CARPATHIAN J. MATH. Online version at http://carpathian.ubm.ro
33 (2017), No. 3, 365 - 371 Print Edition: ISSN 1584 - 2851 Online Edition: ISSN 1843 - 4401

Solving the Two-Stage Fixed-Charge Transportation Problem with a Hybrid Genetic Algorithm

PETRICĂ C. POP[1], COSMIN SABO[1], BENJAMIN BIESINGER[2], BIN HU[2] and GÜNTHER R. RAIDL[3]

ABSTRACT. This article considers the two-stage fixed-charge transportation problem which models an important transportation application in a supply chain, from manufacturers to customers through distribution centers. For solving this optimization problem we describe a hybrid algorithm that combines a steady-state genetic algorithm with a local search procedure. The computational results for an often used collection of benchmark instances show that our proposed hybrid method delivers results that are competitive to those of other state-of-the-art algorithms for solving the two-stage fixed-charge transportation problem.

1. INTRODUCTION

Supply chains (SCs) are worldwide networks between a company and its suppliers, involving the following entities: suppliers, manufacturers, distribution centers, retailers, and customers. A classic SC executes the functions of procurement of raw materials, converting of those into intermediate and finished products, and finally the distribution of these products to customers, for more information we refer to [9, 15]. The main objective of a SC optimization problem is to satisfy the customer demands, while fulfilling the constraints imposed to the other actors of the SC: capacity restrictions that include manufacturers, distributions and transportation resources.

Supply chain management (SCM) is an important and central process, because an optimized supply chain will result in lower costs and faster production cycles. Network design plays a crucial role in achieving an efficient and effective management of SC systems. Typically, a SC can be represented as a form of multi-stage based structure, the optimal design of which has been recognized as an NP-hard problem [1].

In this work, we are focusing on a particular supply chain network design, namely the capacitated fixed-cost transportation problem (FCTP) in a two-stage supply chain network. We focus on identifying and selecting the manufacturers and the distribution centers fulfilling the demands of the customers under minimal costs.

The two-stage transportation problems have been introduced by Geoffrion and Graves [4]. Since then several versions of the problem have been considered and various solution approaches based on exact and heuristic algorithms have been proposed, see for example [10, 16].

One of these versions involves just one manufacturer and was introduced by Molla-Alizadeh-Zavardehi et al. [6]. They provided a mathematical model based on integer programming for the problem and as well developed a spanning tree-based genetic algorithm and an artificial immune algorithm for solving it. Some comments regarding the mathematical model of the problem were published by El-Sherniny [3] and Pintea et al.

Received: 24.12.2016. In revised form: 04.04.2017. Accepted: 15.04.2017

2010 *Mathematics Subject Classification.* 90C08.

Key words and phrases. *Transportation problem, supply chain, two-stage fixed-charge transportation problem, genetic algorithm.*

Corresponding author: Cosmin Sabo; cosmin_sabo@cunbm.utcluj.ro

[7] proposed some hybrid classical approaches for solving the problem. Recently, Pintea et al. [9] described an improved hybrid algorithm combining the Nearest Neighbor search heuristic with a local search procedure and Pop et al. [12] proposed novel hybrid heuristic approach obtained by combining a genetic algorithm based on a hash table coding of the individuals with a powerful local search procedure.

Another version of the two-stage transportation problem considers its impact on the environment by limiting the greenhouse gas emissions and was proposed by Santibanez-Gonzalez et al. [2]. For this variant of the problem, Pintea et al. [8] provided some classical hybrid heuristic approaches and Pop et al. [11] an efficient reverse distribution system for solving it.

In the form considered in our paper, the problem was introduced by Jawahar and Balaji [5]. They described a genetic algorithm (GA) with a specific coding scheme suitable for two-stage problems. The same authors introduced benchmark instances and their results have been compared to lower bounds and approximate solutions obtained from a relaxation. Raj and Rajendran [13] developed a two-stage genetic algorithm (TSGA) in order to solve a two-stage transportation problem with two different scenarios, where one scenario represents the FCTP that we consider in this work. They also proposed a solution representation that allows a single-stage genetic algorithm (SSGA) [14] to solve it. The major feature of these methods is a compact representation of a chromosome based on a permutation.

Our paper is organized as follows: In Section 2 we define the capacitated fixed-cost transportation problem in a two-stage supply chain network. Section 3 describes the developed hybrid genetic algorithm for solving this problem. The proposed algorithm is applied on the benchmark instances from Jawahar and Balaji [5] in Section 4, where the obtained results are presented and analyzed. Finally, in the last section, we summarize the obtained results in this paper and present future research directions.

2. The Two-Stage Fixed-Charge Transportation Problem

The capacitated fixed-cost transportation problem in a two-stage supply chain network is defined as follows. Given are a set of p manufacturers, a set of q distribution centers (DC's) and a set of r customers with the following properties:

- Each manufacturer can ship to any distribution center at a transportation cost C_{ij} per unit from manufacturer $i \in \{1, ..., p\}$ to DC $j \in \{1, ..., q\}$, plus a fixed cost F_{ij} for operating the route.
- Each DC can ship to any customer at a transportation cost C_{jk} per unit from DC $j \in \{1, ..., q\}$ to customer $k \in \{1, ..., r\}$, plus a fixed-cost F_{jk} for operating the route.
- Each manufacturer $i \in \{1, ..., p\}$ has S_i units of supply, each DC $j \in \{1, ..., q\}$ has SC_j units of stocking capacity and each customer $k \in \{1, ..., r\}$ has a demand D_k.

the aim of the two-stage capacitated fixed-cost transportation problem is to determine the routes to be opened and corresponding shipment quantities on these routes, such that the customer demands are fulfilled, all shipment constraints are satisfied, and the total distribution costs are minimized.

An illustration of the investigated two-stage fixed-charge transportation problem is presented in the next figure.

3. Hybrid Genetic Algorithm

In the following we describe the main features of the proposed hybrid genetic algorithm. Our approach consists of a steady-state genetic algorithm (GA) and an embedded

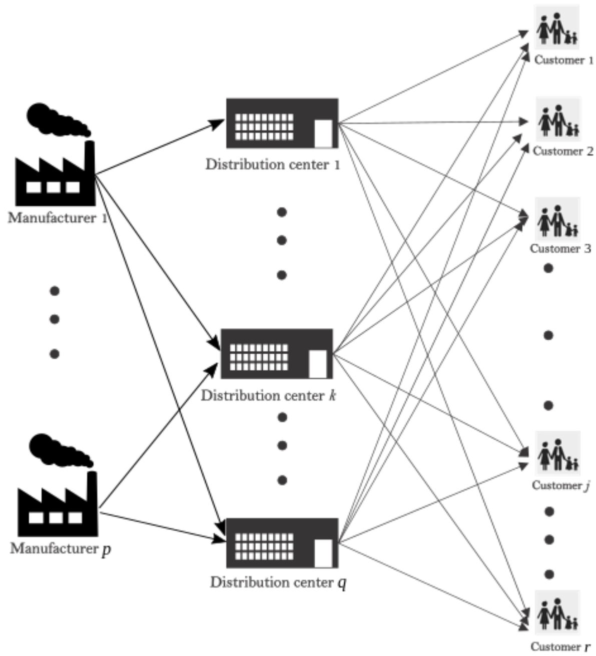

FIGURE 1. Illustration of the two-stage fixed-charge transportation problem

local search. It is based on an incomplete solution representation as only the transported amounts of good from the customers to the DCs are stored and optimized by the GA, while the amount transported from the manufacturers to the DCs are always implicitly derived in a second stage.

The objective function can be stated as

$$\min Z = Z^1 + Z^2$$

with

$$Z^1 = \sum_{j=1}^{q}\sum_{k=1}^{r}(C_{jk}Y_{jk} + F_{jk}\gamma_{jk}) \quad \text{and} \quad Z^2 = \sum_{i=1}^{p}\sum_{j=1}^{q}(C_{ij}X_{ij} + F_{ij}\delta_{ij}),$$

where the variables X_{ij} represent the quantities shipped from each manufacturer $i = 1, \dots, p$ to each DC $j = 1, \dots, q$, variables Y_{jk} represent the quantities shipped from each DC $j = 1, \dots, q$ to each customer $k = 1, \dots, r$, and the binary variables $\delta_{ij} \in \{0, 1\}$ and $\gamma_{jk} \in \{0, 1\}$ indicate whether or not a non-zero quantity is shipped from manufacturer i to DC j and DC j to customer k, respectively.

In order to represent candidate solutions, we only use the Y_{jk} variables. The values for the X_{ij} variables, which give us the second part of the cost function Z^2, are derived in a second-stage by using a greedy construction heuristic combined with a subsequent local search. This solution evaluation algorithm is described in Section 3.5.

3.1. **Initial Solutions.** For the initial population the following randomized greedy heuristic is employed. All of DC's supplies are iteratively assigned to customers in a locally best way. In each iteration, a not yet processed DC j is chosen randomly and the demand of customers is assigned to j until there is no supply left or all customer demands are satisfied. All not yet fully served customers are considered hereby in ascending order according to $F_{jk} + U_{jk}C_{jk}$, $\forall k = 1, \ldots, r$, where $U_{jk} = \min\{D_k, SC_j - \sum_{l=1, l \neq k}^{r} X'_{jl}\}$ is the amount of demand that can be satisfied by DC j with X'_{jl} is the amount of demand of customer l already assigned to DC j.

3.2. **Mutation.** The mutation operator changes the set of used DCs, which we denote by O. First, it chooses a DC, $l \in O$ which supplies at least one customer with a non-zero amount leaving us with a set $P = O \setminus \{l\}$ of prioritized DCs. Then, the assignments are deleted and the demand is reassigned by using an algorithm similarly to the greedy construction heuristic but instead of choosing the DCs randomly a random location from P is considered first until this set is empty.

3.3. **Crossover.** With the crossover operator the GA derives one new solution from two parent solutions. As we aim for building an offspring solution mostly from the properties, i.e., customer-DC assignments, appearing in its parents, an adapted uniform crossover is applied. Starting with an offspring that does initially not contain any assignments, the algorithm iterates over the parent solutions' assignments in a random order. When the assignment of either parent solution can be adopted in the offspring without violating feasibility with respect to the available capacity of the corresponding DC, one of them is selected uniformly at random. If only one assignment is feasible then that assignment is adopted, and if none is feasible the assignment of the remaining demand of the customer is postponed.

If all customer demands are satisfied the crossover operator is finished. Otherwise, the postponed customers are handled in random order as follows. The remaining demand of a postponed customer is assigned to one randomly chosen DC that can satisfy the whole remaining demand. If no such DC exists, as many DCs as necessary are iteratively selected at random until the customer's remaining demand is fulfilled.

3.4. **Local Search.** Both, the mutation and the crossover operator, can open new DCs so the purpose of the employed local search is to intensify the search for a fixed set of DCs. Therefore, each satisfied demand of each customer is tried to be moved to another opened DC that has enough supply left. As step function *best improvement* is chosen and the procedure is iterated until no further improvements regarding the objective value can be achieved.

3.5. **Solution Evaluation.** As already mentioned, we use an incomplete solution representation and the assignments of the DCs to the manufacturers is part of the solution evaluation. We solve this second-stage problem heuristically by using a similar randomized greedy heuristic and subsequent local search algorithm as for the customer-to-DC assignment problem. The initial solution construction heuristic is the same as for the original problem but we randomly open manufacturers and assign the DCs having a demand of $\sum_{k=1}^{r} X_{jk}$, $\forall j = 1, \ldots, q$ to them. After this step a local search is performed on the resulting solution which is based on an exchange neighborhood structure: each satisfied demand of each DC is tried to be satisfied by each other manufacturer. Note that due to the limited supply of the manufacturers infeasible assignments can be obtained. These infeasibilities are repaired in a second step. If a manufacturer i supplies too many DCs by being assigned the demand of DC j from manufacturer i' then it is tried to move each demand of each other assigned DC $j' \neq j$ back to the manufacturer i' until the assignment becomes feasible. As the original solution before the move was feasible this repair method always results in a feasible solution.

4. Computational Results

In order to asses the effectiveness of our proposed hybrid genetic algorithm, we conducted computational experiments on a set of 20 benchmark instances introduced by Jawahar and Balaji [5]. We performed 30 independent runs for each instance on a single core of an Intel Xeon processor with 2.54 GHz. Based on preliminary experiments, we have used the following parameter settings in our GA: population size 300, crossover rate 100%, mutation rate 10%, local search rate 10%, termination criterion 10,000 iterations without improvement or 30 seconds of run-time. Furthermore, the proposed algorithm uses a tournament selection of size 2, replaces in each iteration the solution candidate with the worst objective value, and does not allow duplicate solutions in the same population.

Table 1 shows the computational results of the proposed method (HGA) in comparison to those of the GA described by Jawahar and Balaji [5], called JRGA, and the two GAs by Raj and Rajendran [13], called TSGA and SSGA. The average objective values are given in the column obj along with the standard deviations sd. The values in the columns #eval correspond to the rounded average numbers of solution evaluations needed to find the best solution. Values in bold indicate the best existing solution with respect to that problem instance.

TABLE 1. Computational results of the proposed HGA compared to existing approaches.

	JRGA [5]		TSGA[13]	SSGA[13]		HGA		
	obj	#eval	obj	obj	#eval	obj	sd	#eval
2 x 2 x 3	**112600**	1444	**112600**	**112600**	637	**112600**	0	2
2 x 2 x 4	**237750**	1924	**237750**	**237750**	857	**237750**	0	2
2 x 2 x 5	**180450**	2404	**180450**	**180450**	1214	**180450**	0	319
2 x 2 x 6	**165650**	2884	**165650**	**165650**	1354	**165650**	0	324
2 x 2 x 7	**162490**	3364	**162490**	**162490**	1889	**162490**	0	335
2 x 3 x 3	**59500**	2164	**59500**	**59500**	1503	**59500**	0	317
2 x 3 x 4	**32150**	2884	**32150**	**32150**	1859	**32150**	0	339
2 x 3 x 6	69970	4324	67380	**65945**	2577	**65945**	0	356
2 x 3 x 8	263000	5764	**258730**	**258730**	5235	**258730**	0	546
2 x 4 x 8	80400	7684	84600	**77400**	5246	78550	0	1039
2 x 5 x 6	94565	7204	80865	**75065**	3574	80865	0	430
3 x 2 x 4	**47140**	2884	**47140**	**47140**	1429	**47140**	0	321
3 x 2 x 5	178950	3604	178950	**175350**	2061	178950	0	320
3 x 3 x 4	**57100**	4324	61000	**57100**	3060	**57100**	0	354
3 x 3 x 5	**152800**	5404	156900	**152800**	4555	**152800**	0	335
3 x 3 x 6	**132890**	6484	**132890**	**132890**	2981	**132890**	0	3
3 x 3 x 7 (a)	104115	7564	106745	**99095**	7095	103815	0	1330
3 x 3 x 7 (b)	287360	7564	295060	**281100**	7011	**281100**	0	380
3 x 4 x 6	77250	8644	81700	**76900**	7105	77250	0	373
4 x 3 x 5	**118450**	7204	**118450**	**118450**	4227	**118450**	0	394

Analyzing the computational results reported in Table 1, we can observe that our approach compares favorably to JRGA [5] and TSGA [13]. Compared to the SSGA [13], our algorithm delivered the same solution in 15 out of the 20 considered instances. We would like to emphasize the fact that the number of solution evaluations, including the evaluations within the local search, in order to obtain the best solutions by our proposed hybrid algorithm is significantly lower compared to the number of solutions enumerated to obtain the corresponding solutions by the other approaches. Also, the median run-time is less than three seconds on all instances. The observation that the HGA finds the same solution in each run of an instance confirms the robustness of the proposed approach.

5. Conclusions

This paper considers the two-stage fixed-charge transportation problem which models an important transportation application in a supply chain, from manufacturers to customers through distribution centers. For solving this optimization problem we developed a hybrid algorithm which combines a steady-state genetic algorithm with an incomplete solution representation and a heuristic second-stage decoder with local search. Computational results show that our hybrid genetic algorithm is robust and compares favorably to existing approaches. The new algorithm yields high-quality solutions in run-times of just a few seconds.

In future, we plan to asses the generality and scalability of the proposed hybrid heuristic approach by testing it on larger instances. It would also be promising to investigate advanced techniques for solving the second-stage problem, in particular also exact solution methods for example integer linear programming.

References

[1] Chen, S. Y., Zheng, Y. J., Cattani, C. and Wang, W. L., *Modeling of biological intelligence for SCM system optimization*, Comput. Math. Methods Med. **2012**, Art. ID 769702, 10 pp.

[2] Del, R., Santibanez-Gonzalez, E., Robson Mateus, G. and Pacca Luna, H., *Solving a public sector sustainable supply chain problem: A Genetic Algorithm approach*, Proc. of Int. Conf. of Artificial Intelligence (ICAI), Las Vegas, USA, 507-512, 2011

[3] El-Sherbiny, M. M., *Comments on "Solving a capacitated fixed-cost transportation problem by artificial immune and genetic algorithms with a Prüfer number representation" by Molla-Alizadeh-Zavardehi, S. et al. Expert Systems with Applications (2011)*, Expert Syst. Appl., **39** (2012), 11321–11322

[4] Geoffrion, A. M. and Graves, G. W., *Multicommodity distribution system design by Benders decomposition*, Management Sci., **20** (1974), 822–844

[5] Jawahar, N. and Balaji, A., *A genetic algorithm for the two-stage supply chain distribution problem associated with a fixed charge*, European J. Oper. Res., **194** (2009), No. 2, 496–537

[6] Molla-Alizadeh-Zavardehi, S., Hajiaghaei-Kesteli, M. and Tavakkoli-Moghaddam, R., *Solving a capacitated fixed-cost transportation problem by artificial immune and genetic algorithms with a Prüfer number representation*, Expert Syst. Appl., **38** (2011), 10462–10474

[7] Pintea, C.-M., Pop Sitar, C., Hajdu-Macelaru, M. and Pop, P. C., *A Hybrid Classical Approach to a Fixed-Charge Transportation Problem*, In Proc. of HAIS 2012, Part I, Editors E. Corchado et al., Lecture Notes in Computer Science, Springer, 7208, 557-566 (2012)

[8] Pintea, C.-M., Pop Sitar, C. and Hajdu-Macelaru, M., *Classical Hybrid Approaches on a Transportation Problem with Gas Emissions Constraints*, Adv. Intelligent and Soft Computing, **188** (2013), 449–458

[9] Pintea, C. M., and Pop, P. C., *An improved hybrid algorithm for capacitated fixed-charge transportation problem*, Logic J. IJPL, **23** (2015), No. 3, 369–378

[10] Pirkul, H. and Jayaraman, V. A., *Multi-commodity, multi-plant, capacitated allocation problem: formulation and efficient heuristic solution*, Comput. Oper. Res., **25** (1998), 869–878

[11] Pop, P. C., Pintea, C.-M., Pop Sitar, C. and Hajdu-Măcelaru, M., *An efficient Reverse Distribution System for solving sustainable supply chain network design problem*, J. Appl. Logic, **13** (2015), No. 2, 105–113

[12] Pop, P. C., Matei, O., Pop Sitar, C. and Zelina, I., *A hybrid based genetic algorithm for solving a capacitated fixed-charge transportation problem*, Carpathian J. Math., **32** (2016), 225–232

[13] Raj, K. A. A. D. and Rajendran, C., *A genetic algorithm for solving the fixed-charge transportation model: Two-stage problem*, Comput. Oper. Res., **39** (2012), No. 9, 2016–2032

[14] Raj, K. A. A. D. and Rajendran, C., *A Hybrid Genetic Algorithm for Solving Single-Stage Fixed-Charge Transportation Problems*, Techn. Oper. Manag., **2** (2011), No. 1, 1–15

[15] Simchi-Levi, D., Kaminsky, P. and Simchi-Levi, E., *Designing and Managing the Supply Chains: Concepts, Strategies and Case Studies*, 3-rd Edition, 2016

[16] Syarif, A., Yun, Y. and Gen, M., *Study on multi-stage logistic chain network: a spanning tree-based genetic algorithm approach*, Comput. Ind. Eng., **43** (2002), 299–314

[1] Technical University of Cluj-Napoca
North University Center at Baia Mare
Department of Mathematics and Computer Science
Victoriei 76, Baia Mare, Romania
E-mail address: petrica.pop@cunbm.utcluj.ro
E-mail address: cosmin_sabo@cunbm.utcluj.ro

[2] AIT Austrian Institute of Technology
Center for Mobility Systems - Dynamic Transportation Systems
Giefinggasse 2, 1210, Vienna, Austria
E-mail address: benjamin.biesinger@ait.ac.at
E-mail address: bin.hu@ait.ac.at

[3] Institute of Algorithms and Computer Graphics
TU Wien, Austria
E-mail address: raidl@ac.tuwien.ac.at