Using Genetic Algorithms to Solve Discounted Generalized Transportation Problem

Cristina SERBAN[1]*, Doina CARP[2]

[1] Ovidius University of Constanta, 124 Mamaia Blvd., 900527, Constanta, Romania
cgherghina@gmail.com (**Corresponding author*)

[2] University POLITEHNICA of Bucharest, 313 Splaiul Independentei, 060042, Bucharest, Romania
doina.carp@gmail.com

Abstract: In the Discounted Generalized Transportation problem (DGT) the cost of transporting a type of product from a source to a destination depends on the amount of transported commodity. Based on this property, in this paper a genetic algorithm is developed to find a high-quality solution of the DGT problem. A numerical example is also presented. The proposed method could be easily integrated into a decision support platform intended to help transport planners to solve problems and make decisions in very limited timeframes.

Keywords: Genetic algorithms, Discounted Generalized Transportation problem, Optimization.

1. Introduction

The classical transportation problem (Hitchcock, 1941) is one of the most studied linear programming problems in the specialized literature, for which practical computational algorithms that take advantage of the special structure of the problem have been developed, whether it is consistent (Dantizg, 1951) or inconsistent and Simplex-type algorithms are no more applicable (Carp et al., 2015). The generalized transportation problem extends the linear transportation problem, assuming that the amounts of goods transported from the supply points to the destinations change during the transportation process. This problem was discussed by Balas & Ivanescu, 1964; Gupta, 1978, and others.

Depending on the type of the cost function, a transportation problem can be classified into linear and nonlinear problem. Both the classical and the generalized transportation problems are linear, considering that the transportation cost is constant, regardless the shipped quantity. However, in many practical applications the cost per unit commodity shipped from a source to a destination is not fixed, depending by a series of factors like the costs of raw materials and transport. This would lead to a cost function either piecewise linear or separable concave.

The Discounted Generalized Transportation problem (DGT) assumes that the transportation costs depend on the amount of transported goods (e.g., discounts offered for large quantities). The DGT problem was studied by several authors. Balachandran & Perry (1976) developed an algorithm based upon a branch-and-bound solution procedure. Goossens et al. (2007) studied several variants of DGT problem and proposed and tested three exact algorithms (min-cost flow-based branch-and-bound, linear programming-based branch-and-bound, and branch-and-cut) on randomly generated instances. Ojha et al. (2010) considered an extension of the DGT problem, where all unit discounts or incremental quantity discounts are offered, and the cost depends on the amount offered, source, destination and shipping. They developed a genetic algorithm and applied it on several numerical examples. Mubashiru (2014) introduced a Karush-Kuhn-Tucker optimality algorithm to solve a transportation problem with volume discount, with a convex cost function meaning that the objective function is composed of the unit transportation cost and also of production cost related to each commodity. Acharya et al. (2013) proposed an algorithm based on the linear programming model of the generalized transportation problem that required introducing slack or artificial variables into the model. Several algorithms are introduced by Arpita & Bikash (2014) to solve the DGT problem by a two-vehicle cost varying transportation model. The transportation cost depends on the amount of transport quantity, which, in turn, depends on the capacity of the transport vehicles.

Discounted fixed cost transportation problem can be considered as a version of DGT problem, with two types of costs: a fixed charge which is independent of the shipping amount, and a variable cost which is assessed based on the amount of shipped goods. Yousefi et al. (2017) considered this problem, with discount assumptions on both fixed and variable costs, and developed a genetic algorithm based on spanning tree-based

https://doi.org/10.24846/v30i3y202103

representation and priority-based representation. George et al. (2014) tackled the transportation problem with volume discount on distribution cost by using the Karush-Kuhn-Tucker (KKT) optimality condition. Ghaseemi Tari (2016) proposed a discrete nonlinear optimization model and developed a hybrid dynamic programming algorithm for finding the optimal solution.

This paper focuses on the development of an evolutionary optimization technique to solve the nonlinear DGT problem with the cost function taken as step function, stated as:

$$\min \sum_{i=1}^{n} \sum_{j=1}^{m} c_{ij} x_{ij} \quad (1)$$

subject to

$$\sum_{j=1}^{m} a_{ij} x_{ij} \le s_i,\ a_{ij} > 0,\ i = 1,..,n$$

$$\sum_{i=1}^{n} x_{ij} = d_j,\ j = 1,...,m$$

$$x_{ij} \ge 0,\ i = 1,...,n,\ j = 1,...,m$$

$$c_{ij} = \begin{cases} c_{ij}^1, \text{ if } 0 \le x_{ij} \le x_{ij}^1 \\ c_{ij}^2, \text{ if } x_{ij}^1 < x_{ij} \le x_{ij}^2 \\ \dots \\ c_{ij}^t, \text{ if } x_{ij}^{t-1} < x_{ij} \end{cases}$$

and $c_{ij}^1 > c_{ij}^2 > ... > c_{ij}^t$,

where

$(s_i)_{i=1,...,n}$ – supplies of some goods available at supply warehouses called sources, $(S_i)_{i \in \{1,...,n\}}$

$(d_j)_{j=1,...,m}$ – demands requested by receiving centers called destinations, $(D_j)_{j \in \{1,...,m\}}$

$(x_{ij})_{i \in \{1,...,n\}, j \in \{1,...,m\}}$ – number of units transported from source S_i to destination D_j.

$(c_{ij})_{i \in \{1,...,n\}, j \in \{1,...,m\}}$ – the cost of moving one unit from source S_i to destination D_j.

$(a_{ij})_{i \in \{1,...,n\}, j \in \{1,...,m\}}$ – positive constants rather than unity, called multipliers, showing the amount of good provided by source i

The equality constraints ensure that the product unit requirements will be satisfied at destinations, whereas the inequality ones express that the supply of product units available at each source must not be exceeded.

To illustrate the efficiency of the proposed evolutionary optimization algorithm, we applied the model on the numerical example presented in (Acharya et al., 2013) and the results of these two methods are compared. We aimed to calibrate the genetic algorithm we developed, i.e., to find the best fitting set of evolutionary parameters that favors finding the best transportation cost for a benchmark instance of the DGT problem that has been solved optimally (Acharya et al., 2013). Afterwards, the calibrated GA may be applied on other instances of DGT problem to get a high-quality solution.

Scientific approach of transport planning has contributed a lot to an efficient real-time transport management, reducing delivery cost and times. Use of information systems with user friendly interfaces (Suduc et al., 2009) gives transport decision makers the possibility to handle more data, apply complex modelling functions and visualize a set of solutions, each one having its own advantages and disadvantages. Currently, there are supply chain networks (Pop et al., 2017; Cosma et al., 2020), decision support systems (DSS) or platforms (Candea & Filip, 2016) that transport planners may use to make decisions in very limited timeframes regarding the transported goods, both in terms of quantity and distribution on warehouses and the price obtained for the entire transport service. Consequently, it can be considered that multicriteria linear problems are to be solved, whose restrictions are given by the shape and volume of the warehouses and the objective functions are about to maximize the quantity of goods, to minimize costs and to maximize profit. The method presented in this paper can be easily integrated into a DSS-based solution of a multicriteria linear problem in order to refine the distribution of maximum quantity of goods on the warehouses so as to obtain the best (low) cost using discount facilities.

Following this introduction, the proposed methodology for solving the DGT problem is presented in Section 2. In Section 3, the analysis of the results of the proposed approach is provided. Finally, the conclusions are discussed in Section 4.

2. Methodology and Data

2.1 Evolutionary Approach of the DGT Problem

The DGT problem is reformulated in order to suit the genetic algorithm requirements. In this respect, first the unknowns are renumbered as

$$x_{ij} \to y_l, i = \overline{1,n}, j = \overline{1,m}, l = \overline{1,mn} \quad (2)$$

Let now $c \in \mathbb{R}^{mn}$ be the cost vector of the above problem, B_1, B_2 be the $m \times mn, n \times mn$ matrices corresponding to the m equalities and the n inequalities from (1), respectively. Furthermore, let $d \in \mathbb{R}^m, s \in \mathbb{R}^n$ be the demands and supplies vectors, respectively. Then, the problem (1) can be written as

$$\min \langle c, y \rangle \quad (3)$$

subject to

$$B_1 y = d, \quad B_2 y \le s, \quad y \ge 0$$

$$c_l = \begin{cases} c_l^1, if\ 0 \le y_l \le y_l^1 \\ c_l^2, if\ y_l^1 < y_l \le y_l^2 \\ \dots \\ c_l^t, if\ y_l^{t-1} < y_l \end{cases}$$

and $c_l^1 > c_l^2 > \dots > c_l^t$.

2.2 Genetic Algorithms

Genetic Algorithms (GAs) are evolutionary algorithms inspired from the natural process of selection which leads to the survival of the fittest individuals. Using GAs to solve a problem means exploring different regions of the solution space and refining the collected information. A function called "fitness" guides the search in the solution space and measures the closeness to the optimal solution. A solution is represented by a chromosome.

GAs maintain a population of potential solutions that evolves across generations by selecting the fittest individuals according to their level of fitness and altering them by means of crossover and mutation to form new solutions (Sivanandam & Deepa, 2008).

Developing a GA means determining the encoding scheme of solutions, the fitness function, and the genetic operators.

In the present model, a chromosome is a feasible set of *mn* real parameters $y_l = x_{ij}$, which are referred to as genes and represent the amount of commodities transported from source S_i to destination D_j. In order to have feasible chromosomes and preserve the stochastic characteristics of the genetic algorithm, only the equality constraints from (3) are imposed for each individual, by applying a procedure, called *FCP (Feasible Chromosome Procedure)*, that randomly generates appropriate numbers. To exemplify the FCP procedure, let's suppose that one wants to find real numbers x_1, x_2, x_3 such that $S = x_1 + x_2 + x_3$. Firstly, let a and b be two numbers randomly chosen in $[0, S]$. Then, order them such that $0 \le a \le b \le S$ and set $x_1 = a, x_2 = b - a, x_3 = S - b$.

The fitness function evaluates the performance of the chromosomes and represents an objective value. The goal is to minimize the total cost of transporting items from sources to destinations, so the objective function is $f(y) = \langle c, y \rangle$. The cost dependencies from (3) are imposed according to each individual whose objective function is calculated.

Since the chromosomes initially do not satisfy the inequality constraints $g_i(y) = s - B_2 y, i = \overline{1,n}$, penalty functions must be added to the objective function.

$$F(y) = \begin{cases} f(y), & y \in feasible\ region \\ f(y) + P(y), & otherwise \end{cases} \quad (4)$$

A dynamic penalty function described by Joines & Houck (1994), which changes as the GA proceeds, is used to model the present penalty function.

$$P(y, \alpha, \beta) = \rho_k^{\alpha} \times SVC(\beta, y) \quad (5)$$

$$\rho_k = C \times k \quad (6)$$

$$SVC(\beta, y) = \sum_{i=1}^{n} p_i^{\beta}(y), \quad \beta = 1, 2, \dots \quad (7)$$

$$p_i(y) = \begin{cases} 0, g_i(y) \ge 0 \\ |g_i(y)|, otherwise \end{cases} \quad (8)$$

where α, β, C are parameters of the method, and k is the number of the current generation considered. The parameters of the method were empirically tuned, the best results being obtained with $C = 1$, $\alpha = 1$ and $\beta = 1$.

GAs have proved highly robust methods of finding good solutions to difficult optimization problems due to the choice of well-designed genetic operators and optimal parameters. Genetic operators are applied to individuals to obtain genetic diversity and generate a new population. There are three operators: selection, crossover, and mutation.

According to the "survival of the fittest" concept, the selection operator implements a probabilistic selection of individuals and those with higher fitness values will have a better chance to survive and will form the mating pool. Here, the tournament selection is used (Yadav & Sohal, 2017).

The crossover operator guides the search process to the good solutions. Individuals are selected with a user-definable probability, called "crossover rate", from the mating pool and produce one or two offspring, which will replace one or two individuals from the current population. Due to the nature of the present problem, the whole arithmetic crossover was chosen for implementation (Furqan et al., 2017), thus ensuring that the equality constraints from (3) are met by the offspring. Then, a weak parent replacement (Sivanandam & Deepa, 2008) is applied.

Mutation is the genetic operator that prevents the algorithm falling into a local optimum. By mutation, some offspring resulted from crossover are chosen with a user-definable probability called "mutation rate", and one or more of their genes are altered. In this paper, a case of random resetting mutation is designed in order not to violate the equality constraints from (3). Firstly, a gene is randomly selected, and then all the genes that violate the corresponding equality constraint from (3) are set to values from the set of permissible values using the FCP procedure.

The accuracy of the results returned by a genetic algorithm depends on the parameters it uses, which are problem-specific, and there is no best global value for them. There are four configuration parameters that must be considered: stop condition, population size, crossover rate and mutation rate.

There may be various stopping conditions in Genetic Algorithms (Sivanandam & Deepa, 2008). In this paper, the evolution is stopped when most of the population (e.g., 97%) has the same fitness value or a user-specified maximum number of generations is reached.

A very large population size would mean a better exploration of the search space, but also an increase in the runtime of the algorithm (Sivanandam & Deepa, 2008). If crossover and mutation rates are too high or too low, the search will pass over either good solutions, or over the entire regions of solution space, respectively. Most studies in the field of GAs recommend a population size between 20 and 100 individuals, a crossover rate over 60% and a mutation rate of at most 10%. In this paper, to set the appropriate parameters, multiple tests were configured and run, each test being repeated 5 times and the results being averaged to increase their precision.

2.3 The proposed GA for Solving the DGT Problem

The implemented genetic algorithm is presented in the following.

Input: The destination demands (d), the source supplies (s) and the unit cost (c) dependencies of the amount of commodity transported

Output: The best solution (y), the associated cost vector (c), and the best (minimum) cost of transportation $(\langle c, y \rangle)$.

Begin

1. **Generate** a random population of individuals, which are represented as real values arrays of length *mn* and satisfy the equality constraints from (3)
2. **Compute** the fitness function
   a) **Select** some chromosomes for the crossover operation (the number of selected individuals is defined by the crossover rate)
   a) **Apply** the crossover operator described in 2.2 to generate new offspring
   a) **Copy** the remaining chromosomes (that were not recombined) to the next generation
3. **Select** a few chromosomes for mutation (the number of selected individuals is defined by the mutation rate)
4. If the maximum number of generations is reached **then** stop, **else** go to step 2

End

3. Results and Discussion

The present GA is implemented using MATLAB R2015a. To validate the proposed approach of the DGT problem, it was tested over the instance discussed in (Acharya et al., 2013). The problem is given in Table 1.

Table 1. The DGT problem (Acharya et al., 2013)

	D_1	D_2	D_3	D_4	Supply (s)
S_1	c_{11} $a_{11} = 0.35$	c_{12} $a_{12} = 0.5$	c_{13} $a_{13} = 0.35$	c_{14} $a_{14} = 0.5$	200
S_2	c_{21} $a_{21} = 0.9$	c_{22} $a_{22} = 0.84$	c_{23} $a_{23} = 0.3$	c_{24} $a_{24} = 0.4$	500
S_3	c_{31} $a_{31} = 0.8$	c_{32} $a_{32} = 0.4$	c_{33} $a_{33} = 0.74$	c_{34} $a_{34} = 0.9$	400
Demand (d)	200	400	500	1000	

In the first instance, multiple settings of the GA parameters were tested.

To choose the proper population size, the stop condition was set to 300 generations, the crossover rate to 0.75, and the mutation rate to 0.01. The population size is chosen as the value against which population growth no longer brings significant changes to fitness. The best value of the total cost for which the inequality constraints from (3) are also met was 1 214 150.79, obtained with a population size of 60 individuals (marked in Table 2).

For the stop condition, the population size was set to the value previously estimated, the crossover rate to 0.75 and the mutation rate to 0.01. Then, the number of generations after which the fitness does not improve and the inequality constraints from (3) are satisfied had to be found. It resulted that the best fitting number of generations was 600 (marked in Table 3).

Table 2. The effect of population size over the GA results

Pop. Size	No. Gen.	Crossover Rate	Mutation Rate	Cost	Time (s)	Supply		
						s1	s2	s3
10	300	0.75	0.01	1218466.90	0.44	199.48	403.92	255.25
20	300	0.75	0.01	1217570.15	0.71	198.45	397.46	275.23
30	300	0.75	0.01	1215113.71	0.99	199.96	402.06	255.08
40	300	0.75	0.01	1215839.23	1.26	199.80	404.15	250.88
50	300	0.75	0.01	1216164.35	1.51	199.92	412.07	235.32
60	300	0.75	0.01	1214150.79	1.80	199.96	406.08	248.42
70	300	0.75	0.01	1214588.45	2.10	199.98	406.34	245.73
80	300	0.75	0.01	1214975.66	2.46	199.80	400.72	260.39
90	300	0.75	0.01	1214485.56	2.61	199.93	402.86	258.51
100	300	0.75	0.01	1214615.24	3.06	199.75	402.91	253.67

Table 3. The effect of number of generations over the GA results

Pop. Size	No. Gen.	Crossover Rate	Mutation Rate	Cost	Time (s)	Supply		
						s1	s2	s3
60	100	0.75	0.01	1215947.35	0.71	200.26	404.12	253.48
60	200	0.75	0.01	1216280.05	1.23	199.99	399.66	263.35
60	300	0.75	0.01	1215636.20	1.90	200.00	410.93	232.64
60	400	0.75	0.01	1214217.10	2.41	199.96	402.93	251.65
60	500	0.75	0.01	1214305.22	3.00	199.99	399.56	262.14
60	600	0.75	0.01	1213743.29	3.59	199.95	400.22	258.33
60	700	0.75	0.01	1213828.79	4.19	199.99	402.55	254.83
60	800	0.75	0.01	1213938.43	5.13	199.97	405.75	248.01
60	900	0.75	0.01	1214047.37	5.57	200.00	404.60	249.77
60	1000	0.75	0.01	1214100.51	6.01	200.00	404.75	247.79

Next, the optimal crossover rate was chosen as the value which gives the best fitness. The tests were run using the parameters previously determined and the mutation rate was set to 0.01. Table 4 shows that a proper crossover rate is 0.70. This value reassures that the search will explore much of the solutions space.

For the last parameter, the evolution of GA results for different values of the mutation rate was analyzed (Table 5). The other parameters were set to values established in previous tests. The best mutation rate was considered to be 0.02. For this value, the average transportation cost has the best evolution along the iterations, being reduced from 1 220 450 to 1 216 516.

Table 6 shows the parameters used to implement the proposed genetic algorithm (columns 1-4), the execution time (column 5) and the resulted supplies values (columns 9, 10 and 11). Columns 6 and 7, titled Cost (Min) and Cost (Mean) provide the lowest and the average transportation cost out of 5 runs in the execution, respectively. Column 8 shows the standard deviation of the complete population, whose values (approximately 0.07% of the mean transportation cost) show that GA can find solutions which are very close to each other in its various runs.

Table 7 presents the best solution and the associated cost vector estimated with the proposed approach of the DGT problem, whereas Table 8 shows the results from (Acharya et al., 2013). Both approaches determine the same transportation cost (1 213 514.5) and layout of solution, while respecting both the destination

Table 4. The effect of crossover rate over the GA results

Pop. Size	No. Gen.	Crossover Rate	Mutation Rate	Cost	Time (s)	Supply		
						s1	s2	s3
60	600	0.50	0.01	1214843.87	3.60	200.00	409.52	236.80
60	600	0.55	0.01	1214578.34	3.52	199.96	403.66	253.80
60	600	0.60	0.01	1214369.16	3.58	199.96	406.92	246.01
60	600	0.65	0.01	1214299.12	3.56	199.91	404.56	255.06
60	600	0.70	0.01	1213889.68	3.57	199.98	405.55	245.97
60	600	0.75	0.01	1213982.54	3.64	199.90	407.16	243.63
60	600	0.80	0.01	1213987.91	3.69	200.00	417.86	213.44
60	600	0.85	0.01	1214642.25	3.68	198.64	401.03	261.10
60	600	0.90	0.01	1214764.61	3.71	199.96	409.64	238.45
60	600	0.95	0.01	1214248.91	3.78	199.97	413.99	225.12

Table 5. The effect of mutation rate over the GA results

Mut. Rate / Gen.	0.01	0.02	0.03	0.04	0.05	0.06	0.07	0.08	0.09	0.1
100	1219489	1220450	1223788	1225937	1229528	1232356	1234116	1240458	1240202	1243496
200	1218606	1220208	1221948	1225411	1228641	1226185	1230468	1239524	1239649	1241204
300	1216731	1219085	1221784	1225899	1228474	1228638	1230916	1238720	1239198	1237954
400	1216900	1216608	1221724	1222513	1225542	1228188	1228186	1233572	1238661	1237406
500	1216795	1216516	1221515	1221518	1225975	1228458	1228057	1235559	1233150	1235612

Table 6. The results and parameters settings of the proposed GA

Pop. Size	No. Gen.	Cross. Rate	Mut. Rate	Time (s)	Cost (Min)	Cost (Mean)	St. Dev.	Supply		
								s1	s2	s3
60	600	0.70	0.02	3.93	1 213 514.5	1 214 074.4	878.8	200	400	262.8

Table 7. Best solution and the associated cost vector for GA approach

	D_1	D_2	D_3	D_4	Supply (s)
S_1	$c_{11}=203$ $a_{11}=0.35$ $x_{11}=71.5$	$c_{12}=401$ $a_{12}=0.5$	$c_{13}=398$ $a_{13}=0.35$ $x_{13}=500$	$c_{14}=751$ $a_{14}=0.5$	200.02
S_2	$c_{21}=502$ $a_{21}=0.9$	$c_{22}=604$ $a_{22}=0.84$	$c_{23}=602$ $a_{23}=0.3$	$c_{24}=749$ $a_{24}=0.4$ $x_{24}=1000$	400
S_3	$c_{31}=400$ $a_{31}=0.8$ $x_{31}=128.5$	$c_{32}=499$ $a_{32}=0.4$ $x_{32}=400$	$c_{33}=602$ $a_{33}=0.74$	$c_{34}=901$ $a_{34}=0.9$	262.8
Demand (d)	200	400	500	1000	

Table 8. Optimal solution and the associated cost vector (Acharya et al., 2013)

	D_1	D_2	D_3	D_4	Slack	Supply (s)
S_1	$c_{11}=203$ $a_{11}=0.35$ $x_{11}=71.5$	$c_{12}=401$ $a_{12}=0.5$	$c_{13}=398$ $a_{13}=0.35$ $x_{13}=500$	$c_{14}=751$ $a_{14}=0.5$	$c_{15}=0$ $a_{15}=1$	200.02
S_2	$c_{21}=502$ $a_{21}=0.9$	$c_{22}=604$ $a_{22}=0.84$	$c_{23}=602$ $a_{23}=0.3$	$c_{24}=749$ $a_{24}=0.4$ $x_{24}=1000$	$c_{25}=0$ $a_{25}=1$ $x_{25}=100$	400
S_3	$c_{31}=400$ $a_{31}=0.8$ $x_{31}=128.5$	$c_{32}=499$ $a_{32}=0.4$ $x_{32}=400$	$c_{33}=602$ $a_{33}=0.74$	$c_{34}=901$ $a_{34}=0.9$	$c_{35}=0$ $a_{35}=1$ $x_{35}=137.2$	262.8
Demand (d)	200	400	500	1000		

and supply constraints (the first supply is slightly exceeded in both cases).

We consider that the set of parameters determined by multiple tests is the most adequate one to get the best (lowest) transportation cost, thus, the proposed Genetic Algorithm may be used to solve other instances of the DGT problem.

Further on, for evaluating the efficiency of the proposed algorithm with the tuned parameter set, it was applied to some test problems.

Firstly, several unbalanced test problems of six sizes are randomly generated and solved: (n,m) = (10,10), (10, 20), (20,40), (40,40), (40,60) and (60, 60). Each test problem is executed 5 times. The multipliers are chosen from [0.1,0.9], and the supply and demand from the interval [100,500], uniformly distributed. To generate the transportation cost, two uniform density functions, c^1 and c^2, each with the range of [10–20], are generated. To consider the real- world cases, the costs are decreased as the units transported are increased. The discount mechanism is illustrated by (9)

$$c_{ij}=\begin{cases}2(c_{ij}^1+c_{ij}^2),\ if\ 0\le x_{ij}\le C^1\\ c_{ij}^1+c_{ij}^2,\ if\ C^1<x_{ij}\le C^2\\ c_{ij}^1,\ if\ C^2<x_{ij}\end{cases} \quad (9)$$

where $C^1=25$ and $C^2=50$.

The solution determined for each test problem fully respected both the destination and the supply constraints. Table 9 shows the average transportation cost and the running time. Columns 3 and 5 display the standard deviation of the cost and of the execution time.

Table 9. Computational results of the random DGT problems

Problem size (n,m)	Cost (Mean)	St.Dev (Cost)	Time (Mean)	St.Dev. (Time)
(10,10)	59 034	2382	10.97	0.16
(10,20)	127 560	4849.9	25.20	0.98
(20,40)	438 240	5011.3	98.54	3.21
(40,40)	642 271	2458	187.69	10.35
(40,60)	914 040	6870.2	487.2	8.16
(60,60)	1 045 100	5741.5	947.3	18.5

From the results of this computational test, it can be concluded that the submitted algorithm can obtain robust solutions which are very close to each other in its various runs (standard deviation is less than 10% of the mean transportation cost).

For further evaluating the efficiency of the proposed GA, a balanced test DGT problem (Table 10) was randomly generated and solved by the proposed GA. The discount mechanism is illustrated by (10). Then, the test problem was also solved by using the North West (NW) corner method for determining initial basic feasible solution, and then setting up the unit transportation cost, which varies in each iteration, based on the amount of the transported goods (NW-DGT algorithm). The results of this computational experiment are presented in Tables 11, 12 and 13. Referring to these tables, it can be noticed that the solution of the present GA is 1.03% better than the one computed with NW-DGT method.

Table 10. An example of the DGT problem

	D_1	D_2	D_3		D_4	D_5	D_6	Supply (s)
S_1	c_{11} $a_{11}=1$	$c_{12}=2$ $a_{12}=1$	$c_{13}=4$ $a_{13}=1$		$c_{14}=3$ $a_{14}=1$	$c_{15}=5$ $a_{15}=1$	$c_{16}=4$ $a_{16}=1$	25
S_2	$c_{21}=4$ $a_{21}=1$	c_{22} $a_{22}=1$	$c_{23}=2$ $a_{23}=1$		$c_{24}=6$ $a_{24}=1$	$c_{25}=8$ $a_{25}=1$	$c_{26}=7$ $a_{26}=1$	45
S_3	$c_{31}=3$ $a_{31}=1$	$c_{32}=5$ $a_{32}=1$	$c_{33}=7$ $a_{33}=1$		$c_{34}=11$ $a_{34}=1$	$c_{35}=4$ $a_{35}=1$	$c_{36}=5$ $a_{36}=1$	36
S_4	$c_{41}=4$ $a_{41}=1$	$c_{42}=10$ $a_{42}=1$	$c_{43}=8$ $a_{43}=1$		$c_{44}=3$ $a_{44}=1$	$c_{45}=4$ $a_{45}=1$	c_{46} $a_{46}=1$	44
Demand (d)	21	12	33		44	10	30	

Table 11. Best solution of the DGT test problem, computed with GA

	D_1	D_2	D_3	D_4	D_5	D_6	Supply (s)
S_1	$c_{11}=4$ $a_{11}=1$	$c_{12}=2$ $a_{12}=1$	$c_{13}=4$ $a_{13}=1$	$c_{14}=3$ $a_{14}=1$ $x_{14}=25$	$c_{15}=5$ $a_{15}=1$	$c_{16}=4$ $a_{16}=1$	25
S_2	$c_{21}=4$ $a_{21}=1$	$c_{22}=2$ $a_{22}=1$ $x_{22}=10.5$	$c_{23}=2$ $a_{23}=1$ $x_{23}=30.5$	$c_{24}=6$ $a_{24}=1$ $x_{24}=2$	$c_{25}=8$ $a_{25}=1$	$c_{26}=7$ $a_{26}=1$ $x_{26}=2$	45
S_3	$c_{31}=3$ $a_{31}=1$ $x_{31}=21$	$c_{32}=5$ $a_{32}=1$ $x_{32}=1.5$	$c_{33}=7$ $a_{33}=1$ $x_{33}=2.5$	$c_{34}=11$ $a_{34}=1$	$c_{35}=4$ $a_{35}=1$ $x_{35}=10$	$c_{36}=5$ $a_{36}=1$ $x_{36}=1$	36
S_4	$c_{41}=4$ $a_{41}=1$	$c_{42}=10$ $a_{42}=1$	$c_{43}=8$ $a_{43}=1$	$c_{44}=3$ $a_{44}=1$ $x_{44}=17$	$c_{45}=4$ $a_{45}=1$	$c_{46}=2$ $a_{46}=1$ $x_{46}=27$	44
Demand (d)	21	12	33	44	10	30	

Table 12. Best solution of the DGT test problem, computed with NW-DGT method

	D_1	D_2	D_3	D_4	D_5	D_6	Supply (s)
S_1	$c_{11}=4$ $a_{11}=1$	$c_{12}=2$ $a_{12}=1$ $x_{12}=12$	$c_{13}=4$ $a_{13}=1$	$c_{14}=3$ $a_{14}=1$ $x_{14}=13$	$c_{15}=5$ $a_{15}=1$	$c_{16}=4$ $a_{16}=1$	25
S_2	$c_{21}=4$ $a_{21}=1$	$c_{22}=2$ $a_{22}=1$	$c_{23}=2$ $a_{23}=1$ $x_{23}=33$	$c_{24}=6$ $a_{24}=1$ $x_{24}=12$	$c_{25}=8$ $a_{25}=1$	$c_{26}=7$ $a_{26}=1$	45
S_3	$c_{31}=3$ $a_{31}=1$ $x_{31}=21$	$c_{32}=5$ $a_{32}=1$	$c_{33}=7$ $a_{33}=1$	$c_{34}=11$ $a_{34}=1$	$c_{35}=4$ $a_{35}=1$ $x_{35}=10$	$c_{36}=5$ $a_{36}=1$ $x_{36}=5$	36
S_4	$c_{41}=4$ $a_{41}=1$	$c_{42}=10$ $a_{42}=1$	$c_{43}=8$ $a_{43}=1$	$c_{44}=3$ $a_{44}=1$ $x_{44}=19$	$c_{45}=4$ $a_{45}=1$	$c_{46}=2$ $a_{46}=1$ $x_{46}=25$	44
Demand (d)	21	12	33	44	10	30	

$$
\begin{aligned}
c_{11} &= \begin{cases} 4, \text{ if } 0 \le x_{11} \le 14 \\ 3, \text{ if } 14 < x_{11} \end{cases} \\
c_{22} &= \begin{cases} 5, \text{ if } 0 \le x_{22} \le 7 \\ 3, \text{ if } 7 < x_{22} \end{cases} \\
c_{46} &= \begin{cases} 5, \text{ if } 0 \le x_{46} \le 20 \\ 2, \text{ if } 20 < x_{46} \end{cases}
\end{aligned} \tag{10}
$$

Table 13. Comparison of the proposed genetic algorithm with NW-DGT method

Algorithm	Cost (Min)	Cost (Mean)	St. Dev.	Time (s)
GA	431.5	442.47	29.8428	4.03
NW-DGT	436.0	-	-	-

To evaluate the efficiency of the present GA in comparison with other existing algorithms, TP1 and TP2 are selected from the specialized literature (Arpita & Bikash, 2014). Two of the Two-vehicle CVTP test problems presented were settled on: example 1, solved with Algorithm TP1, and example 2, solved with Algorithm TP2. The proposed GA was set with the control parameters from Table 6, and, for each test problem, the standard deviation, the lowest and average transportation cost were extracted, all out of 5 runs in the execution. Tables 14 and 15 show the results of the proposed GA and those obtained by Arpita & Bikash (2014). One can see that the cost estimated by GA is smaller, whilst all the destinations demands are satisfied and sources stocks are not exceeded.

Table 14. Comparison of the proposed Genetic Algorithm with TP1 (Arpita & Bikash, 2014)

	TP1	GA
Solution	$x_{12}=8$ $x_{13}=7$ $x_{21}=10$ $x_{22}=2$ $x_{33}=3$	$x_{11}=8$ $x_{13}=7$ $x_{21}=2$ $x_{22}=10$ $x_{33}=3$
Demand	$d_1=10, d_2=10, d_3=10$	
Supply	$s_1=25, s_2=12, s_3=3$	
Cost (Min)	35	34
Cost (Mean)	-	37.04
St.Dev.	-	3.21

Table 15. Comparison of the proposed Genetic Algorithm with TP2 (Arpita & Bikash, 2014)

	TP2	GA
Solution	$x_{11}=23$ $x_{13}=2$ $x_{22}=15$ $x_{23}=2$ $x_{33}=8$	$x_{11}=18$ $x_{12}=7$ $x_{21}=5$ $x_{23}=12$ $x_{32}=8$
Demand	$d_1=23, d_2=15, d_3=12$	
Supply	$s_1=25, s_2=17, s_3=8$	
Cost (Min)	51	40
Cost (Mean)	-	45.38
St.Dev.	-	4.23

4. Conclusion

This article presents a new algorithm for finding the optimal transportation cost for the Discounted Generalized Transportation problem. The validity of the proposed algorithm is given, by comparing the solution of this evolutionary approach with the optimal solution obtained by Acharya et al. (2013), while respecting all constraints. Other numerical experiments show that the proposed GA is an efficient algorithm for the DGT problem. Therefore, the proposed GA may be used to get a high-quality solution of other instances of the DGT problem (non-linear transportation problem with costs which depend on the quantity of the goods shipped).

REFERENCES

Acharya, D., Basu, M. & Das, A. (2013). Discounted Generalized Transportation Problem, *International Journal of Scientific and Research Publications*, *3*(7), 1-6.

Arpita, P. & Bikash, C. D. (2014). 2-Vehicle Cost Varying Transportation Problem, *Journal of Uncertain Systems*, *8*(1), 44-57.

Balachandran, V. & Perry, A. (1976). Transportation type problems with quantity discounts, *Naval Research Logistics Quarterly, 23(2), 195-209.*

Balas, E. & Ivanescu, P. L. (1964). On the Generalized Transportation Problem, *Management Science*, *11*(1), 188-202.

Candea, C. & Filip, F. G. (2016) Towards Intelligent Collaborative Decision Support Platforms, *Studies in Informatics and Control*, *25*(2), 143-152. DOI: 10.24846/v25i2y201601

Carp, D., Popa, C. & Serban, C. (2015). Modified Han algorithm for inconsistent linear inequalities, *Carpathian Journal of Mathematics*, *31*(1), 45-52.

Cosma, O., Pop, P. C. & Danciulescu, D. (2020). A novel metaheuristic approach for a two-stage transportation problem with fixed costs associated to the routes, *Computers and Operations Research*, *118*(1), art. no. 104906.

Dantzig, G. B. (1951). Application of the Simplex Method to a Transportation problem. In Koopmans, T. C. (ed.), *Activity Analysis of Production and Allocation*, 359-373. New York-London.

Furqan, Mhd., Hartono, H., Ongko, E. & Ikhsan, M. (2017). Performance of Arithmetic Crossover and Heuristic Crossover in Genetic Algorithm Based on Alpha Parameter, *IOSR Journal of Computer Engineering*, *19*(5), 31-36.

George, A., Chukwudi, J. & Jude, O. (2014). Transportation Algorithm with Volume Discount on Distribution Cost (A Case Study of the Nigerian Bottling Company Plc Owerri Plant), *American Journal of Applied Mathematics and Statistics*, *2*(5), 318-323.

Ghassemi Tari, F. (2016). A hybrid dynamic programming for solving fixed cost transportation with discounted mechanism. *Journal of Optimization,*1-9, 8518921.

Goossens, D. R., Maas, A. J. T., Spieksman, F. C. R. & de Klundert, J. J. (2007). Exact algorithms for procurement problems under a total quantity discount structure, *European Journal of Operational Research, 178(2)*, 603-626.

Gupta, R. (1978). Solving the generalized transportation problem with constraints, *Journal of Applied Mathematics and Mechanics, 58*(10), 451-458.

Hitchcock, F. L. (1941) The distribution of a product from several resources to numerous localities, *Journal of Mathematical Physics*, *20*(1), 224-230.

Joines, J. & Houck, C. (1994) On the Use of Non-Stationary Penalty Functions to Solve Nonlinear Constrained Optimization Problems with GA's. In *Proceedings of first IEEE Conference on Evolutionary Computation* (pp. 579-584).

Mubashiru, A. S. S. (2014). Transportation with volume discount: a case study of a logistic operator in Ghana, *Journal of Transport Literature, 8*(2), 7-37.

Ojha, A., Das, B., Mondal, S. & Maiti, M. (2010). A solid transportation problem for an item with fixed charge, vehicle cost and price discounted varying charge using genetic algorithm, *Applied Soft Computing, 10*(1), 100-110.

Pop, P. C., Sabo, C., Biesinger, B., Hu, B. & Raidl, G. (2017). Solving the two-stage fixed-charge transportation problem with a hybrid genetic algorithm, *Carpathian Journal of Mathematics, 33*(3), 365-371.

Sivanandam, S. N. & Deepa, S. N. (2008). *Introduction to Genetic Algorithms*. Springer Berlin, Heidelberg.

Suduc, A. M., Bîzoi, M., Duță, L. & Gorghiu, G. (2009). Interface Architecture for a Web-Based Group Decision Support System, *Studies in Informatics and Control, 18*(3), 241-246.

Yadav, S. L. & Sohal, A. (2017). Comparative Study of Different Selection Techniques in Genetic Algorithm, *International Journal of Engineering, Science and Mathematics*, *6*(3), 174-180.

Yousefi, K., Afshari, A. J. & Hajiaghaei-Keshteli, M. (2017). Genetic algorithm for fixed charge transportation problem with discount models. In *13th International Conference on Industrial Engineering*, Iran (pp. 22-23).