for any change in instructions or trip details.

The geographical scope of ambulance work is divided into sectors, where a hospital is designated for medical isolation of the suspected cases and several ambulances are allocated for corona only. The operating room receives the reports, and a schedule is prepared according to the geographical location and the condition of the patient [6]. Ambulance drives in shuttle movement back and forth from the isolation hospital to the places of suspected COVID-19 infection, Fig. 37.1.

Here, scientific methods must be used to ensure the best scheduling of ambulances. The ultimate objective is to carry out the evacuation duties by making the best use of available time. Two competing objectives arise: transporting the largest number of suspects patients, while also considering the priority of cases according to their health status. This statement of this problem resembles the well-known Multi-objective Multiple 0–1 Knapsack Problem.

3. Multi-objective Multiple Knapsack Problem: an overview

The KP is a fundamental problem in combinatorial optimization. The KP involves the maximization or minimization of a value, such as profits or costs. A knapsack can only hold a certain weight or volume that can accommodate different types of items but with

![FIGURE 37.1 Scheme of shuttle ambulance vehicle.](image-url)
limitation in total volume, weights, or both. The question is to decide the number of each item to pick-up so that to minimize the total cost or to maximize the total profit [7]. In the Integer type KP, items can have arbitrary integer values, whereas the 0–1 KP limits the number of each type to 0 or 1. The 0–1 KP does not allow the user to put multiple copies of the same items in their knapsack. The 0–1 knapsack is a special case in which each input can be loaded or not into the knapsack.

There are other variants of KP like the Multi-dimensional 0–1 KP [8–11] or the Multiple 0–1 Knapsack Problem [12–14]. The integer KP and the 0–1 KP can be solved using some designed dynamic programming algorithms by deriving a recurrence equation expressing a solution to an instance in terms of solutions to its small-scale instances [15]. Dynamic Programming follows the principle of optimality to reach the final solution [16]. The 0–1 KP and its versions are famous NP-hard problems; dynamic programming technique can solve such problems in pseudo-polynomial time [17].

In combinatorial optimization problems like that of KP, exact methods are impractical in finding an optimal solution because the run time is exponentially increased as the problem size increases [18]. Therefore, interest in the application of the metaheuristic algorithms has become necessary to solve these problems and obtain the results in a reasonable time [19–22].

The 0–1 KP can be solved also by greedy genetic algorithm (GA), GA, and rough set theory and ant weight-lifting algorithm [23].

One interesting generalization of KP is the MKP, where a set of items and a set of knapsacks are given [24]. Each item has a size and a profit value, and each knapsack has a capacity. The goal of the MKP is to find a subset of maximum total profit such that the picked items can be packed into the knapsacks without exceeding their capacities [25–27].

It is well-known that the MKP is NP-hard. Therefore, it is reasonable to design an approximation algorithm to find a feasible solution for the MKP [28]. Various heuristic and exact methods have been devised to solve it. GA shows good performance on solving static optimization problem [29].

In MOKP, a major challenge is to generate efficient solutions that have the property that no improvement on any objective is possible without sacrificing on at least another objective. A survey and an annotated bibliography about multi-objective combinatorial optimization (MOCO) can be found in Ref. [30]. Many real-world applications are reported in the multi-objective case dealing with capital budgeting, selection of transportation investment alternatives, relocation issues arising in conservation biology, and planning remediation of contaminated lightstation sites [31]. Several exact approaches have been proposed in the literature to find the efficient set especially designed for the bi-objective case [32–34]. Besides exact methods investigated in this paper, approximation algorithms [35] and metaheuristics [36–38] have been proposed. Many evolutionary algorithms (EAs) for solving multi-objective 0/1 KP are proposed [39].

In the case of multi-objective EAs, the outcome is usually an approximation of the Pareto-optimal set, which is denoted as an approximation set, [35,40,41].
The MOMKP combines the two cases of MOKP and the multiple knapsack problem (MKP). Many approximate solution methods have been developed to solve MOMKP [42]. EAs have been successfully applied to MOCO problems in the last decades. EAs maintain a population of solutions, and thus they can obtain multiple efficient solutions in a single run [43].

4. Mathematical model for scheduling the shuttle ambulance vehicles

The Shuttle Ambulance Scheduling (SAS) is defined as follows:

- Each patient place is visited only once by only one ambulance vehicle.
- The shuttle ambulance performs several shuttle movements for transferring each patient from his place to the isolation hospital, the process is performed back and forth repeatedly limited by the time the scheduling time slot.
- The overall goal of the problem to be achieved is the best utilization of the available time expressed by two competing objectives: maximizing the number of evacuated patients and maximizing the efficiency through prioritizing dangerous cases.

4.1 Mathematical model

The model will be formulated as multi-objective multiple 0–1 knapsack problem as follows:

Decision Variables:
Let:
\[ x_{ij} = \begin{cases} 1, & \text{if patient } i \text{ is approached by the Ambulance car number } m, i=1,2,\ldots,m \text{ and } j=1,2,\ldots,n. \\ 0, & \text{otherwise.} \end{cases} \]

4.2 Constraints

(1) Patients constraints:
Each patient \( i \) is either approached by only one ambulance vehicle or not picked up in the considered time slot:
\[ \sum_{j=1}^{n} x_{ij} \leq 1, \quad i = 1, 2, \ldots, m. \]  

(2) Time slot constraints:
Time slot \( T \) is the time spent planning and scheduling movements of the ambulance vehicle according to received calls. The total time spent by the shuttle ambulance in all
transportation segments in the current time segment $t$ consists of traveling, riding patient, and handing him over should be within the available slot time $T$.

\[
2 \sum_{i=1}^{m} t_{0i} x_i^j + \sum_{i=1}^{m} t_i^j x_i^j + \sum_{i=1}^{m} t_i^h x_i^j \leq T, \quad j = 1, 2, \ldots, n
\]  

(37.2)

where:
- $t_{0i}$ = Transportation time between the isolation hospital and patient place $i$, $i = 1, 2, \ldots, n$.
- $t_i^j$ = Riding time patient $i$ to the ambulance vehicle, $i = 1, 2, \ldots, n$.
- $t_i^h$ = Handing over time patient $i$ to the hospital, $i = 1, 2, \ldots, n$.
- $T$ = Time slot.

The first part is for traveling from the isolation hospital (denoted by 0) to all the picked-up patient’ places forth and back, the second part is the sum of riding time of patients to the ambulance, and the third part is the sum of handing over times of patient at the hospital.

(3) Binary constraints:

All the decision variables are $0\rightarrow1$.

\[
x_i^j = 0 \text{ or } 1, \quad i = 1, 2, \ldots, m \text{ and } j = 1, 2, \ldots, n.
\]  

(37.3)

(4) Avoid the trivial solution

To avoid the trivial solution that all patients are assigned to one ambulance vehicle, the following condition should hold:

The traveling from the isolation hospital to all the picked-up patient’ places forth and back, the sum of riding time of patients to the ambulance and the sum of handing over times of patient at the hospital should be greater than the considered scheduled time slot. This condition should be checked before writing the mathematical model:

\[
\text{Total transportation time + Total riding time of patients to the ambulance +}
\text{+ Total handing over to the hospital} > T
\]  

(37.4)

In case of violation, then all patients can be transported in one vehicle and no need to perform scheduling process. In this case, priority should be given to cases with severe health conditions.

(5) The objective functions

The COVID-19 crisis operation room decided on two main competing objectives, the first is to maximize the total number of evacuated patients, and the second one is to give priorities depending on the patient’s conditions in terms of his age, health, and number of contacts who are close to him.
The problem is then a multi-objective one with two objectives. The Multi-Objective Optimization (MOO) refers to finding the optimal solution values of more than one desired objective. The motivation in MOO is that it allows for a compromise (trade-off) on some contradictory issues. There is a vector of the objective function in a MOO; there is no single best solution for all purposes, but rather several solutions.

The weighted sum or scalarization method is one of the classic (MOO) methods; it puts a set of objectives into one by adding each objective premultiplied by a user-supplied weight. The weight of an objective is chosen in proportion to the relative importance of the objective. A large weight that is given to an objective function shows that said function has a higher priority compared to the ones with a smaller weight.

The weighted sum method is simple, but it is difficult to set the weight vectors to obtain a Pareto-optimal solution in a desired region in the objective space, and it cannot find certain solutions in case of a nonconvex objective space. The predominant concept in defining an optimal point is that of Pareto optimality [44].

In the scalarization method, answer is a set of solutions that define the best trade-off between competing objectives that form in its entirety the nondominated Pareto-optimal set for the problem. The boundary defined by the set of all point mapped from the Pareto-optimal set is the Pareto-optimal front [45].

The weighted sum approach treats the MOO as composite objective function [46]. The composite objective function is expressed as follows:

$$Z = \text{Maximize} \sum_{i=1}^{q} w_i \cdot f_i(x)$$

where $w_i$ is the positive weight values, $f_i(x)$ is one of the objective functions and $q$ is the number of objective functions. Maximizing $Z$ will provide an enough condition for optimal multiobjective solution to be found. Since the objective of this research is to provide a compromise between maximizing of Total Relative Importance (TRI) and maximizing of Total Number of Patients (TNP), and the following composite objective functions are considered:

$$Z = w_1 f_1(x) + w_2 f_2(x)$$  \(\text{(a)}\)

Maximize $f_1(x) = \text{(TRI)} = \sum_{i=1}^{n} p_i \sum_{m=1}^{n} x_{ij}^i$, and  \(\text{(b)}\)

Maximize $f_2(x) = \text{(TNP)} = \sum_{i=1}^{n} \sum_{m=1}^{n} x_{ij}^i$  \(\text{(c)}\)

The weights $w_1$ and $w_2$ are related based on the following expression:

$w_2 = 1 - w_1$, $w_1$ is chosen in the range of [0–1],

where $p_i$ is the relative importance of patient No $i$. 
Considering that the available medical equipment of ambulance vehicles is different, then a variation of priorities in riding up ambulance vehicles will vary, and this will be considered by giving higher weights to better vehicles. From (a, b, and c), and the composite objective function will be:

\[
\text{Maximize } Z = w_1 \left( \sum_{i=1}^{n} p_i^{p} \sum_{j=1}^{n} p_j^{a} x_{ij} \right) + w_2 \left( \sum_{i=1}^{n} \sum_{j=1}^{n} x_{ij} \right)
\]  

(37.5)

where \( p_i^{p} \) = Priority of patient number \( i \) based on his medical status.

\( p_j^{a} \) = Priority assigned to ambulance vehicle number \( j \), more priority is given to more equipped vehicles.

To assign the proper weightage values in Eq. (37.5), an investigation is carried out by varying the value \( w_1 \) from 1 to 0 with a step size of 0.1. This is done to observe the significance of each weightage set toward the objective function [47].

Finally, we have a suggested model that contains \((m, n)\) binary variables and \((m + n)\) constraints.

Any final chosen efficient solution will produce two distinct situations:

**1.** If all \( x_{ij} \), \( \forall i \) and \( j \), then all patients are transported to the hospital during the planned time slot, this problem is completed, and the planning team will go to the following time slot.

**2.** If \( \sum_{j=1}^{n} x_{ij} = 0 \) for any \( i \), then the corresponding patient \( i \) is not transported. In this case, add this patient to the updated list and repeat the procedure for another time slot.

The solution procedure is presented in Fig. 37.2.

## 5. An illustrated case study

The illustrated case study will take place in Cairo, Egypt, Fig. 37.2. During a given time slot of 120 min, the ambulance operation room received 10 calls from suspicious patients to be transported to the isolation hospital. Table 37.1 and Fig. 37.3 show names and positions of the locations of different patients. Time for riding a patient to the ambulance vehicle, \( t_i^r \) and time of handing over patient to the isolation hospital \( t_i^h \) are estimated to be equal for all patients to be as follows: \( t_i^r = 8 \) min and \( t_i^h = 2 \) min.

After considering the opinion of a group of specialists practicing this work in these sites for many years, Table 37.2 shows the time required to travel from each site to another and the extension time required for each local unit.

The mathematical formulation for the given case is worked out by substituting in the previously described model for the given problem (Fig. 37.4).
Table 37.1 Patient's locations in Cairo map.

| Patient No | Position                                           |
|------------|----------------------------------------------------|
| 1          | Merry land Park                                   |
| 2          | El-Galaa Bridge                                   |
| 3          | Al Hayy Ath Thamin, Nasr City                     |
| 4          | Port Saeed and Sekat Al Waili Streets Intersection |
| 5          | Salah El Din Al Ayoubi Citadel                    |
| 6          | Shobra                                             |
| 7          | Al Sabtiah                                         |
| 8          | Cairo Opera House                                 |
| 9          | Emtidad Ahmed Helmy                                |
| 10         | Al Teraa Al Bolakia                               |
6. Proposed methodology

Metaheuristic approaches are developed for the complex optimization problems with continuous variables. Mohamed et al. [48] recently proposed a novel Gaining-Sharing Knowledge-based optimization algorithm (GSK), setup on acquiring knowledge and share it with others throughout their lifetime. The original GSK solves optimization problems over continuous space, but it cannot solve the problem with binary space. So, a new variant of GSK is introduced to solve the proposed (SSAV). A novel DBGSK is proposed over discrete binary space with new binary junior and senior gaining and sharing stages.
On the other hand, there are many constraint handling techniques in the literature [49–51]. In their work, the augmented Lagrangian method is used in which an unconstrained optimization problem is obtained from a constrained optimization one [52,53]. The proposed methodology is described below:

6.1 Gaining sharing-knowledge-based optimization algorithm

An optimization problem with constraints is worked out as:

\[
\begin{align*}
\text{Min} & \quad f(X); \quad X = [x_1, x_2, \ldots, x_{\text{Dim}}] \\
\text{s.to.} & \quad g_i(X) \leq 0; \quad i = 1, 2, \ldots, m \\
& \quad X \in [\alpha_p, \beta_p]; \quad p = 1, 2, \ldots, \text{Dim}
\end{align*}
\]

where \( f \) denotes the objective function; \( X = [x_1, x_2, \ldots, x_{\text{Dim}}] \) are the decision variables; \( g_i(X) \) are the inequality constraints; and \( \alpha_p, \beta_p \) are the lower and upper bounds of decision variables, respectively, and \( \text{Dim} \) represents the dimension of individuals. If the objective function is in maximization form, then consider minimization = - maximization.

The human-based algorithm GSK is of two-stages: junior and senior gaining and sharing stage. All persons acquire knowledge and share their views with others. The people from early-stage gain knowledge from their small networks such as family members, relatives, neighbors, etc., and want to share their opinions with the others who might not be from their networks because of curiosity of exploring others. These may not have the experience to categorize the people. In the same way, the people from the middle or later age enhance their knowledge by interacting with friends, colleagues, social media friends, etc., and share their views with the most suitable person, so that they can improve their knowledge. These people have the experience to judge other
people and can categorize them (good or bad). The process mentioned above can be formulated mathematically in the following steps:

**Step 1:** To get a starting point of the optimization problem, the initial population must be obtained. The initial population is created randomly within the boundary constraints as:

\[ x_{tp}^0 = \alpha_p + \text{rand}_p(\beta_p - \alpha_p) \]  

(37.6)

where \( t \) is for the number of populations; \( \text{rand}_p \) denotes random number uniformly distributed in 0–1 range.

**Step 2:** At this step, the dimensions of junior and senior stages should be computed through the following formula:

\[ \text{Dim}_j = \text{Dim} \times \left( \frac{\text{Gen}_{\max} - G}{\text{Gen}_{\max}} \right)^k \]  

(37.7)

\[ \text{Dim}_s = \text{Dim} - \text{Dim}_j \]  

(37.8)

where \( k (> 0) \) denotes the learning rate that monitors the experience rate. \( \text{Dim}_j \) and \( \text{Dim}_s \) represent the dimension for the junior and senior stage, respectively. \( \text{Gen}_{\max} \) is the maximum count of generations, and \( G \) is the count of generation.

**Step 3:** Junior gaining sharing knowledge stage: In this stage, the early aged people gain knowledge from their small networks and share their views with the other people who may or may not belong to their group. Thus, individuals are updated through as:

i. According to the objective function values, the individuals are arranged in ascending order. For every \( x_t (t = 1, 2, \ldots, \text{NP}) \), select the nearest best \((x_{t-1})\) and worst \((x_{t+1})\) to gain knowledge, also choose randomly \((x_r)\) to share knowledge. Therefore, to update the individuals, the pseudo-code is presented in Fig. 37.5.

```plaintext
for t=1:NP
  for p=1:Dim
    if rand <= k_r (知识比例)
      if f(x_t) > f(x_r)
        \[ x_{tp}^{new} = \left( x_t + k_f \ast ((x_{t-1} - x_{t+1}) + (x_r - x_t)) \right) \]
      else
        \[ x_{tp}^{new} = \left( x_t + k_f \ast ((x_{t-1} - x_{t+1}) + (x_t - x_r)) \right) \]
      end
    else
      \[ x_{tp}^{new} = x_{tp}^{old} \]
    end
  end
end
```

**FIGURE 37.5** Pseudo-code for Junior gaining sharing knowledge stage.
where $k_f (> 0)$ is the knowledge factor.

Step 4: Senior gaining sharing knowledge stage: This stage comprises the impact and effect of other people (good or bad) on the individual. The updated individual can be determined as follows:

i. The individuals are classified into three categories (best, middle, and worst) after sorting individuals into ascending order (based on the objective function values).

Best individual = $100 \, p\% \, (x_{\text{best}})$, middle individual = $\text{Dim} - 2 \times 100 \, p\% \, (x_{\text{middle}})$, worst individual = $100 \, p\% \, (x_{\text{worst}})$

For every individual $x_t$, choose the top and bottom 100 $p\%$ individuals for gaining part and the third one (middle individual) is chosen for the sharing part. Therefore, the new individual is updated through the following pseudo-code dictated in Fig. 37.6, where $p \in [0, 1]$ is the percentage of best and worst classes.

### 6.2 Discrete Binary Gaining-Sharing knowledge-based optimization algorithm

To solve problems in discrete binary space, a novel DBGSK is suggested. In DBGSK, the new initialization and the working mechanism of both stages (junior and senior gaining sharing stages) are introduced over discrete binary space, and the remaining algorithms remain the same as the previous one. The working mechanism of DBGSK is described in the upcoming subsections:

#### 6.2.1 Discrete binary initialization

A first population is obtained in GSK using Eq. (37.6), and it must be updated using the following equation for binary population:

$$x_{t_p}^0 = \text{round}(\text{rand}(0, 1))$$  \hspace{1cm} (37.9)
where the round operator is used to convert the decimal number into the nearest binary number.

6.2.2 Discrete binary junior gaining and sharing stage

This stage is based on the original GSK with \( k_f = 1 \). The individuals are updated in original GSK using the pseudo-code (Fig. 37.6) which contains two cases. These two cases are defined for discrete binary-stage as follows:

**Case 1: When** \( f(x_r) < f(x_t) \): There are three different vectors \( (x_{t-1}, x_{t+1}, x_r) \), which can take only two values (0 and 1). Therefore, a total of \( 2^3 \) combinations are possible, which are listed in Table 37.3. Furthermore, these eight combinations can be categorized into two different subcases [(a) and (b)] and each subcase has four combinations. The results of each possible combination are presented in Table 37.3.

- **Subcase (a):** If \( x_{t-1} \) is equal to \( x_{t+1} \), the result is equal to \( x_r \).
- **Subcase (b):** When \( x_{t-1} \) is not equal to \( x_{t+1} \), then the result is the same as \( x_{t-1} \) by taking \(-1\) as 0 and 2 as 1.

The mathematical formulation of Case 1 is as follows:

\[
x_{\text{new}} = \begin{cases} 
  x_r; & \text{if } x_{t-1} = x_{t+1} \\
  x_{t-1}; & \text{if } x_{t-1} \neq x_{t+1}
\end{cases}
\]

**Case 2: When** \( f(x_r) \geq f(x_t) \): There are four different vectors \( (x_{t-1}, x_t, x_{t+1}, x_r) \) that consider only two values (0 and 1). Thus, there are \( 2^4 \) possible combinations that are presented in Table 37.4. Moreover, the 16 combinations can be divided into two subcases [(c) and (d)] in which (c) and (d) has four and 12 combinations, respectively.

- **Subcase (c):** If \( x_{t-1} \) is not equal to \( x_{t+1} \), but \( x_{t+1} \) is equal to \( x_r \), the result is equal to \( x_{t-1} \).

- **Subcase (d):** If any of the condition arises \( x_{t-1} = x_{t+1} \neq x_r \) or \( x_{t-1} = x_{t+1} \neq x_r \) or \( x_{t-1} = x_{t+1} = x_r \), the result is equal to \( x_t \) by considering \(-1\) and \(-2\) as 0, and 2 and 3 as 1.

### Table 37.3 Results of the discrete binary junior gaining and sharing stage of Case 1, \( k_f = 1 \).

| \( x_{t-1} \) | \( x_{t+1} \) | \( x_r \) | Results | Modified results |
|---|---|---|---|---|
| Subcase (a) | 0 | 0 | 0 | 0 |
| | 0 | 0 | 1 | 1 |
| | 1 | 1 | 0 | 0 |
| | 1 | 1 | 1 | 1 |
| Subcase (b) | 1 | 0 | 0 | 1 |
| | 1 | 0 | 1 | 2 |
| | 0 | 1 | 0 | -1 |
| | 0 | 1 | 1 | 0 |
The mathematical formulation of Case 2 is as follows:

\[
x_{\text{new}} = \begin{cases} 
  x_{t-1} & ; \text{if } x_{t-1} \neq x_{t+1} = x_r \\
  x_t & ; \text{Otherwise}
\end{cases}
\]

### 6.2.3 Discrete binary senior gaining and sharing stage

The working mechanism of discrete binary senior gaining and sharing stage is the same as the binary junior gaining and sharing stage with value of \( k_f = 1 \). The individuals are updated in the original senior gaining sharing stage using pseudo-code (Fig. 37.7) that contains two cases. The two cases further modified for binary senior gaining sharing stage in the following manner:

**Case 1:** \( f(x_{\text{middle}}) < f(x_t) \): It contains three different vectors (\( x_{\text{best}}, x_{\text{middle}}, x_{\text{worst}} \)), and they can assume only binary values (0 and 1), thus total eight combinations are possible to update the individuals. These total eight combinations can be classified into two subcases [(a) and (b)] and each subcase contains only four different combinations. The obtained results of this case are presented in Table 37.5.

**Subcase (a):** If \( x_{\text{best}} \) is equal to \( x_{\text{worst}} \), then the obtained results are equal to \( x_{\text{middle}} \).

**Subcase (b):** On the other hand, if \( x_{\text{best}} \) is not equal to \( x_{\text{worst}} \), then the results are equal to \( x_{\text{best}} \) with assuming \(-1\) or \(2\) equivalent to their nearest binary value (0 and 1, respectively).

Case 1 can be mathematically formulated in the following way:

\[
x_{\text{new}} = \begin{cases} 
  x_{\text{middle}} & ; \text{if } x_{\text{best}} = x_{\text{worst}} \\
  x_{\text{best}} & ; \text{if } x_{\text{best}} \neq x_{\text{worst}}
\end{cases}
\]
Case 2: \( f(x_{\text{middle}}) > f(x_t) \): It consists of four different binary vectors 
\((x_{\text{best}}, x_{\text{middle}}, x_{\text{worst}}, x_t)\), and with the values of each vector, a total of 16 combinations 
are presented. The 16 combinations are also divided into two subcases [(c) and (d)]. The 
subcases (c) and (d) further contain four and 12 combinations, respectively. The sub-
cases are explained in detail in Table 37.6.

Subcase (c): When \( x_{\text{best}} \) is not equal to \( x_{\text{worst}} \) and \( x_{\text{worst}} \) is equal to \( x_{\text{middle}} \), then the 
obtained results are equal to \( x_{\text{best}} \).

Subcase (d): If any case arises other than (c), then the obtained results is equal to \( x_t \) 
by taking \(-2\) and \(-1\) as 0 and 2 and 3 as 1.
The mathematical formulation of Case 2 is given as

\[
x_{\text{new}}^{ip} = \begin{cases} x_{\text{best}} ; & \text{if } x_{\text{best}} \neq x_{\text{worst}} = x_{\text{middle}} \\ x_t ; & \text{Otherwise} \end{cases}
\]

The pseudo-code of DBGSK is presented in Fig. 37.7.

7. Experimental results

The (SSAV) is handled by using the proposed novel DBGSK algorithm; the used parameters are presented in Table 37.7. DBGSK runs over personal computer Intel CoreTM i5-7200U CPU @ 2.50 GHz and 4 GB RAM and coded on MATLAB R2015a. To get the compromise or effective solutions, 30 independent runs are complete, and the

Table 37.6 Results of discrete binary senior gaining and sharing stage of Case 2 with \(k_f = 1\).

| \(x_{\text{best}}\) | \(x_t\) | \(x_{\text{worst}}\) | \(x_{\text{middle}}\) | Results | Modified results |
|-------------------|--------|-------------------|-------------------|--------|------------------|
| Subcase (c)       |        |                   |                   |        |                  |
| 1                 | 1      | 0                 | 0                 | 3      | 1                |
| 1                 | 0      | 0                 | 0                 | 1      | 1                |
| 0                 | 1      | 1                 | 1                 | 0      | 0                |
| 0                 | 0      | 1                 | 1                 | −2     | 0                |
| Subcase (d)       |        |                   |                   |        |                  |
| 0                 | 0      | 0                 | 0                 | 0      | 0                |
| 0                 | 1      | 0                 | 0                 | 2      | 1                |
| 0                 | 0      | 1                 | 0                 | −1     | 0                |
| 0                 | 0      | 0                 | 1                 | −1     | 0                |
| 1                 | 0      | 1                 | 0                 | 0      | 0                |
| 1                 | 0      | 0                 | 1                 | 0      | 0                |
| 0                 | 1      | 1                 | 0                 | 1      | 1                |
| 0                 | 1      | 0                 | 1                 | 1      | 1                |
| 1                 | 1      | 1                 | 0                 | 2      | 1                |
| 1                 | 0      | 1                 | 1                 | −1     | 0                |
| 1                 | 1      | 0                 | 1                 | 2      | 1                |
| 1                 | 1      | 1                 | 1                 | 1      | 1                |

Table 37.7 Numerical values of parameters.

| Parameters of DBGSK | Considered values |
|---------------------|------------------|
| NP                  | 200              |
| K                   | 10               |
| \(k_r\)             | 0.9              |
| P                   | 0.1              |
| \(k_f\)             | 1                |
| Max number of iterations | 100              |
obtained statistics are provided in Table 37.8, including the best, median, average, worst solutions, and the standard deviations at the three groups of weights. Moreover, Fig. 37.8 shows the convergence graph of the solutions of (SSAV) using first group of the weights from the figure, it can be observed that after the 57th iteration, it converges to the global optimal solution (14.8), which shows high-convergence speed of DBGSK.

The effective solutions for the problem for different weights of the two objective functions are presented in Table 37.9. The effective solution \( x_1^9, x_1^{10}, x_2^5 \), and \( x_6^2 = 1 \) is obtained for \( w_1 \) in the range \([0.6:1]\), the effective solution \( x_9^1, x_{10}^1, x_1^2, x_2^2 \), and \( x_5^2 = 1 \) is obtained for \( w_1 \) in the range \([0.3:0.5]\) and the effective solution \( x_1^1, x_2^2, x_3^1, x_4^1, x_5^2 \), and \( x_{10}^2 = 1 \) is obtained for \( w_1 \) in the range \([0.0:0.2]\). It can be seen that as the relative weight for the objective function representing the number of patients increases (and that for the status of patients decreases), the number of patients transported to the isolation hospital increases from four to and then to six.

The shuttle routes and the total times are provided for both ambulance vehicles can be seen in Fig. 37.9. In the first effective solution, the first ambulance vehicle will transport patient number 9 and then patient number 10, while the second ambulance vehicle will transport patient number 5 and then patient number 6 with total time for each vehicle = 120 min which is by chance exactly equals to the available time slot.

| Weights  | \( f_1(x), f_2(x) \) | Best  | Median | Average | Worst  | Standard deviation |
|----------|----------------------|-------|--------|---------|--------|-------------------|
| \( w_1 \in [0.6:1] \) | (14.8,4) | (14.8,4) | (14.8,4) | (14.8,4) | (14.8,4) | 0.00 |
| \( w_1 \in [0.3:0.5] \) | (13.8,5) | (13.8,5) | (13.8,5) | (13.8,5) | (13.8,5) | 0.00 |
| \( w_1 \in [0:0.2] \) | (11.4,6) | (11.4,6) | (11.4,6) | (11.4,6) | (11.4,6) | 0.00 |
8. Conclusions and points for future researches

The main conclusions for this paper can be summarized as follows:

1. An improved scheduling the shuttle ambulance vehicles for COVID-19 patients problem is presented. The scheduling is aiming at achieving the best utilization of the available time slot, which is determined by to competing objectives of maximizing the total number of evacuated patients and considering the priorities of patients according to their health status.

2. Scheduling the shuttle ambulance vehicles for COVID-19 patients’ problem formulation looks like that of the MOMKP.

3. A binary constrained multi-objective multiple 0–1 Knapsack mathematical model is formulated for the given problem. The binary decision variables represent candidate patients to be transported to the isolation hospitals. The obtained mathematical model and the solution method for obtaining efficient solutions are used to solve an illustrated case study in Cairo, Egypt.

| Major Weight | Ambulance Number | Patient No. | Total Time (minutes) |
|--------------|------------------|-------------|----------------------|
| Health Status | 1 | 120 | 120 |
| Both Objectives | 2 | 120 | 110 |
| Number | 1 | 120 | 120 |
| | 2 | 120 | 120 |
4. The proposed problem is solved by a novel DBGSK, which involves two main stages: Discrete Binary Junior and Senior gaining and sharing stages with a knowledge factor $k_f = 1$. DBGSK is discrete binary variant of GSK that solves the problem with binary decision variables.

5. DBGSK shows that it has the ability of finding the solutions of the problem, and the obtained results demonstrate the robustness and convergence of DBGSK toward the efficient solutions.

The points for future researches can be stated in the following points:

1. To propose other mathematical models’ formulation for the same problem comprising designing the objective function, decision variables and constraints and then comparing the effectiveness of computations for each model.

2. To extend the problem to cover possible variations like the Stochastic case and other variations. The stochastic situations may include preparedness of the ambulance vehicles, transportation times, times for riding patients to the ambulance vehicles, and handing them over to the isolation hospital.

3. To build an online decision support system that can handle streaming calls to the ambulance facility, and in turn, update instantaneously the schedule of ambulance vehicles for such an important problem.

4. To check the performance of the DBGSK approach in solving different complex optimization problems, and further works can be investigated by the extension of DBGSK with different kinds of constraint handling methods.

References

[1] S. Cleemput, W. Dumon, V. Fonseca, W. Abdool Karim, M. Giovanetti, L.C. Alcantara, K. Deforche, T. Oliveira, Genome Detective Coronavirus Typing Tool for rapid identification and characterization of novel coronavirus genomes, Bioinformatics 145 (February 28, 2020). Retrieved at: https://doi.org/10.1093/bioinformatics/btaa145.

[2] The Lancet website, A Novel Coronavirus Outbreak of Global Health Concern, Retrieved on, March 30, 2020 at: https://www.thelancet.com/journals/lancet/article/PIIS0140-6736(20)30185-9/fulltext.

[3] World Health Organization website, Guidance for Health Workers, 2020. Retrieved on April 29, 2020 at: https://www.who.int/emergencies/diseases/novel-coronavirus-2019/technical-guidance/health-workers.

[4] Worldometer website, COVID-19 Coronavirus Pandemic, 2020. Retrieved on April 15, 2020 at: https://www.worldometers.info/coronavirus/#ref-13.

[5] J. Black, Ambulance Services Working Collaboratively with Community Partners, Association of Ambulance Chief Executives, 2017. https://www.kingsfund.org.uk/sites/default/files/201711/John%20Black%2031.101.17%20pdf_1.pdf.

[6] Al-Ahram Gate, 2020. Retrieved on April 21, 2020 at: http://gate.ahram.org.eg/News/2391717.aspx.

[7] B. Render, R.M. Stair Jr., Quantitative analysis for management, twelfth ed., Pearson Education India, 2016.
[8] I. Alaya, C. Solnon, K. GHeDIRA, Ant algorithm for the multi-dimensional knapsack problem, in: International Conference on Bioinspired Optimization Methods and their Applications (BIOMA 2004), 2004.

[9] S. Fidanova, Ant Colony Optimization for Multiple Knapsack Problem and Heuristic Model, 2004.

[10] S. Fidanova, Ant colony optimization for multiple knapsack problem and model bias, in: International Conference on Numerical Analysis and its Applications, Springer, Berlin, Heidelberg, June 29, 2004, pp. 280–287.

[11] S. Fidanova, Probabilistic model of ant colony optimization for multiple knapsack problem, in: International Conference on Large-Scale Scientific Computing 2007 June 5, Springer, Berlin, Heidelberg, 2008, pp. 545–552.

[12] L. Ke, Z. Feng, Z. Ren, X. Wei, An ant colony optimization approach for the multi-dimensional knapsack problem, J. Heuristics 16 (1) (2010) 65–83.

[13] S. Iqbal, M.F. Bari, M.S. Rahman, Solving the multi-dimensional multi-choice knapsack problem with the help of ants, in: International Conference on Swarm Intelligence, Springer, Berlin, Heidelberg, September 8, 2010, pp. 312–323.

[14] J. Ji, Z. Huang, C. Liu, X. Liu, N. Zhong, An ant colony optimization algorithm for solving the multidimensional knapsack problems, in: Proceedings of the 2007 IEEE/WIC/ACM International Conference on Intelligent Agent Technology, IEEE Computer Society, Los Alamitos, 2007, pp. 10–16.

[15] P. Vickram, A.S. Krishna, V.S. Srinivas, A survey on design paradigms to solve 0/1 knapsack problem, Int. J. Sci. Eng. Res. 7 (11) (2016).

[16] L. Anany, Introduction to the Design and Analysis of Algorithms, Villanova University, 2003.

[17] H. Kellerer, U. Pferschy, D. Pisinger, Multidimensional Knapsack Problems, in: Knapsack Problems, Springer, Berlin, Heidelberg, 2004, pp. 235–283.

[18] G.T. Basheer, Z.Y. Algamal, Nature-inspired optimization algorithms in knapsack problem: a review, Iraqi J. Stat. Sci. 16 (30) (December 1, 2019) 55–72.

[19] T. El-Ghazali, Metaheuristics: From Design to Implementation, vol. 9, Jonh Wiley and Sons Inc., Chichester, 2009, pp. 10–11.

[20] Z. Beheshti, S.M. Shamsuddin, S.S. Yuhaniz, Binary accelerated particle swarm algorithm (BAPSA) for discrete optimization problems, J. Global Optim. 57 (2) (October 1, 2013) 549–573.

[21] X.S. Yang (Ed.), Cuckoo Search and Firefly Algorithm: Theory and Applications, Springer, October 31, 2013.

[22] X.S. Yang (Ed.), Recent Advances in Swarm Intelligence and Evolutionary Computation, Springer International Publishing, 2015.

[23] N. Raj, J. Vitthalpura, Literature review on implementing binary knapsack problem, IJARIIE 3 (5) (2017). ISSN(O)-2395-4396.

[24] D. Kuchta, R. Rynca, D. Skorupka, A. Duchaczek, The use of the multiple knapsack problem in strategic management of a private Polish university, Int. J. Educ. Manag. (February 4, 2019).

[25] C. Chekuri, S. Khanna, A polynomial time approximation scheme for the multiple knapsack problem, SIAM J. Comput. 35 (3) (2005) 713–728.

[26] K. Jansen, Parameterized approximation scheme for the multiple knapsack problem, SIAM J. Comput. 39 (4) (2010) 1392–1412.

[27] K. Jansen, A fast approximation scheme for the multiple knapsack problem, in: International Conference on Current Trends in Theory and Practice of Computer Science, Springer, Berlin, Heidelberg, January 21, 2012, pp. 313–324.

[28] J. Li, W. Li, H. Wang, The Multiple Knapsack Problem with Compatible Bipartite Graphs, 2015.
[29] A.N. Ünal, A genetic algorithm for the multiple knapsack problem in dynamic environment, in: Proceedings of the World Congress on Engineering and Computer Science, vol. 2, 2013.

[30] M. Ehrgott, X. Gandibleux, A survey and annotated bibliography of multiobjective combinatorial optimization, OR-spektrum 22 (4) (November 1, 2000) 425–460.

[31] B. Chabane, M. Basseur, J.K. Hao, A practical case of the multiobjective knapsack problem: design, modelling, tests and analysis, in: International Conference on Learning and Intelligent Optimization, Springer, Cham, January 12, 2015, pp. 249–255.

[32] K. Klamroth, M.M. Wiecek, Dynamic programming approaches to the multiple criteria knapsack problem, Nav. Res. Logist. 47 (1) (February 2000) 57–76.

[33] M.E. Captivo, J. Climaco, J. Figueira, E. Martins, J.L. Santos, Solving bicriteria 0–1 knapsack problems using a labeling algorithm, Comput. Oper. Res. 30 (12) (October 1, 2003) 1865–1886.

[34] C.G. Da Silva, J. Climaco, J. Figueira, Core Problems in the Bi-criteria {0, 1} Knapsack: New Developments. Research Report 12, INESC-Coimbra, October 15, 2005.

[35] T. Erlebach, H. Kellerer, U. Pferschy, Approximating multiobjective knapsack problems, Manag. Sci. 48 (12) (December 2002) 1603–1612.

[36] X. Gandibleux, A. Freville, Tabu search based procedure for solving the 0-1 multiobjective knapsack problem: the two objectives case, J. Heuristics 6 (3) (August 1, 2000) 361–383.

[37] C.G. da Silva, J. Climaco, J. Figueira, A scatter search method for bi-criteria {0, 1}-knapsack problems, Eur. J. Oper. Res. 169 (2) (March 1, 2006) 373–391.

[38] C.G. da Silva, J. Figueira, J. Climaco, Integrating partial optimization with scatter search for solving bi-criteria [0, 1]-knapsack problems, Eur. J. Oper. Res. 177 (3) (March 16, 2007) 1656–1677.

[39] C. Grosan, M. Oltean, D. Dumitrescu, A New Evolutionary Algorithm for the Multiobjective 0/1 Knapsack Problem, Proceedings of the International Conference on Theory and Applications of Mathematics and Informatics – ICTAMI, 2003. Alba Iulia. 2003a.

[40] C. Grosan, M. Oltean, D. Dumitrescu, Performance metrics for multiobjective optimization evolutionary algorithms, in: Proceedings of Conference on Applied and Industrial Mathematics (CAIM), Oradea, 2003.

[41] E. Zitzler, L. Thiele, M. Laumanns, C.M. Fonseca, V.G. Da Fonseca, Performance assessment of multiobjective optimizers: an analysis and review, IEEE Trans. Evol. Comput. 7 (2) (May 7, 2003) 117–132.

[42] X. Ma, Y. Yan, Q. Liu, A multi-objective particle swarm optimization for multiple knapsack problem with strong constraints, in: 2018 13th IEEE Conference on Industrial Electronics and Applications (ICIEA), IEEE, May 31, 2018, pp. 1201–1205.

[43] B. Soylu, M. Köksalan, An evolutionary algorithm for the multi-objective multiple knapsack problem, in: International Conference on Multiple Criteria Decision Making, Springer, Berlin, Heidelberg, June 21, 2009, pp. 1–8.

[44] M.R. JS, Survey of multi-objective optimization methods, Struct. Multidiscip. Optim. 26 (2004) 369–395.

[45] N. Gunantara, A review of multi-objective optimization: methods and its applications, Cogent Eng. 5 (1) (January 1, 2018) 1502242.

[46] S. Hemamalini, S.P. Simon, Economic/emission load dispatch using artificial Bee colony algorithm, ACEEE Int. J. Electr. Power Eng. 1 (2) (2009).

[47] K. Naidu, H. Mokhlis, A.A. Bakar, Multiobjective optimization using weighted sum artificial bee colony algorithm for load frequency control, Int. J. Electr. Power Energy Syst. 55 (February 1, 2014) 657–667.
[48] A.W. Mohamed, A.A. Hadi, A.K. Mohamed, Gaining-sharing knowledge-based algorithm for solving optimization problems: a novel nature-inspired algorithm, Int. J. Mach. Learn. Cybern. 11 (2020) 1501–1529.

[49] K. Deb, An efficient constraint handling method for genetic algorithms, Comput. Methods Appl. Mech. Eng. 186 (2–4) (June 9, 2000) 311–338.

[50] C.A.C. Coello, Theoretical and numerical constraint-handling techniques used with evolutionary algorithms: a survey of the state of the art, Comput. Methods Appl. Mech. Eng. 191 (11–12) (2002) 1245–1287.

[51] N. Muangkote, L. Photong, A. Sukprasert, Effectiveness of Constrained Handling Techniques of Improved Constrained Differential Evolution Algorithm Applied to Constrained Optimization Problems in Mechanical Engineering, 2019.

[52] W. Long, X. Liang, Y. Huang, Y. Chen, A hybrid differential evolution augmented Lagrangian method for constrained numerical and engineering optimization, Comput. Aided Des. 45 (12) (December 1, 2013) 1562–1574.

[53] A. Bahreininejad, Improving the performance of water cycle algorithm using augmented Lagrangian method, Adv. Eng. Software 132 (June 1, 2019) 55–64.