

A multiobjective non-dominated sorting genetic algorithm (NSGA-II) for the Multiple Traveling Salesman Problem

Rubén Iván Bolaños^a, Mauricio Granada Echeverry^a and John Willmer Escobar^{b*}

^aDepartment of Electrical Engineering, Universidad Tecnológica de Pereira, Colombia

^bDepartment of Civil and Industrial Engineering, Pontificia Universidad Javeriana Cali, Colombia

CHRONICLE

Article history:
Received March 29, 2015
Received in revised format:
May 12, 2015
Accepted May 12, 2015
Available online
May 14 2015

Keywords:
Multi-objective Optimization
Multi-Traveling Salesman
Problem
Population-based Algorithm
Local Search Operators

ABSTRACT

This paper considers a multi-objective version of the Multiple Traveling Salesman Problem (MOmTSP). In particular, two objectives are considered: the minimization of the total traveled distance and the balance of the working times of the traveling salesmen. The problem is formulated as an integer multi-objective optimization model. A non-dominated sorting genetic algorithm (NSGA-II) is proposed to solve the MOmTSP. The solution scheme allows one to find a set of ordered solutions in Pareto fronts by considering the concept of dominance. Tests on real world instances and instances adapted from the literature show the effectiveness of the proposed algorithm.

© 2015 Growing Science Ltd. All rights reserved.

1. Introduction

The Multiple Traveling Salesman Problem (MTSP) is a generalization of the well-known Traveling Salesman Problem (TSP) arising when several Hamiltonian tours are considered. The MTSP is NP-hard and has many applications in Transportation and Logistics (routing of school buses, mail delivery, etc.). The literature related to the MTSP and its practical applications has been reviewed in Bektas (2006). In this work, different formulations are highlighted, and exact and different heuristics methods for solving the problem are described. Early works that consider the MTSP have been proposed by Hong and Padberg (1977) and Rao (1980). Hong and Padberg (1977) considered the symmetric version of the MTSP with fixed charges (FCSmTSP). The FCSmTSP is reformulated as a standard Symmetric Traveling Salesman problem (STSP). Rao (1980) considered the symmetric and asymmetric versions of the MTSP with fixed charges. In this work, both problems are transformed into their corresponding standard versions of the TSP. Laporte and Nobert (1980) proposed an exact method for the MTSP by considering the sub-tour elimination and the integrality constraints. The algorithm proposed by Gavish

* Corresponding author.

E-mail address: johnwillmer.escobar2@unibo.it (J. W. Escobar)

and Srikanth (1987) considers that all the traveling salesmen must be used to the performed Hamiltonian cycles. This assumption is also considered by Jonker and Volgenant (1988). In this work, a generalization of the standard formulation for the MTSP is proposed by considering Euclidean and non-Euclidean instances.

Population-based metaheuristic algorithms for the MTSP were proposed by Sofge et al. (2002), Junjie and Dingwei (2006), Zhao et al. (2008), Király and Abonyi (2011) and Yu et al. (2012). Heuristics based on local search have been proposed by Hou and Liu (2012) and Levin and Yovel (2012). Successful implementations of the MTSP for real world problems have been proposed by Tang et al. (2000) and by Kergosien et al. (2009). Tang et al. (2000) considered hot rolling scheduling in a steel company in China. A modified genetic algorithm was used to solve the problem. Kergosien et al. (2009) considered the problem of assigning patients to nurses. A mixed integer linear programming model was proposed to solve the real world case. However, only few works have considered the multi-objective version of the MTSP (MOMTSP). Chang and Yen (2012) considered the routing problem of the courier service for a city. The problem is formulated as a MOMTSP by considering hard time windows (MOMTSPTW). A multi-objective scatter search is proposed to minimize the operating costs and to improve the level of service. The proposed method finds a set of Pareto-optimal solutions.

Shim et al. (2012) presented a mathematical formulation of the MOMTSP which considers the objective function as the minimization of the weighted sum of the global distance traveled by the traveling salesmen and the distance traveled by a salesman. An estimation of distribution algorithm (EDA) with a gradient search was used for the solution of the considered problem. Finally, Labadie et al. (2014) considered a version of MOMTSP in which the objective is to determine m cycles by covering a set of potential customers maximizing the corresponding benefit and minimizing the total traveled distance. An evolutionary algorithm with a Path-relinking (PR) strategy is proposed to solve the problem.

In this paper, we introduce a new version of the MOMTSP by considering two objectives: i) the minimization of the total distance traveled and ii) the balance of the times traveled by the salesmen. We propose a multi-objective non-dominated sorting genetic algorithm NSGA-II for the considered problem. The solution scheme allows one to find a set of solutions ordered under the concept of Pareto dominance, coupled with a selection scheme of elite solutions by stacking distances on each Pareto front and a geometric distribution over the whole range of Pareto fronts. At each generation cycle, the proposed algorithm allows one to generate a different population. In addition, a local search strategy is applied, which uses different neighborhood structures such as Shift (1,0), Swap (1,1) and 2-Opt moves. The integration of this local search strategy with the traditional methodology of NSGA-II and the use of diversity strategies allow one to obtain high quality results. Computational experiments on real-world instances and on adapted instances from the literature show the effectiveness of the proposed algorithm.

2. Problem Description

The MTSP could be defined by a complete (directed or undirected) graph $G(V, A)$, where $V \setminus \{0\}$ is the set of n nodes to be visited (customers), node 0 is the depot, and A is the set of arcs connecting the nodes. A c_{ijk} value is associated with each arc (i, j) , which is traversed by the salesman k . The matrix C is symmetric when an undirected graph is considered and asymmetric otherwise. The traveled distance of a tour is calculated as the sum of the lengths of the arcs belonging to the tour.

In addition, the travel time of a salesman for each arc (i, j) of A is given by t_{ijk} . The travel time of a tour (i.e. the working time of a salesman) is calculated as the sum of the travel times of the arcs belonging to the tour.

Given an integer m , the addressed problem consists of finding m tours passing through the depot node.

Each node must be visited by a tour (i.e. by a traveling salesman) only once, so that the total traveled distance is minimized and the working times of the traveling salesmen are similar each other.

2.1 Multi-objective optimization

A multi-objective optimization problem considers several conflicting objectives. This means that a high quality solution for one of the objectives could be a poor quality solution for another objective. In particular, the traditional optimization methods do not provide solutions that are good for all the objectives of the considered problem. A multi-objective optimization problem can be formulated through a mathematical model defined by a set of h objective functions which must be minimized or maximized, subject to a set of m inequality constraints, a set of l equality constraints and lower and upper limits in the k decision variables, as illustrated below:

$$\begin{aligned} & \min / \max f_1(X) \\ & \min / \max f_2(X) \\ & \quad \vdots \\ & \quad \vdots \\ & \quad \vdots \\ & \min / \max f_h(X) \end{aligned}$$

subject to

$$\begin{aligned} g_i(X) &\leq 0; \forall i = 1, \dots, m \\ h_i(X) &= 0; \forall i = 1, \dots, l \\ x_i^l &\leq x_i \leq x_i^u; \forall i = 1, \dots, k \end{aligned}$$

3. Mathematical model

The MOmTSP can be formulated as a multi-objective integer linear model with two objective functions. The first objective function (1) considers the minimization of the distance traveled by all the salesmen, while the second objective function (2) considers the balance of the working times of the salesmen, i.e., the travel times of the salesmen are similar each other. With each arc (i, j) of G , is associated a binary variable x_{ijk} , which takes the value of 1 if arc (i, j) was traversed by the salesman k into the final solution and 0 otherwise.

$$\min Z_1 = \sum_{k=1}^m \sum_{i=1}^n \sum_{j=1}^n c_{ijk} x_{ijk} \tag{1}$$

$$\min Z_2 = \sum_{k=1}^m |t_{Avg} - t_k| \tag{2}$$

where:

t_{Avg} is the average travel time of the tours defined by Eq. (3) as follows,

$$t_{Avg} = \frac{\sum_{k=1}^m \sum_{i=1}^n \sum_{j=1}^n t_{ijk} x_{ijk}}{m} \tag{3}$$

where t_k is the time spent on each tour k ($k = 1, \dots, m$) defined by Eq. (4).

$$t_k = \sum_{i=1}^n \sum_{j=1}^n t_{ijk} x_{ijk}; \quad \forall k = 1, \dots, m \tag{4}$$

In addition, the problem is subject to the following set of constraints:

$$\sum_{k=1}^m \sum_{i=1}^n x_{ijk} = 1; \quad \forall j = 1, \dots, n \quad (5)$$

$$\sum_{k=1}^m \sum_{j=1}^n x_{ijk} = 1; \quad \forall i = 1, \dots, n \quad (6)$$

$$\sum_{k=1}^m \sum_{i=1}^n x_{i0k} = m \quad (7)$$

$$\sum_{k=1}^m \sum_{j=1}^n x_{0jk} = m \quad (8)$$

$$\sum_{i \in S} \sum_{j \notin S} x_{ijk} \geq 1; \quad \forall k = 1, \dots, m, \quad \forall S \subseteq V; 0 \in S \quad (9)$$

$$x_{ijk} \in \{0,1\} \quad (10)$$

Eq. (5) and Eq. (6) are related to the degree restrictions and ensure that each node, except the depot node, is visited only once by a single traveling salesman. Eq. (7) and Eq. (8) ensure that each traveling salesman must leave and return to the depot node. Constraints (9) called connectivity constraints avoid sub-tours generation on the final solution. Finally, constraints (10) impose that the decision variables are binary.

4. Non-dominated sorting genetic algorithm (NSGA-II)

One of the most appropriate strategies to determine a good approximate solution of an NP-hard multi-objective problem is a metaheuristic algorithm based on population which evolves along the solution space to find a set of non-dominated solutions. In this paper, we propose a methodology based on the non-dominated sorting genetic algorithm NSGA-II introduced by Deb (2001). The main objective of algorithm NSGA-II is to find a set of solutions ordered by fronts under the concept of Pareto dominance.

The algorithm starts with a population of feasible solutions (parents) of size N (where N is a given parameter), called P_t , which is generated randomly by an initialization procedure. From P_t , a population Q_t , of feasible solutions (descendants) of size N (where N is a given parameter) is created by using selection, recombination and mutation operators. In this paper, the population of descendants is generated by using an event selection according to the objective function Z_1 described in the previous section. After that, a recombination process and an improvement process (mutation operator) over the solution resulting from the recombination process are performed. Finally, a combination of the population of parents and descendants is performed to obtain a population R_t of size $2N$ (where N is a given parameter). The population R_t is organized in Pareto fronts, which are sorted and ranked from the front having the best quality to the front having the worst quality (F_1, F_2, \dots, F_N) , by using the concept of Pareto dominance.

4.1 Concept of dominance

The concept of dominance is applied to multi-objective problems to compare two solution candidates X_1, X_2 , and determine if a solution dominates the other one. In particular, the dominance is a method for the classification of the solutions which ensures the selection of the best solution in the resulting population R_t .

Definition 4.1. Given two solutions X_1 and X_2 , solution X_1 dominates solution X_2 , if the following conditions are satisfied:

1. solution X_1 is not worse than X_2 for all the objectives;
2. solution X_1 is strictly better than X_2 for at least one objective.

When a complete set R_t is classified according to the above criteria, the best dominant N solutions are selected for making the population P_{t+1} . This process is performed by a controlled elitist selection, which allows each front to have at least one solution representing it in the next generation.

4.1.1 Controlled elitism in NSGA-II

We propose to keep a predetermined distribution of solutions in order to maintain a different population in each generation of the evolutionary process. Therefore, each Pareto front provides solutions to the next generation. In particular, a geometric distribution is used. From the population R_t of size $2N$ organized in k non-dominated fronts, the maximum number N_i of solutions of front i ($i = 1, \dots, k$) which pass to the new population P_{t+1} of size N is calculated by the Eq. (11):

$$N_i = Nr^{i-1} \frac{1-r}{1-r^k} \quad (11)$$

where r is a reduction factor ($r < 1$).

Since $r < 1$, the first Pareto front provides the largest number of candidate solutions of the new generation, and the following fronts contribute with a number of solution which is reduced exponentially. In addition, there are two considerations concerning to the use of Eq. (11). The first issue arises when front i contains a number of solutions N_i^t less than N_i , therefore the solutions of front i are considered for the next generation of solutions. The number of solutions which must fill front i are calculated as $\rho_i = N_i - N_i^t$, and the maximum number of solutions provided by front $i + 1$ is increased to $N_{i+1} + \rho_i$. This process is performed until N individuals are obtained. The second issue occurs when front i contains a number of solutions N_i^t greater than N_i . In this case only N_i solutions which have been allowed on front i by using the concept of "stacking distances" are chosen (see Deb (2001) and the following sections). These aspects show that the use of the geometric distribution contributes to the lateral range of the selected solutions, while the concept of stacking distances provides diversity on each front, as shown in Figure 1.

4.1.2 Stacking Distances

The selection of the maximum number of solutions for each front i is performed by using the concept of "stacking distances". In particular, the main objective is to determine the density of the solution space for a given solution X_i by considering the average distance between two neighborhood solutions located on both sides of X_i for each objective function. The stacking distance is an estimate of the perimeter of the cuboid by using the pair of nearest neighbors of solution X_i as its vertices. The solutions found at the end of the front take a stacking distance equal to infinity.

The concept of stacking distance allows one to rank the solutions belonging to a specific Pareto front i . Therefore, the solutions having the largest diversity (i.e., the solutions which the least populated zones of the front) are chosen. Figure 1 shows an overview of the proposed algorithm. The Figure 1 desirables a classification by Pareto fronts of a complete set R_t and the process of selecting elite solutions which belong to population P_{t+1} .

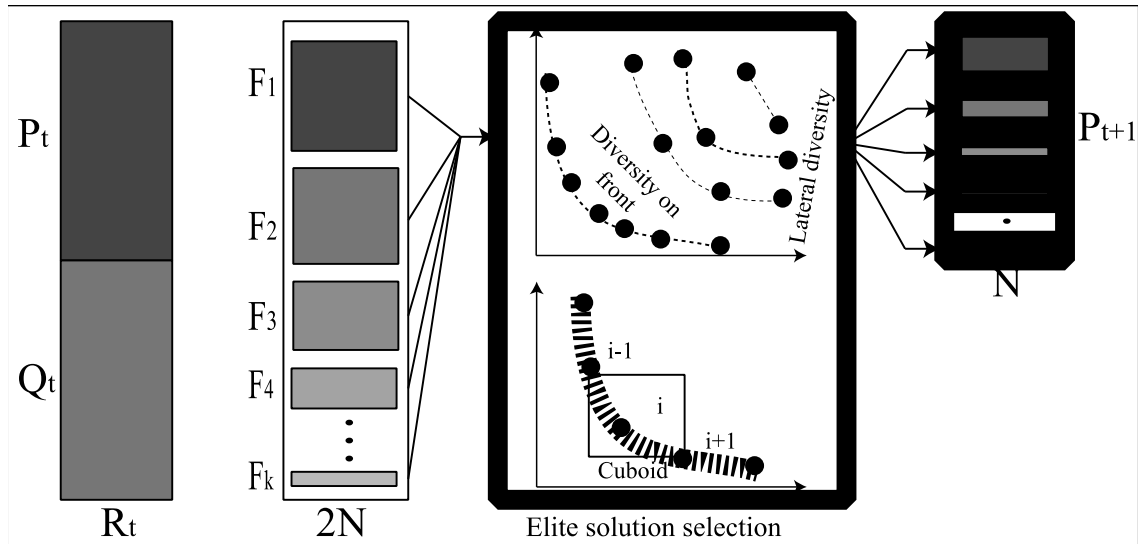


Fig. 1. NSGA-II Algorithm

4.2 Initial population P_t

The initial population P_t is composed of N approximate solutions of the classical TSP obtained by applying the heuristic algorithm proposed by Christofides (1976) and by considering: a) The complete set of nodes except the depot (Escobar et al., 2013), and b) A different starting node at each execution of the heuristic algorithm. This strategy obtains N sequences (permutations) of nodes called giant TSP's. The steps for constructing each solution of the initial population are:

- i. The giant tour is split into m subsequences of nodes by using a sweeping strategy, where the number of nodes belonging to each subsequence is chosen randomly. The nodes of each subsequence are visited according to the order established by the giant tour. The two extreme nodes of each subsequence are connected with the depot node.
- ii. Each solution is evaluated according to Eq. (1) and Eq. (2) in order to obtain the initial value for each of the two objectives considered in the considered problem.

4.3 Population of descendants Q_t

The population of descendants Q_t is obtained from the population P_t by considering several genetic operators.

- **Tournament selection:** From population P_t five solutions are chosen randomly. Then, these solutions are ordered from the best to the worst quality according to the objective function Z_1 . Two solutions having the best quality are selected to be the parents of a single descendant, which is obtained by using recombination and mutation genetic operators.
- **Recombination:** Different methods for recombining sequences for the TSP have been proposed in the literature. These methods try to obtain feasible tours, so that each node is visited exactly once by one of the tours. Among the most used methods, we mention PMX (Goldberg and Lingle, 1985), OX (Oliver et al., 1987), OX2 (Syswerda, 1991), and PBX (Syswerda, 1991). In this paper, we use the recombination method PMX, because it is a coding based on permutations.
- **Mutation:** In this paper, the concept of mutation is applied by using neighborhood structures, while the goal of finding a neighborhood of good quality to improve one of the two objectives

of the considered problem. The mutation is applied as an improvement strategy after the recombination method. The following neighborhood structures are considered:

1. **Shif (1-0)**. A node is removed from its current position, and is inserted in a different position of a different tour.
2. **Swap (1-1)**. Two nodes (belonging to different tours) exchange their positions.
3. **2-Opt**. Two non-adjacent arcs belonging to the same tour are removed, and two new arcs are added in order to find a new tour. The nodes involved in the moves are selected randomly.

The algorithm 1 shows the pseudocode of the proposed approach.

5. Computational results

The proposed algorithm has been implemented in C++ on a CPU Intel Core 2 Duo with 3 GHz and 2 GB of RAM. The parameters used for the proposed algorithm are: the rate of recombination is 95%, the number of iterations is 30, and the size N of the initial population is 50.

The proposed methodology is validated with two instances. The first instance (F1) concurs a real world Colombian courier company in Cali with 29 nodes to be visited (customers) and one depot. Three salesmen are considered ($m=3$).

The second instance (F2) is adapted from the literature. In particular, we have considered initially the instance pr76 proposed in Reinelt (2014), which has 75 customers and one depot. The travel time for each arc has been defined by considering a speed on the arc randomly generated between 20 km/ hour and 90 km / hour. Three salesmen are considered ($m=3$).

```

Data: Instance size  $n$ , population size  $N$ , probability  $\rho_m$  and  $\rho_c$ , maximum cities per
        route  $c_{max}$ ,  $P \leftarrow \{\emptyset\}$ 
Result: Feasible optimal solution  $\Pi$  to the the mTSP
initialization;
while  $|P| \neq N$  do
     $P \leftarrow P \cup$  chromosomes from nearest neighbor, Christofides algorithm and
    Lin-Kernighan heuristic;
end
Calculate the minimum of tours ( $TSP_{min}$ );
Calculate the minimum of nodes per tour ( $c_{min}$ );
for  $i \leftarrow 1 \dots N$  do
    Randomly split each chromosome with the tour size between  $c_{min}$  and  $c_{max}$ ;
end
while  $\neg$  stopping criteria (the fitness of the best solution has converged or a
maximum number of generations has been reached) do
     $p_1, p_2 \leftarrow$  tournament-selection( $P$ );
    if  $(p_1 \neq p_2)$  and  $(rand() \leq \rho_c)$  then
         $h \leftarrow$  Crossover( $p_1, p_2$ );
    else
         $h \leftarrow p_1$ ;
    end
    if  $(rand() \leq \rho_m)$  then
         $h \leftarrow$  local-search( $h$ );
        Place  $h$  into the population;
        generation ++
    else
        Place  $h$  into the population;
        generation ++
    end
end
return The best solution ( $P$ );

```

Algorithm 1. Pseudo code for solving the MOmTSP

The Fig. 2 and Fig. 3 show the Pareto fronts found by the proposed methodology for the instances F1 and F2, respectively. Each of the Pareto fronts offers a range of possibilities depending on the objectives for the proposed problem. In this case, the best solution depends on the specific interests of the decision maker. However, there are different ways to select the optimum solution Pareto front.

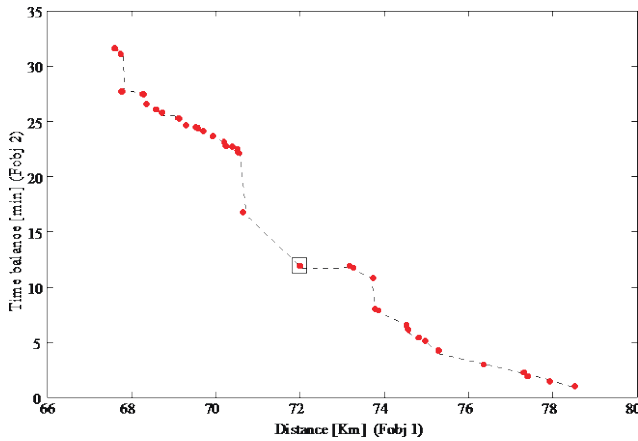


Fig. 2. Optimal Pareto Front for F1

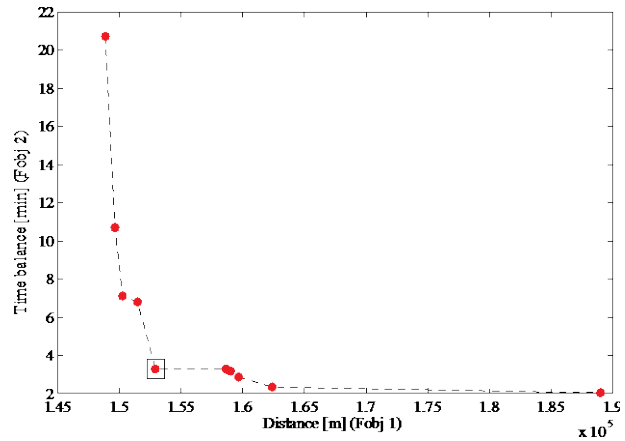


Fig. 3. Optimal Pareto Front for F2

5.1 Selecting a solution of the Pareto front

The criterion max-min is used to select a solution from each Pareto optimal front, in order to show solutions that represent each set of data. This method finds a solution that is equidistant from the ends of each objective. The criterion is defined as follows:

$$\max \left[\min \left(\frac{FO_{1,max} - FO_{1,i}}{FO_{1,max} - FO_{1,min}}, \frac{FO_{2,max} - FO_{2,i}}{FO_{2,max} - FO_{2,min}} \right) \right] \quad (12)$$

where

$FO_{1,max}$ = Maximum value of the objective function 1,
 $FO_{1,min}$ = Minimum value of the objective function 1,
 $FO_{2,max}$ = Maximum value of the objective function 2,
 $FO_{2,min}$ = Minimum value of the objective function 2,
 $FO_{1,i}$ = Value of the objective function 1 for solution i ,
 $FO_{2,i}$ = Value of the objective function 2 for solution i .

The criterion max-min selects from each Pareto front a specific solution (enclosed in boxes in Figures 2 and 3), whose characteristics are shown in Tables 1 and 2.

Table 1

Chosen solution of the Pareto front for F1

Traveling Salesman	Distance [km]	Travel time [min]	Fobj 1	Fobj 2
1	24.9	74.4		
2	23.8	63.0	71.8	11.9
3	23.1	69.5		

Table 2

Chosen solution of the Pareto front for F2

Traveling Salesman	Distance [km]	Travel time [min]	Fobj 1	Fobj 2
1	53353	61.0		
2	47177	62.8	152888	3.3
3	52358	59.6		

6. Conclusion

In this paper, a methodology to solve the Multi-objective Multiple Traveling Salesman Problem is presented by implementing a Non-dominated Sorting Genetic Algorithm (NSGA-II). In order to evaluate the performance of the proposed approach, tests have been performed on two instances. The results show a set of non-dominated solutions organized in a Pareto optimal front, reflecting the conflict between the two considered objectives. Therefore, the proposed approach provides the opportunity to choose any of the solutions of the front according to the criteria chosen by the decision maker. In addition, a criterion for choosing a good quality solution for each instance is proposed. The results show the minimization of the traveled distances and the balance of the work times for the traveling salesmen.

References

- Bektas, T. (2006). The multiple traveling salesman problem: an overview of formulations and solution procedures. *Omega*, 34(3), 209-219.
- Chang, T. S., & Yen, H. M. (2012). City-courier routing and scheduling problems. *European Journal of Operational Research*, 223(2), 489-498.
- Christofides, N. (1976). *Worst-case analysis of a new heuristic for the travelling salesman problem* (No. RR-388). Carnegie-Mellon Univ Pittsburgh Pa Management Sciences Research Group.
- Deb, K. (2001). *Multi-objective optimization using evolutionary algorithms* (Vol. 16). John Wiley & Sons.
- Escobar, J. W., Linfati, R., & Toth, P. (2013). A two-phase hybrid heuristic algorithm for the capacitated location-routing problem. *Computers & Operations Research*, 40(1), 70-79.
- Gavish, B., & Srikanth, K. (1986). An optimal solution method for large-scale multiple traveling salesmen problems. *Operations Research*, 34(5), 698-717.
- Goldberg, D. E., & Lingle, R. (1985, July). Alleles, loci, and the traveling salesman problem. In *Proceedings of the first international conference on genetic algorithms and their applications* (pp. 154-159). Lawrence Erlbaum Associates, Publishers.
- Hong, S., & Padberg, M. W. (1977). Technical Note—A Note on the Symmetric Multiple Traveling Salesman Problem with Fixed Charges. *Operations Research*, 25(5), 871-874.
- Hou, M., & Liu, D. (2012). A novel method for solving the multiple traveling salesmen problem with multiple depots. *Chinese science bulletin*, 57(15), 1886-1892.
- Jonker, R., & Volgenant, T. (1988). Technical Note—An Improved Transformation of the Symmetric Multiple Traveling Salesman Problem. *Operations Research*, 36(1), 163-167.
- Junjie, P., & Dingwei, W. (2006, August). An ant colony optimization algorithm for multiple travelling salesman problem. In *Innovative Computing, Information and Control, 2006. ICICIC'06. First International Conference on* (Vol. 1, pp. 210-213). IEEE.
- Kergosien, Y., Lenté, C., & Billaut, J. C. (2009, August). Home health care problem: An extended multiple traveling salesman problem. In *4th Multidisciplinary International Conference on Scheduling: Theory and Applications (MISTA'09), Dublin (Irlande)* (pp. 10-12).
- Király, A., & Abonyi, J. (2011). Optimization of multiple traveling salesmen problem by a novel representation based genetic algorithm. In *Intelligent Computational Optimization in*

- Engineering* (pp. 241-269). Springer Berlin Heidelberg.
- Labadie, N., Melechovsky, J., & Prins, C. (2014). An Evolutionary Algorithm with Path Relinking for a Bi-objective Multiple Traveling Salesman Problem with Profits. In *Applications of Multi-Criteria and Game Theory Approaches* (pp. 195-223). Springer London.
- Laporte, G., & Nobert, Y. (1980). A cutting planes algorithm for the m-salesmen problem. *Journal of the Operational Research Society*, 31(11), 1017-1023.
- Levin, A., & Yovel, U. (2014). Local search algorithms for multiple-depot vehicle routing and for multiple traveling salesman problems with proved performance guarantees. *Journal of Combinatorial Optimization*, 28(4), 726-747.
- Oliver, I. M., Smith, D., & Holland, J. R. (1987). Study of permutation crossover operators on the traveling salesman problem. In *Genetic algorithms and their applications: proceedings of the second International Conference on Genetic Algorithms: July 28-31, 1987 at the Massachusetts Institute of Technology, Cambridge, MA*.
- Rao, M. (1980). A note on the multiple traveling salesman problem. *Operations Research*, 28 (3), 628–632.
- Reinelt, G. (2014). TSPLIB is a library of sample instances for the TSP (and related problems) from various sources and of various types. Available on-line at <http://comopt.ifl.uni-heidelberg.de/software/TSPLIB95/>.
- Shim, V. A., Tan, K. C., & Tan, K. K. (2012, June). A hybrid estimation of distribution algorithm for solving the multi-objective multiple traveling salesman problem. In *Evolutionary Computation (CEC), 2012 IEEE Congress on* (pp. 1-8). IEEE.
- Sofge, D., Schultz, A., & De Jong, K. (2002). Evolutionary computational approaches to solving the multiple traveling salesman problem using a neighborhood attractor schema. In *Applications of Evolutionary Computing* (pp. 153-162). Springer Berlin Heidelberg.
- Syswerda, G. (1991). Schedule optimization using genetic algorithms. *Handbook of genetic algorithms*, 332–349.
- Tang, L., Liu, J., Rong, A., & Yang, Z. (2000). A multiple traveling salesman problem model for hot rolling scheduling in Shanghai Baoshan Iron & Steel Complex. *European Journal of Operational Research*, 124(2), 267-282.
- Yu, Q., Wang, D., Lin, D., Li, Y., & Wu, C. (2012). A novel two-level hybrid algorithm for multiple traveling salesman problems. In *Advances in Swarm Intelligence* (pp. 497-503). Springer Berlin Heidelberg.
- Zhao, F., Dong, J., Li, S., & Yang, X. (2008, July). An improved genetic algorithm for the multiple traveling salesman problem. In *Control and Decision Conference, 2008. CCDC 2008. Chinese* (pp. 1935-1939). IEEE.