

PAPER • OPEN ACCESS

A new metaheuristics for solving vehicle routing problem: Partial Comparison Optimization

To cite this article: A Adhi *et al* 2019 *IOP Conf. Ser.: Mater. Sci. Eng.* **598** 012023

View the [article online](#) for updates and enhancements.



IOP | ebooks™

Bringing you innovative digital publishing with leading voices to create your essential collection of books in STEM research.

Start exploring the **collection** - download the first chapter of every title for free.

A new metaheuristics for solving vehicle routing problem: Partial Comparison Optimization

A Adhi¹², B Santosa¹ and N Siswanto¹

¹ Industrial Engineering Department Institut Teknologi Sepuluh Nopember Surabaya, Indonesia,

² Industrial Engineering Department Universitas Stikubank Semarang, Indonesia

anthonoadhi@edu.unisbank.ac.id

Abstract. Vehicle Routing Problem (VRP) is a problem of selecting shortest route from a depot to serve several nodes by considering transport capacity. In this study, a new metaheuristics algorithm is proposed to solve VRP in order to achieve optimal solution. This metaheuristics algorithm is Partial Comparison Optimization (PCO). This new optimization algorithm was developed to solve combinatorial optimization problems such as VRP. In this study, PCO was tested to solve the problems that existed in the origin VRP. To prove PCO is a good metaheuristics for solving VRP, several of instances of symmetrical VRP were selected from the VRP library to evaluate its performance. The numerical results obtained from the calculation indicated that the proposed optimization method could achieve results that almost similar with the best-known solutions within a reasonable time calculation. It showed that PCO was a good metaheuristics to solve VRP.

1. Introduction

Logistics is an important part in the manufacture company. One of the tasks of logistics is to send products from a company called a depot to customers who order these products. Although logistics is not a part that gives added value directly to a product, but the inability to manage logistics can cause high costs that disrupt the competitiveness of the company. Therefore, the management of shipping goods from depots to customers must be managed and carried out effectively and efficiently. Effective in the sense that all customer requests are fully met so that the product reaches the customer's hands. Efficiently means that the costs arising from the logistics process can be reduced as small as possible. One of the activities that can do this is to arrange transportation shipping routes from the depot to customers and from one customer to another customer for the smallest possible cost. This activity is known as the Vehicle Routing Problem (VRP).

In a logistics system, transportation plays an important role in bringing raw materials from suppliers to producers and sending finished goods from producers to customers [1]. In the US, transportation itself covers 28% of total energy consumption, and thus transportation costs significantly affect the final price of goods [2]. Because of the wide range of applications in both commercial and public entities, VRP is considered to be one of the most important problems in operational research.

VRP was first introduced by Dantzig and Ramser [3]. They researched the delivery of fuel from the depot to the fuel station. After their research, many researchers followed researchers about VRP. Improvements to the VRP problem are done to better solve this problem [4]. The conditions in the field also give variants that require different handling techniques. The variants of VRP that emerged



and were introduced were mostly still related to the addition of new constraints from the original VRP cases [4]. Constraints that are often added as variants in solving VRP problems are pallets and window time [4]. Within the pallet range, items of different sizes must be transported in standard size boxes and limited capacity [4]. Within the window time limit, items must be sent for a period of time [4].

VRP is a complex combinatorial optimization case [5]. He is a classic problem from the field of logistics and transportation relating to route planning for vehicles starting from the central depot to a set of customers [4]. Braysy and Gendreau [6] state that VRP is a problem of finding the route with the smallest cost from one depot to a number of points scattered in a geographical area in the form of cities, shops, warehouses, schools, customers, and so on. The route designed for each point will only be passed 1 time by 1 vehicle that starts and ends at a depot and the total demand for a route does not exceed the capacity of the vehicle.

VRP and its variants are classified into NP-hard problem [1][7][8]. In this case, the VRP cannot be solved by the exact method at the time of the possible calculations. The NP-Hard case can be solved by using the optimal solution approach with the heuristic method that provides an estimate of the solution with acceptable completion time. Therefore, VRP solutions can be done by the methods of heuristics or metaheuristics [9].

The most commonly used heuristic methods to solve NP-hard problems include evolutionary algorithms inspired by processes that occur in nature such as genetic algorithms, particle swarm optimization, differential evolution, ant colony optimization, and so on. Each algorithm is tested and compared to find out which algorithm is the best and suitable to be applied in order to overcome an NP-hard problem such as VRP. There is still a lot of research on VRP. Many of them are solved by the metaheuristics algorithm such as simulated annealing [1], genetic algorithm [10][11][12], particle swarm optimization [13], variable neighborhood search [12][14][15][16][17], ejections chains neighborhood [18], bee colony [19], tabu search [20], ant colony [9], symbiotic organisms search [1].

In the research that will be carried out, a new method will be developed that will be used to solve VRP problems and provide optimum solutions. This method is called Partial Comparison Optimization (PCO). Some of others metaheuristics is a continues metaheuristics while PCO is a purely discrete metaheuristics. It gives an advantage to PCO not to convert continues problem to discrete problem such as VRP. And PCO is rarely trapped in local optimum therefore PCO will give better solution compared with others metaheuristics. With the implementation of PCO, VRP will be solved more effective.

2. Problem Statement

VRP can be formulated as follows [21]: if x_{ij} ($i \neq j$) is a binary variable equal to 1 if and only if there is an arc (i, j) of A' .

$$\text{Minimize } \sum_{i \neq j} C'_{ij} x_{ij} \quad (1)$$

s.t.

$$\sum_{j=1}^{n'} x_{ij} = 1, \quad (i = 1, \dots, n') \quad (2)$$

$$\sum_{i=1}^{n'} x_{ij} = 1, \quad (j = 1, \dots, n') \quad (3)$$

$$\sum_{i,j \in S} x_{ij} \leq |S| - v(S), (S \subset V \setminus \{1\}; |S| > 2) \quad (4)$$

$$x_{ij} \in \{0,1\}, \quad (i, j = 1, \dots, n'; i \neq j) \quad (5)$$

(1), (2), and (3) define assignments. Distance on diagonals is not allowed. Constraint (4) is an obstacle to eliminating subtour: $v(S)$ is the lower limit of the number of vehicles needed to visit all S vertices in the optimal solution. These constraints are obtained by observing that for every $S \subset V \setminus \{1\}; |S| > 2, S = V \setminus V$, we must have:

$$\sum_{i \in S} \sum_{j \in S} x_{ij} > v(S), \quad (i \neq j) \quad (6)$$

and that the following identity applies:

$$|S| = \sum_{i, j \in S} x_{ij} + \sum_{i \in S} \sum_{j \in S} x_{ij}, \quad (i \neq j) \quad (7)$$

Value of $v(S)$ is dependent on the type of considered VRP. For type capacitated VRP (CVRP), the taken value is:

$$v(S) = \left\lceil \frac{\sum_{i \in S} d_i}{D} \right\rceil \quad (8)$$

That is the value of the amount of cargo below the vehicle based on the capacity of the vehicle D .

3. Partial Comparison Optimization (PCO)

A new metaheuristics algorithm developed to solve the problem of combinatorial optimization is partial comparison optimization (PCO). This algorithm is named like this because the basic step of PCO is finding the optimum point is done partially to complete from a permutation series. PCO is a new metaheuristic method. PCO is used to solve optimization problems by finding the best solution from the alternatives considered to solve the problem. PCO will solve the problem of combinatorial optimization that combines the sequence of elements. The best solutions produced by PCO are optimal solutions such as minimum costs, maximum profits, or the fastest turnaround time. The problem of optimization problems which is the main problem in solving metaheuristics is quite well avoided by PCO.

Regarding its nature as a combinatorial optimization problem solver, PCO has its own advantages in terms of searching time, because PCO does not need to change variables from continue to discrete. One of the problems that can be solved by PCO is scheduling and its variants such as flow-shop, job-shop, and batch-shop scheduling. Another problem related to combinatorial optimization problems that can be solved by PCO is Traveling Salesman Problem (TSP), Vehicle Routing Problem (VRP), and its variants such as the Inventory Routing Problem (IRP). Although PCO is a new method in metaheuristics, PCO in its proof can provide better results than other metaheuristics.

PCO is effective enough to find the optimum value. PCO will not be trapped in a local optimum and more likely to obtain optimal global values. This is because the process of directing the PCO process is in the area of combination values only. With the randomization process, a radical shift from the combination will be possible. This process will provide the possibility of new results that may be more optimal. However, this randomization process is still controlled by guidelines that only choose more optimal results in the combination area. Although for very complex cases the time required for PCO to complete one iteration can be longer than other algorithms, but the time needed to achieve optimal values is relatively not too long because PCO is relatively faster to achieve optimal global. PCO has facultative conditions to get optimal values in faster iterations. However, this option has consequences for a longer processing time.

PCO has a basic guideline which is becoming principle of searching for optimal values, namely:

1. Random Choosing (RC)

A set of elements $J = \{1, 2, 3, \dots, j\}$ is union of elements $I = \{1, 2, 3, \dots, i\}$ and element $K = \{1, 2, 3, \dots, k\}$. A set of elements I contains elements that have not been processed in sequence. A set of elements K contains elements that will be processed in the optimization

process. The RC step is to choose one element I that will be included in the set of elements K . The selection process is carried out one by one until all elements at I enter into element K . The selection of elements is done randomly. This random selection has the purpose of expanding the optimum discovery area. This selection process is controlled and observed by the Partial Comparison (PC) stage. The area of random selection is still within the scope of the permutation set of elements. This random selection will give the possibility of a new series when the partial process is repeated in the next iteration.

2. *Partial Comparison (PC)*

In the PC step, only the elements in K will be processed partially from all elements in J . Partial here is not all elements of J that are processed look for the optimum value, but can also be part of K . Element I_i randomly selected from one set of elements I will find the optimal position in the order of elements K . Element I_i will try to take the position before, between, and after from a series of elements K that has been arranged. Each position will be compared to its fitness value. The position that has the best fitness value will be chosen. This process is carried out as in the NEH algorithm [21].

3. *Changing Neighborhood (CN)*

Based on experiments, for a position taken by a new element I_i , it can change the value of fitness to be better, when the position of neighboring elements exchanges. Element I_i , which takes position n , has a neighbor element K_x at position $n-1$ and element K_y at position $n+1$. The CN process will exchange K_x and K_y positions. This change provides an opportunity to change the level of fitness for the better. If the exchange process provides a better fitness value, this process will be chosen as a better position. The consequences received are CN steps that will extend processing time. But this stage will give you the chance to search for optimal values faster. Based on the experiment, the time generated by the CN process to obtain more optimal results is relatively faster than not using the CN stage. CN can be done more than one position level around the element that is placed I_i . The change process is not only at position $n-1$ and $n+1$, but $n-2$ and $n+2$.

4. *Looping Process (LP)*

Step LP will repeat the search for optimum values iteratively. Obtaining a more optimum fitness value for each iteration is possible because the PCO has an RC step. From the LP step, global optimum will be obtained by comparing the optimum global value with the optimum local. The optimum global value of the order of elements in K to the last iteration is the desired optimal result from PCO calculations.

5. *Stopping Iteration (SI)*

PCO usually gets the optimum solution in initial iterations. At first, PC and CN are processed with probability values 1 to effective iterations ei . ei is determined as an iterative value which usually has reached the optimum value. But this cannot be determined exactly in what iteration. If it is not continued, the optimum value can still be obtained, but if it is continued, it will increase the calculation time in vain. To reduce the processing time but also still want the possibility of getting a better value, PC and CN will be done at a certain position on the new element if the random value is generated below the cp benchmarking probability after iterating to ei . The cp value will be determined between 0 and 1. The cp value will be reduced after ei to the maximum iteration as shown in Figure 1.

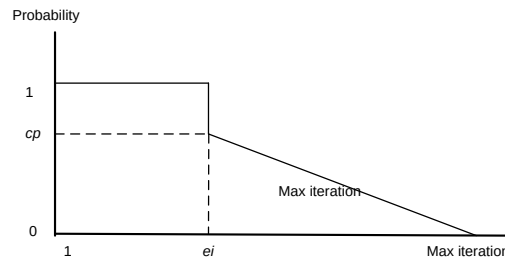


Figure 1 Effective iterations e_i and probability of comparing c_p to determine the processing of PC and CN.

The PCO algorithm is as follows:

- Step 1. Specify $iteration = 1$, $GlobalFitness = \max integer$, elements to be processed $J = \{1, 2, 3, \dots, j\}$, maximum iteration $MaxIter$, effective iteration e_i , and maximum comparable probability cp .
- Step 2. Delete K . Take one element from J and enter K
- Step 3. Take one element from J
- Step 4. If $iteration > e_i$ then $prob = ce - (ce \times ((iteration - e_i) / (MaxIter - e_i)))$ other $prob = 1$
- Step 5. The process of calculating the position of an element if the random number is $rand < prob$. Place and try to position the new element before, between, and after the series of elements that exist in K . Calculate the fitness value of the new sequence from a position occupied by the new element. If the sequence with the new element occupies a position has a better fitness value, set this sequence as a new sequence as the best partial sequence for that element.
For each position occupied by a new element, process the following steps if $rand < prob$. If the position of the new element is between the order of the elements in K , exchange the position of the two elements between the new elements and calculate the fitness of the new sequence with the new element. If the new sequence has a better fitness value, specify a new sequence as the best partial sequence for the position of the new element.
Do Step 5 until all positions in K are occupied by new elements.
- Step 6. Do Step 3 until all the elements in J are placed on K .
 K is the best local sequence for the iteration.
- Step 7. If $fitness(K) < GlobalFitness$ then K becomes the best new global sequence.
- Step 8. Add 1 for iteration. Do Step 2 and repeat the calculation until $iteration = MaxIter$. The best global sequence is a solution that is given a PCO that will give the optimum value.

4. Result and Discussion

To prove that PCO is a good algorithm for completing VRP, a number of examples of symmetric VRP data from vehicle routing problem library (VRPLIB) are used to evaluate PCO performance. 6 examples of normal scale city data are taken from the example VRPLIB as shown in Table 1.

Data from VRPLIB are only coordinate data for each city. Inter-city distance data is not provided. Therefore the optimal distance solution provided by VRPLIB is recalculated based on the optimum route sequence indicated by VRPLIB.

Table 1 The calculation results of VRP by PCO.

No	VRPLIB	Capacity of Truck	Best-known Solution	PCO	SD (%)
1	eil22	6000	375.28	375.28	0.00
2	eil23	4500	568.56	568.55	0.00

No	VRPLIB	Capacity of Truck	Best-known Solution	PCO	SD (%)
3	eil30	4500	497.53	508.13	2.09
4	eil33	8000	826.41	845.81	2.29
5	eil51	160	524.61	572.31	8.33

Table 1 show that the new metaheuristics algorithm PCO has a small standard deviation result referring to best-known solution. It means that PCO provide optimum solution to solve the problem. The route solution of PCO shown as in Figure 2 until Figure 6.

From Figure 2 until Figure 6, PCO will search and set the delivery truck to move to the shortest node near the last node as long as the capacity of the truck still can fulfil the order. This searching method give a chance to provide shortest routes.

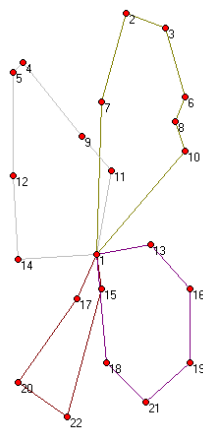


Figure 2 Route of eil22

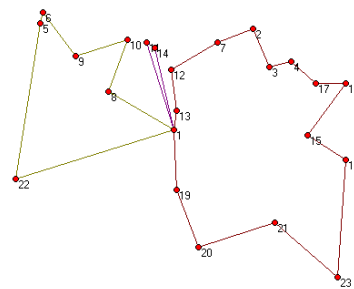


Figure 3 Route of eil23.

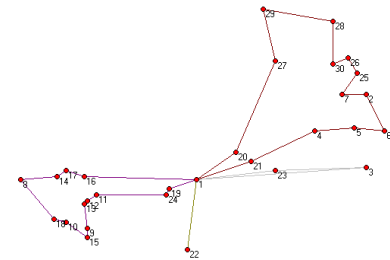


Figure 4 Route of eil30.

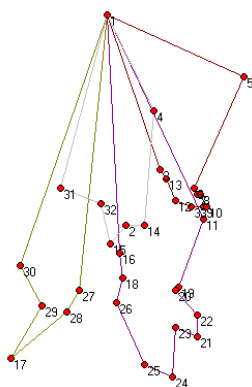


Figure 5 Route of eil33

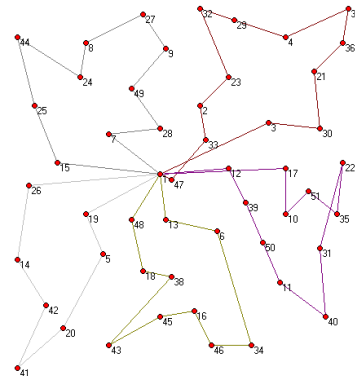


Figure 6 Route of eil51.

5. Conclusion

PCO is a new metaheuristics algorithm that can be used to solve VRP. PCO can provide optimum solution by providing shortest route for the truck in order to deliver product from depot to the nodes. This better solution achieved from PCO to solve VRP will reduce cost of logistic and give better market competition to the company.

In the next research, PCO can be improved to give better performance to solve VRP. Some techniques such as 2-opt or 3-opt can be embedded to improve effectiveness of PCO in order to give

shorter route and other techniques such as tabu search can be embedded to improve efficiency of process in order to reduce time calculation.

6. Reference

- [1] Yu V F, Redi A A N P, Yang C, Ruskartina E and Santosa B. 2017 *Appl Soft Comput* **52**, 657–72
- [2] Farahani R Z, Rezapour S and Kardar L. 2011. *Logist. Oper. Manag.:Concepts and Models* (Elsevier)
- [3] Dantzig G B and Ramser J H. 1959 *Manage. Sci.* **6**, 80–91
- [4] Zhang D, Cai S, Ye F, Si Y and Nguyen T T. 2017 *Inf Sci* **394–395**, 167–82
- [5] Fajarwati I A and Anggraeni W. 2012 *Jurnal Teknik ITS* **1**, 391–6
- [6] Braysy O and Gendreau M. 2001 *Arpakannus* **1**, 33–38
- [7] Lau H C, Sim M and Teo K M. 2003 *Eur. J. Oper. Res.* **148**, 559–69
- [8] Ghani N E A, Shariff S S R and Zahari S M. 2016 *Advances in Pure Mathematics* **6**, 342–50
- [9] Yakici E. 2017 *Comput. Ind. Eng.* **109**, 288–94
- [10] Costa P R O C, Mauceri S, Carroll P and Pallonetto F. 2018 *Electronic Notes in Discrete Mathematics* **64**, 65–74
- [11] Mohammed M A, Ghani M K A, Hamed R I, Mostafa S A, Ahmad M S and Ibrahim D A. 2017 *J. Comput. Sci.* **21**, 255–62
- [12] Baniamerian A, Bashiri M and Tavakkoli-Moghaddam R. 2019 *Appl. Soft Comput.* **75**, 441–60
- [13] Peng Z, Manier H and Manier M A. 2017 *IFAC-PapersOnLine* **50**, 1468–7
- [14] Bula G A, Prodhon C, Gonzalez F A, Afsar H M and Velasco N. 2017 *J. Hazard. Mater.* **324**, 472–80
- [15] Huber S and Geiger M J. 2017 *Eur. J. Oper. Res.* **263**, 419–45
- [16] Polat O. 2017 *Comput. Oper. Res.* **85**, 71–86
- [17] Marinaki M and Marinakis Y. 2016 *Expert Syst Appl* **46**, 145–63
- [18] Euch J. 2017 *Vehicular Communications* **9**, 97–108
- [19] Ng K K H, Lee C K M, Zhang S Z, Wu K and Ho W. 2017 *Comput. Ind. Eng.* **109**, 151–68
- [20] Silvestrin P V and Ritt M. 2017 *Comput. Oper. Res.* **81**, 192–202
- [21] Laporte G. 1992 *Eur. J. Oper. Res.* **59**, 345–58
- [22] Nawaz M, Ensore E E J and Ham I. 1983 *OMEGA Int. J. Manag. Sci.* **11**, 91–5