A New Hybrid-Heuristic Approach for Vertex *p*-Median Location Problems

Hassan Mohamed Rabie[1]
Institute of National Planning
Egypt

Said Salhi[2]
Kent Business School, University of Kent
Canterbury CT2 7FS

***Abstract*—In this paper, a new hybridization of a Myopic and Neighborhood approaches is proposed to solve large-size vertex *p*-median location problems. The effectiveness and efficiency of our approach are demonstrated empirically through an intensive computational experiment on large-size instances taken from TSPLib and BIRCH datasets, with the number of nodes varying from 734 to 9,976 for the former and from 9,600 to 20,000 nodes for the latter. The results show that the new approach, though relatively simple, yields better solutions compared to the ones in the literature. This demonstrates that a simpler approach that takes into account the advantages of other methods can lead to promising outcome and has the potential of being adopted in other combinatorial optimization problems.**

Keywords*—*P-median; discrete location problems; myopic heuristic; neighborhood heuristic

I. Introduction

The *p*-median location problem is one of the oldest discrete location problems. The objective of the vertex *p*-median location problem is to find the location of *p* median facilities among *n* demand points to minimize the sum of the distances between customers and their nearest median facilities. This problem is also known as the minisum vertex location problem [1]. In the uncapacitated type, each median facility is not restricted by the number of demand points/customers to serve, however, in the capacitated *p*-median location problem each median facility has a fixed capacity. In this paper, we are interested in addressing the former for the case of large-scale instances where exact methods may not be suitable. This type of location problems has many applications such as, locating the locations of the ambulances, schools, firefighters and hospitals among others [2-5].

The problem was first introduced in [6, 7], and has been proved to be an NP-hard optimization problem [8]. For large-size location problems, optimal solutions may not be reached, therefore, heuristic and metaheuristic approaches are usually the best way forward for solving these vertex *p*-median location problems [9, 10]. For more information on the vertex *p*-median location problems, see [11-13]. The vertex *p*-median location problem was formulated by ReVelle and Swain in [14] and its implementation was enhanced by Rosing et al. in [15]. The following notation is used.

Let I be the set of nodes (demand points), J the set of potential sites, and C_{ij} the distance between site i ($i \in I$) and demand point j ($j \in J$).

Let p the number of median facilities to be located and let y_i and x_{ij} the following decision variables with:

$$y_i = \begin{cases} 1, if\ a\ facility\ is\ located\ at\ candidate\ site\ i \\ 0, otherwise \end{cases}$$

and x_{ij} the fraction of the demand of customer j that is supplied from facility i.

The p-median location problem can be formulated as follows:

$$minimize\ \sum_{i \in I} \sum_{j \in J} c_{ij} x_{ij} \quad (1)$$

$$subject\ to\ \sum_{i \in I} x_{ij} = 1\ \forall\ j\ \in J \quad (2)$$

$$\sum_{i \in I} y_i = p \quad (3)$$

$$x_{ij} - y_i \le 0\ \forall\ i\ \in I\ ;\ j\ \in J \quad (4)$$

$$y_i\ \in \{0\ ,1\}\ ;\ \forall\ i\ \in I \quad (5)$$

$$x_{ij}\ \ge\ 0\ ;\ j\ \in J \quad (6)$$

Equation (1) is the objective function which minimizes the sum of the total distances, (2) states that all demand at demand site (*j*) must be satisfied, (3) guarantees that exactly *p* median facilities are to be located. Equation (4) ensures that demand nodes can be only assigned to the open median facilities, (5) specifies that the location variables have to be binary, and finally, (6) requires that the assignment variables have to be non-negative.

Though there exist heuristic approaches dedicated for the vertex *p*-median location, there is no recent research that has concentrated on comparing the performance of the heuristic approaches with the state-of-the-art techniques. In this paper, we aim to fill that gap.

A. Contribution and Organization of the Paper

Here, the basic Myopic construction and Neighborhood improvement approaches are first outlined. A new Hybrid-heuristic that integrates the two above techniques is also proposed. In other words, the contribution of this study is three folds:

- Introduce a new Hybrid-heuristic approach which integrates Myopic and Neighborhood for solving large-size vertex *p*-median problems,

- Compare the performance of heuristic approaches versus the performance of recent existing approaches, and
- Generate new best solutions for large instances that can be used for benchmarking purposes or even as bounds for exact methods if need be.

The paper is organized as follows. The next Section presents a brief review of the related literature. The third Section describes some vertex *p*-median heuristic approaches, with an emphasis on the new proposed approach. In Section 4, an illustrative example is presented followed by the computational results in Section 5. The final Section outlines our conclusion and highlights some research avenues.

II. Literature Review

The difficulty of solving large-size vertex *p*-median location problems led several authors to investigate alternative approaches and techniques [16-18]. This section provides a brief overview of some work on recent large-size vertex *p*-median location problems. See [19, 20] for a review on this topic.

In [16], Avella et al. presented a Branch-and-Cut-and-Price approach, which yields reasonable solutions for instances with demand points/customers (*n*) less than 3,795. They applied their approach on the OR library instances, namely, the TSP-Lib instances [21]: Fl1400, PCB3038 and Rl5934, and the Optimal Diversity Management (ODM) instances BN1284, BN3373 and BN5535. In [22], Hansen et al. used a Variable Neighborhood Search (VNS) to find a solution for the clustering problem as a large-size vertex *p*-median location problem. They reported the results for two large-size datasets, where each dataset consists of *p* groups (clusters) of two dimensional data points generated in a square. The datasets consist of BRICH I instances and BRICH III instances [23]. The size of each dataset ranged from 10,000 demand points to 89,600 demand points. The TSP-Lib library is used to choose the second set of instances from [21]: PCB3038, RL5915, RL5934, RL11849, USA13509, IT16862, SW24978, and BM33708, where the number attached to each instance title indicates the number of demand points (*n*) in that instance. In [24], Avella et al. introduced a new Lagrangean relaxation heuristic to solve large-size *p*-median location instances. The algorithm consists of three main components: (1) Subgradient column generation, (2) Core heuristic, and (3) Aggregation procedure. The authors reported the solution of instances from BRICH (two of type BRICH I and two of type BRICH III), and also two instances from the TSP-Lib library [21], namely PCB3038 and USA13509. In [25], Irawan et al. introduced a multiphase approach that includes three parts; (1) demand points aggregation, (2) Variable Neighbourhood Search and (3) an exact approach to solve large-size unconditional and conditional *p*-median location vertex problems. The approach consists of four phases (stages). The first phase solves several aggregated problems using a ‘‘Local Search with Shaking’’ procedure to generate candidate solutions which are then used to solve a reduced location problem in Phase two using Variable Neighbourhood Search or an exact method. The new candidate solution set is then introduced as an input for the iterative learning process to tackle the aggregated *p*-median location problem in Phase three. Finally, Phase four is a post optimization phase applied to solve the original problem using a local search, starting from the best solution obtained in the previous phase. This multiphase approach is tested on three well-known datasets. The first is the BIRCH datasets (BIRCH I and BIRCH III), the second is based on the TSP-Lib library [21] which includes IT16862, SW24978, BU33708 and CH71009 whereas the third is the Circle dataset, which is a newly geometrically generated by the authors to guarantee optimal solutions and hence provides a strong comparison with the heuristic produced. In [26], Irawan and Salhi further designed a hybrid technique based on clustering and VNS with the aim to find a solution to large-size *p*-median location problems. The new approach is a multi-step methodology in which learning from previous steps is taken into account when tackling the next step. Each step consists from sub-problems which are solved by a fast procedure to produce good feasible solutions. Within each step, the solutions are grouped together to produce a new promising subset of potential medians/facilities. This is similar in principle to data mining and heuristic concentration developed by Rosing and ReVelle [27]. The proposed approach is tested on BIRCH datasets. In [3], Janáček and Kvet studied the public service system design which is formulated as a *p*-median location problem, through focusing on the approximate radial approach using dividing points. The approximate approach can be implemented using any commercial integer programming solver. The proposed approach is tested on TSP-Lib instances; RL1304, FL1400, U1432, V1748, D2103, and PCB3038. In [28], Vasilyev and Ushakov proposed a new modified hybrid sequential Lagrangian heuristic that uses a shared memory parallel implementation which can be used in suitable technology. They integrate their Lagrangian relaxation approach with a sub gradient column generation and a core selection method in combination with a simulated annealing to identify the sequences of lower and upper bounds for the optimal value. The proposed approach is also tested on BIRCH datasets. In [29], Vasilyev et al. addressed a general fault-tolerant version of the p-median location problem. The authors adapted their earlier method to determine the upper and lower bounds. They tested the proposed method on large-scale problem instances taken from TSPLIB library: JA9847, USA13509, IT16862, and SW24978.

In summary, due to the importance of the vertex *p*-median location problem and its real-life applications, a considerable amount of *p*-median location approaches have been proposed such as Branch-and-cut-and-price, Variable neighborhood search, Lagrangean relaxation, among others. Though the above approaches are promising, they are not easily and widely applicable. We therefore believe there could still be the need to propose a simple but powerful and effective heuristic-based approach.

III. Heuristics Approaches for the *p*-Median Problem

This section outlined two classes of heuristic approaches: the Myopic approach, and the Neighborhood Search approach. The Myopic approach is a construction method that builds a good solution from scratch, while the neighborhood search approach is an improvement algorithm. More details can be found in [19, 20, 30]. This will then be followed by the new hybrid myopic-neighborhood which we propose.

A. *Myopic Approach*

Myopic is the simplest greedy add (i.e., construction) heuristic approach. The Myopic method starts with an empty set of medians (vertices/points) and successively adds the candidate vertex (point) that yields the best decrease in the minisum objective function value. The process continues until the solution includes p median facilities [19]. The Myopic Approach is simple to understand and to implement. However, it suffers from the fact that once a facility is added, it is not removed in subsequent iterations and therefore will restrict the search space. Fig. 1 shows the commonly used Myopic pseudocode. Let us define the function $Z(J,X) = \sum_{j \in J} min_{m \in X} \{c_{mj}\},$; where X is the current set of candidate solutions. The function depends on both the set of demand nodes to be considered and the candidate locations to be used.

```
Step 1. Let X ← Ø. /Where X is the set of locations, starting with an empty set.
Step 2. Find i* =argmin_{i∈I} {Z(J, X U{i}x)}. / the best node to add to the
        solution set.
Step 3. Set X ← X U { i* }./ Adds that site to the solution.
.       If |X| < P, go to Step 2; else stop.
```

Fig. 1. Myopic Algorithm Pseudocode [20].

Step 1 starts by initializing the set of candidate solutions to an empty set. Step 2 relates to the best vertex (point) to be added to the candidate solution set. Step 3 adds that vertex/point to the candidate solution set. Step 4 checks if less than p median facilities have been added to the solution set. If so, the Myopic approach continues with Step 2; if not, the search terminates. According to Daskin [30], the solution obtained using Myopic method may not be optimal as outlined earlier. This is because once a site is chosen, it remains there which restricts the search making the solution suboptimal as the optimal solution may not necessarily have the additive property. In other words, there is no guarantee of optimality for the Myopic approach, unless we are locating only a single median facility [20].

B. *Neighborhood Approach*

Neighborhood approach attempts to improve a given solution made up of p candidates. It can be considered as one of the most-widely and oldest improvement mechanism [19, 30]. The approach starts with any feasible candidate solution to the vertex p-median location problem (For example, it could begin with the solution set identified by the Myopic approach), then the approach assigns each demand vertex to its nearest median facility. Then the one median location problem within each neighborhood is selected through examining each candidate demand point. If the solution of the one median location results in a new location for the median facility, the approach reallocates all demand points to the nearest open median facility. Otherwise (i.e., if no change for the median facility locations), the approach stops. If there is no new assignments, the approach also stops; otherwise, the search continues [20]. Fig. 2 shows the pseudocode of the Neighborhood search approach. Note that this approach was initially developed for the Weber problem (i.e, the p-median problem but on the plane) by [31] which is known as the locate-allocate method.

```
Step 1.   Input: X. /X is a set of p median facility locations.
Step 2.   Set: N_i ← Ø; ∀i ∈ I ./ N_i is the set of demand nodes for which
          candidate site i is the closest median open facility.
Step 3.   For j ∈ J do
Step 4.      Set i* =argmin_{i∈I} {C_ij}.
Step 5.      Set N_i* ← N_i* U { j }
Step 6.   End for
Step 7.   Set X^new ← Ø / X^new is the set of new facility locations.
Step 8.   For i ∈ I do
Step 9.      If | N_i | > 0 then
Step 10.        Find k* = argmin_{k∈Ni} Z(N_i, {k}).
Step 11.        Set X^new ← X^new U {k*}).
Step 12.     End If
Step 13.  End for
Step 14.  If X ≠ X^new then set X ← X^new and go to Step 2; else stop
```

Fig. 2. Neighborhood Search Approach Pseudocode [20].

Step 1 starts by initializing the solution with any set of p median facilities. Step 2 to Step 6 initialize and set the vertices/points neighborhood. Step 7 initializes a new candidate set of median facility locations. Step 8 to Step 13 the new candidate locations is found. Step 10 calculates the 1-median location problem within each neighborhood and adds that vertex/point to the solution set in Step 11.

C. *The Proposed Hybrid Myopic-Neighborhood Approach*

This section outlines the new Hybrid Myopic-Neighborhood. The main idea of the proposed approach is that, at each iteration of the Myopic and after determining the next point (vertex) to be added to the current solution; the Neighborhood approach is used as many times as possible till there is no improvement in the current candidate solution. In other words, this embedded local search acts as a filtering mechanism during the search process as the open set of facilities is augmented. Fig. 3 shows the pseudocode of the Hybrid Myopic-Neighborhood approach.

```
Step 1.  Let X← Ø. /Where X is the set of locations to be an empty set.
Step 2.  Find i* =argmin_{i∈I} {Z(J, X U{i}}. / the best node/vertex to add to the solution set.
Step 3.  Set X ← X U { i* }./ Adds that site to the solution.
Step 4.  Set: N_i ← Ø; ∀i ∈ I.
Step 5.  For j ∈ J do
Step 6.     Set i* =argmin_{i∈I} {C_ij}.
Step 7.     Set N_i* ← N_i* U { j }
Step 8.  End for
Step 9.  Set X^new ← Ø / X^new is the set of new facility locations.
Step 10.    For i ∈ I do
Step 11.       If | N_i | > 0 then
Step 12.          Find k* = argmin_{k∈Ni} Z(N_i, {k}).
Step 13.          Set X^new ← X^new U {k*}).
Step 14.     End If
Step 15.    End for
Step 15.    If X ≠ X^new then set X ← X^new and go to Step 4; else stop
Step 16.    If |X| < P, go to Step 2; else stop.
```

Fig. 3. Hybrid Myopic-Neighborhood Approach Pseudocode.

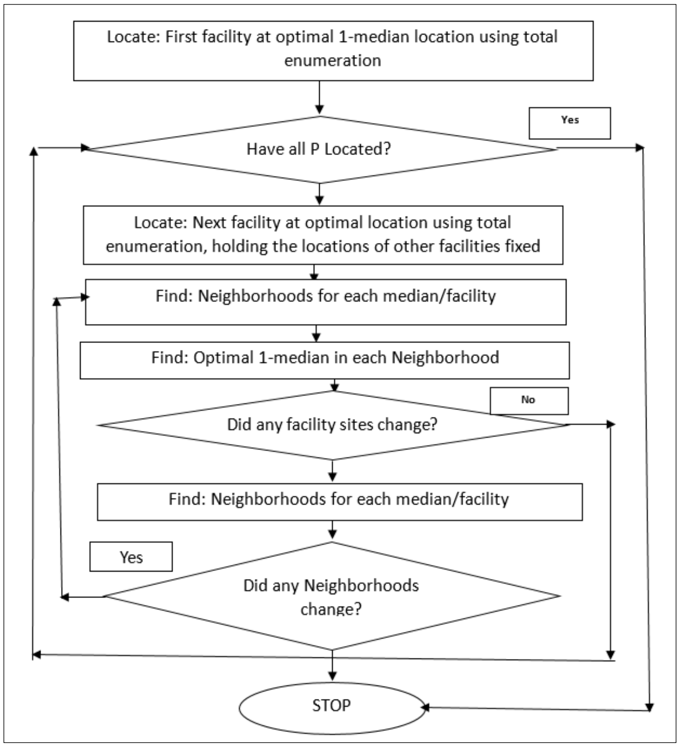

Fig. 4. A Simple Flowchart of the New Hybrid Myopic-Neighborhood Approach.

For completeness, we also provide Fig. 4 as a flowchart for the Hybrid Myopic-Neighborhood approach.

IV. Numerical Example

This section introduces a small example to illustrate the new Hybrid Myopic-Neighborhood approach to solve vertex location problem. For simplicity let us consider an example from Daskin [30] with Table I shows the distance matrix.

For the first median (p=1), using total enumeration, the location of the 1st median is located at vertex (point) 9.

For the second median (p=2), (1) applying Myopic: The location of the 2nd median is located at vertex 7 and the set of solution X= {9, 7}. According to the Myopic, we should proceed by adding a new third vertex which yields the lowest objective function value, however, the new Hybrid approach proposed applying Neighborhood approach; (2) applying Neighborhood, no change has been happened in the solution set, therefore, go to next median (i.e., *median* # 3).

For the third median (p=3), (1) applying Myopic, the location of the 3rd median is located at vertex 6 and the set of solution X={9,7,6}. (2) applying Neighborhood, The new location set, X^{new}={11,7,6}, a change happened at site 9 and swapped with site 11, therefore, go to step 4, and apply again Neighborhood. Now, there is no median facility changed; go to Next median, and so on. The detailed calculations can be found in Appendix A.

It is clear that using the new Hybrid approach improves the objective function value of the Myopic, which considered an added sequence approach. The best solution at p is not necessarily the best solution at p-1 with adding an additional vertex. This is because the additivity property is not satisfied. This observation was also taken advantage of when applying the ‘drop’ method instead of the ‘add’ method, as demonstrated by the flexible drop method, known as subdrop, originally developed by Salhi and Atkinson [32]. Also, it is worth noting that the new Hybrid either improves the objective function value of the Neighborhood or retain that value but never worsen it, since the Hybrid keeps starting from a better intermediate initial solution.

Table II compares the results of using the new Hybrid approach to solve the example in Table I, versus the results of Myopic, Neighborhood, Exchange, and Lagrangian approaches in [30]. The table shows that the new Hybrid Myopic-Neighborhood outperformed other heuristics, namely, Myopic, Neighborhood, and Exchange heuristics approaches. The Hybrid approach achieved optimality at all levels of medians (q=1,…,p), such as the Lagrangian exact approach. However, the new Hybrid approach is very simple to understand and relatively easier to implement in practical setting.

TABLE I. The Distance Matrix for a p-median Example from [30]

	1	2	3	4	5	6	7	8	9	10	11	12
1	0	225	555	825	360	900	270	495	720	600	870	1005
2	150	0	220	400	380	520	330	480	420	550	610	610
3	444	264	0	216	192	360	492	336	240	696	468	468
4	990	720	324	0	612	216	1062	828	432	1116	774	612
5	120	190	80	170	0	180	125	60	120	235	185	215
6	1440	1248	720	288	864	0	1368	1008	288	1200	744	528
7	198	363	451	649	275	627	0	165	495	242	440	671
8	528	768	448	736	192	672	240	0	480	592	400	736
9	624	546	260	312	312	156	585	390	0	494	247	247
10	880	1210	1276	1364	1034	1100	484	814	836	0	418	880
11	1102	1159	741	817	703	589	760	475	361	361	0	399
12	1340	1220	780	680	860	440	1220	920	380	800	420	0

TABLE II. COMPARING THE HYBRID APPROACH VERSUS OTHER HEURISTIC AND EXACT APPROACHES

P	Myopic	Exchange	Neighborhood	Lagrangian	Hybrid
1	**4,772**	**4,772**	**4,772**	**4,772**	**4,772**
2	**3,145**	**3,145**	**3,145**	**3,145**	**3,145**
3	2,641	2,498	2,641	**2,438**	**2,438**
4	2,157	**1,884**	2,157	**1,884**	**1,884**
5	1,707	**1,444**	1,572	**1,444**	**1,444**
6	1,327	**1,083**	1,192	**1,083**	**1,083**
7	966	**747**	**747**	**747**	**747**
8	666	**531**	666	**531**	**531**
9	426	**366**	426	**366**	**366**
10	**210**	**210**	**210**	**210**	**210**
11	**60**	**60**	**60**	**60**	**60**
12	**0**	**0**	**0**	**0**	**0**

V. COMPUTATIONAL RESULTS

To assess the performance of the new Hybrid Myopic-Neighborhood approach to solve large-size vertex *p*-median location problems, an extensive computation experiment using two frequently types of datasets is carried out. These datasets consist of the TSP-Lib, and BIRCH instances. Computational results of the vertex *p*-median instances are listed in Table III, Table IV, Table V and Table VI. This section reports the computational results on a subset of large-size instances which have previously been used in [22, 24, 25, 28]. Computational experiments were carried out on a processor Intel(R) Core(TM) i7-8550U CPU@1.80GHz 1.99 with 8 GB of RAM, under Windows 10, 64-bit. The code was written and executed in MATLAB.

A. TSP-Lib Instances

The first dataset of instances is taken from the TSP-Lib, a travelling salesman library [21]. There are 11 instances (UY734, ZI929, MU1979, CA4663, TZ6117, EG7146, YM7663, EI8246, JA9847, GR9882, and KZ9976), where each instance is solved with *p* varying from 25 to 75 with an increment of 5. The number attached to each instance name indicate the number of demand points (*n*) of that instance. For example, UY734 contains the coordinates of 734 cities in the Uruguay. The instances are ranged in size from *n* = 734 to 9,976.

1) Comparison vs Neighborhood “NBHD” and Myopic: The computational results for the new Hybrid approach on the TSP-Lib dataset are presented in Table III alongside those obtained by the Neighborhood “NBHD” and Myopic approaches, with ‘Bold’ showing the best solutions. For clarity, the items in Table III are as follows:

a) n is the number of demand points,

b) p is the number of median facilities to be located,

c) z is the minisum objective function obtained by the three approaches,

d) Deviation(%): The percentage gap between a given solution and the best solution. It is computed as: *Deviation*(%)= $100*(Z_H-Z_{Best})/Z_{Best}$), where Z_H and Z_{Best} correspond to the *Z* value obtained with heuristic ‘*H*’ and the best *Z* value respectively. ‘Bold’ values in the table refer to the best solutions [25].

e) Time (*Sec*); Time in seconds. We should notice that the time of Neighborhood "NBHD" is significantly small since the approach starts by the candidate solution obtained by the Myopic approach and apply one pass of improvement instead of repeated ones in earlier steps.

Generally speaking, the new Hybrid approach provides better results than Neighborhood and Myopic approach, for all TSP-Lib instances listed in Table III. This means that the new Hybrid approach outperforms both the Neighborhood and Myopic approaches of the vertex *p*-median location problem.

2) Comparison vs. existing techniques: Table IV compared the performance of the Hybrid Myopic-Neighborhood versus the performance of two versions of Variable Neighborhood Search approach called Var1 and Var2 presented in [25]. The results are given in Table IV which shows the value of the objective function (*Z*), the deviations in % and the CPU time in seconds for the Hybrid approach. The notations in the table are the same as the ones given earlier for Table III.

To our surprise, the results demonstrated that the new Hybrid provides better solutions compared to Variable Neighborhood Search on all TSP-Lib instances listed in Table IV. The Hybrid approach yields new benchmarking solutions for all of the instances by producing 11 new best solutions which can be used for further benchmarking.

B. BIRCH Instances

BIRCH is a generated-synthetic dataset suggested by Zhang et al. [23]. Each BIRCH dataset contains *p* two-dimensional clusters demand points (data points) generated in a square. Dataset of type I is the easiest to solve while datasets II and III are harder [22]. Type 1 and Type 3 instances results are reported in this paper; these are the most frequently used types in the vertex *p*-median literature. The largest problem instance generated in this category contains 20,000 demand points. The number of medians (clusters) *p* is ranged from 25 and 100 as shown in Table V which also shows the results of the 24 BIRCH instances.

TABLE III. COMPUTATIONAL RESULTS FOR THE HYBRID, NEIGHBORHOOD "NBHD" AND MYOPIC APPROACHES ON THE TSP-LIB

n	***p***	***Z***			***Deviation (%)***		***Time (Sec)***		
		Hybrid	***NBHD***	***Myopic***	***NBHD***	***Myopic***	***Hybrid***	***NBHD***	***Myopic***
734	25	**205538.00**	213171.90	221036.85	3.71	7.54	1.4	0.02	0.47
929	30	**199171.23**	200646.89	208460.24	0.74	4.66	5.5	0.05	0.72
1979	35	**305849.50**	313868.19	329945.58	2.71	7.88	163.6	0.12	3.46
4663	40	**6508942.87**	6651807.60	6858853.82	2.14	5.38	376.0	1.04	44.62
6117	45	**2293933.42**	2388862.31	2483309.72	4.13	8.26	639.4	1.68	124.25
7146	50	**926303.48**	938372.66	982511.38	1.03	6.07	673.0	2.97	235.86
7663	55	**1337941.96**	1359608.60	1431263.27	1.41	6.97	1021.6	3.60	333.77
8246	60	**1206962.23**	1233756.02	1263392.14	2.29	4.68	1532.0	3.19	474.34
9847	65	**3211308.65**	3271312.72	3386102.36	1.99	5.44	2328.5	6.67	1134.28
9882	70	**1768824.70**	1780385.42	1892195.85	0.77	6.97	1958.3	8.52	1289.16
9976	75	**6172025.46**	6275422.16	6549036.63	1.72	6.11	2250.1	4.09	1161.07
Average					**2.06**	**6.36**			

TABLE IV. COMPUTATIONAL RESULTS FOR THE HYBRID, VAR1 AND VAR2 APPROACHES ON THE TSP- LIB INSTANCES

n	***p***	***Z***				***Deviation (%)***	
		Hybrid	***Var1***	***Var2***		***Var1***	***Var2***
734	25	**205538.00**	209214	207647		1.79	1.03
929	30	**199171.23**	208002	209374		4.43	5.12
1979	35	**305849.50**	320885	320777		4.92	4.88
4663	40	**6508942.87**	6885883	6817921		5.79	4.75
6117	45	**2293933.42**	2412269	2371727		5.16	3.39
7146	50	**926303.48**	1010761	1003380		9.12	8.32
7663	55	**1337941.96**	1430669	1416127		6.93	5.84
8246	60	**1206962.23**	1235810	1241036		2.39	2.82
9847	65	**3211308.65**	3365986	3311024		4.82	3.11
9882	70	**1768824.70**	1869116	1859538		5.67	5.13
9976	75	**6172025.46**	6378764	6340439		3.35	2.73
Average						**4.94**	**4.28**

1) Comparison vs. Neighborhood "NBHD" and Myopic: The computational results for our new Hybrid approach on the BIRCH dataset are presented in Table V, where the summary results of the three approaches (Hybrid, Neighborhood "NBHD" and Myopic) are shown. Here, we have the obtained objective function (*Z*) of the three approaches, the deviation (%) of the NBHD and Myopic form the Hybrid, and the run time (in seconds). On the BIRCH instances of Type 1 and Type 3, the Hybrid approach, as shown in earlier experiments, provides again better solutions compared to Neighborhood "NBHD" and Myopic approaches.

2) Comparison vs. existing techniques: The results of our computational experiments on the BIRCH I and III datasets are also compared with the results obtained by Hansen et al. [22], who presented a primal–dual variable neighborhood search (VNS) algorithm; and Avella et al. [24] who introduced a Lagrangean relaxation approach, which consists of three components: (1) subgradient column generation; (2) core heuristic; and (3) an aggregation procedure. The two approaches are referred to as VNS and CH respectively. The results of the VNS and CH approaches are taken from [24].

The computational results of the new Hybrid approach on the BIRCH dataset are presented in Table VI versus the results of the other two approaches (VNS and CH). For the BIRCH I instances the new Hybrid provides better solutions compared to VNS and CH. The Hybrid approach yielded the same results as the best-known results with deviation equal to (0.000), while VNS and CH yielded (0.039) and (3.674) respectively. On the BIRCH III instances, Hybrid outperforms VNS and CH where Hybrid found three best solutions and yielded the smallest deviation (0.002). As stated in [22] our experiments also show that the BIRCH instances of type 3 are harder to solve compared to type 1 instances. Also, Table VI shows that not only the Hybrid approach yielded better results than VNS and CH, but also requires relatively less computational burden compared to its nearest competitor in terms of quality, namely, the VNS metaheuristic. This reduction in computational time is even more significant in those larger instances where the hybrid consumes only approximately 15% of the time spent by the VNS.

TABLE V. COMPUTATIONAL RESULTS FOR THE HYBRID, NBHD AND MYOPIC APPROACHES ON THE BIRCH INSTANCES

Type 1		**Z**			***Deviation (%)***		***Time (Sec)***		
n	***p***	***Hybrid***	***NBHD***	***Myopic***	***NBHD***	***Myopic***	***Hybrid***	***NBHD***	***Myopic***
10000	100	**12,428.50**	12,428.50	15,587.56	0.00	25.42	3496.5	3.4	1920.1
15000	100	**18,639.30**	19,079.36	22,934.81	2.36	20.21	6515.9	12.9	4137.4
20000	100	**24,840.30**	25,462.10	30,931.91	2.50	21.48	7483.5	13.0	8648.0
9600	64	**11,934.80**	12,407.19	15,287.29	3.96	23.21	751.7	1.9	686.3
12800	64	**15,863.80**	15,863.81	20,005.21	0.00	26.11	1242.3	2.5	1382.2
16000	64	**20,004.60**	21,423.51	25,370.31	7.09	18.42	2135.9	6.8	1821.3
19200	64	**24,018.30**	24,964.97	31,225.47	3.94	25.08	2914.6	3.7	2711.1
10000	25	**12,455.70**	12,455.71	15,548.69	0.00	24.83	116.3	0.4	105.2
12500	25	**15,597.10**	15,597.15	19,734.60	0.00	26.53	1331.4	0.5	308.8
15000	25	**18,949.30**	18,949.25	23,531.83	0.00	24.18	413.1	0.9	261.6
17500	25	**21,937.40**	21,937.40	27,119.47	0.00	23.62	543.7	1.0	409.8
20000	25	**25,096.80**	25,096.82	31,082.81	0.00	23.85	897.9	0.8	558.7
Average					**1.65**	**23.58**			
Type 3		**Z**			***Deviation (%)***		***Time (Sec)***		
n	***p***	***Hybrid***	***NBHD***	***Myopic***	***NBHD***	***Myopic***	***Hybrid***	***NBHD***	***Myopic***
10000	100	**9,624.79**	10,023.04	10,542.17	4.14	5.18	2881.3	7.3	2737.2
15000	100	**15,904.12**	16,461.50	17,297.50	3.50	5.08	6252.1	11.4	5685.3
20000	100	**19,989.02**	20,757.61	21,958.13	3.85	5.78	7372.9	26.0	6783.1
9600	64	**8,225.58**	8,470.84	8,793.05	2.98	3.80	776.4	4.3	766.2
12800	64	**10,210.36**	10,597.02	11,779.22	3.79	11.16	1218.3	6.4	1193.5
16000	64	**13,340.47**	13,805.74	14,653.65	3.49	6.14	2337.1	12.1	1904.1
19200	64	**15,207.56**	15,671.28	16,915.79	3.05	7.94	3367.3	8.0	2606.2
10000	25	**7,203.39**	7,507.42	7,813.95	4.22	4.08	115.2	1.1	104.9
12500	25	**8,576.10**	9,219.43	10,033.18	7.50	8.83	278.5	1.5	171.9
15000	25	**9,513.64**	9,864.70	10,188.10	3.69	3.28	287.1	0.9	252.4
17500	25	**12,535.68**	13,686.14	14,877.32	9.18	8.70	465.5	1.4	347.2
20000	25	**13,052.81**	13,935.27	15,085.42	6.76	8.25	582.2	1.7	491.4
Average					**4.68**	**6.52**			

TABLE VI. COMPUTATIONAL RESULTS FOR THE VNS, CH AND HYBRID APPROACHES ON THE BIRCH INSTANCES

BIRCH instances of Type 1				***Deviation (%)***			***Time (Sec)***		
n	***p***	**Best-Known**		**VNS**	**CH**	**Hybrid**	**VNS**	**CH**	**Hybrid**
10000	100	**12428.5**		0.021	0.001	**0.000**	786	47	3,496.5
15000	100	**18639.3**		0.213	0.002	**0.000**	3,386	101	6,516
20000	100	**24840.3**		0.000	0.001	**0.000**	3,982	210	7,484
9600	64	**11934.8**		0.023	0.002	**0.000**	1,205	56	**752**
12800	64	**15863.8**		0.015	0.001	**0.000**	2,451	84	**1,242**
16000	64	**20004.6**		0.000	0.001	**0.000**	2,739	129	**2,136**
19200	64	**24018.3**		0.021	0.002	**0.000**	3,698	219	**2,915**
10000	25	**12455.7**		0.065	0.001	**0.000**	1,091	82	**116**
12500	25	**15597.1**		0.049	8.794	**0.000**	2,073	115	**1,331**
15000	25	**18949.3**		0.028	16.681	**0.000**	2,353	175	**413**
17500	25	**21937.4**		0.026	8.437	**0.000**	2,615	241	**544**
20000	25	**25096.8**		0.001	10.168	**0.000**	3,055	365	**898**
Average				0.039	3.674	**0.000**	2,453	152	**2,320**
BIRCH instances of Type 3				***Deviation (%)***			***Time (Sec)***		
n	***p***	**Best-Known**		**VNS**	**CH**	**Hybrid**	**VNS**	**CH**	**Hybrid**
10000	100	**9624.79**		0.096	0.002	**0.002**	2609	60	2,737.2
15000	100	**15904.12**		0.094	21.767	**0.005**	3,495	121	5,685
20000	100	**19989.02**		0.181	27.983	**0.003**	3,429	222	6,783
9600	64	**8225.58**		0.123	21.912	**0.002**	1,483	57	**766**
12800	64	**10210.36**		0.117	11.412	**0.001**	2,503	98	**1,194**
16000	64	**13340.47**		1.890	23.142	**0.001**	3,169	170	**1,904**
19200	64	**15207.56**		0.907	38.925	**0.006**	3,243	229	**2,606**
10000	25	**7203.39**		0.834	11.349	**0.000**	1,016	94	**105**
12500	25	**8576.1**		0.788	0.956	**0.000**	1,606	144	**172**
15000	25	**9513.64**		3.099	52.041	**0.003**	2,742	192	**252**
17500	25	**12535.68**		1.141	38.387	**0.000**	2,803	250	**347**
20000	25	**13052.81**		2.060	54.700	**0.003**	3,364	364	**491**
Average				0.944	25.215	**0.002**	2,622	167	**1,920**

VI. Conclusion

This paper introduces a new Hybrid heuristic to solve large scale vertex *p*-median location problems varying in size ranging from 734 to 20,000 demand points. Many Heuristics, Meta-heuristics and Exact approaches have been developed for this purpose. This paper presented a new Hybrid-heuristic approach which integrates two heuristic approaches, namely, the Myopic approach as a construction method to find the solution of each q ($q=1,..,p$) median facility; while the Neighborhood approach improves this solution as much as possible at each level of q. By embedding the Neighborhood approach into the Myopic heuristic within the search and not put in a sequential manner as a post optimizer at the very end, as usually applied, excellent results have been produced which are highly competitive with the state-of-the-art approaches on large-size instances with up to 20,000 demand points. This can be seen to act as a continuous filtering mechanism to guide the search.

The new Hybrid approach was tested on the TSP-Lib instances (n = 734−9976) and outperformed the ones by [25]. In addition, the new approach was assessed on several large-size BIRCH instances (n = 9600−20000), each instance is solved with p ranging from 25 to 100. The results show that our method gives better solutions compared to [24] and [22] results.

In brief, the results show that the new approach gives in general better solutions which can be then used for benchmarking purpose in the future. This demonstrates that a simpler approach that takes into account the advantages of other methods can lead to promising outcome besides having the potential to be adopted in tackling other combinatorial optimization problems.

Acknowledgment

I would like to acknowledge Prophet Muhammad (PBUH) who has taken all human being from the darkness to the lightness.

References

[1] Irawan, S. Salhi, and Z. Drezner, "Hybrid meta-heuristics with VNS and exact methods: application to large unconditional and conditional vertex p-centre problems " Journal of Heuristics, 2015.

[2] H. M. Rabie, I. A. El-Khodary, and A. A. Tharwat, "Particle Swarm Optimization algorithm for the continuous p-median location problems," in 2014 10th International Computer Engineering Conference (ICENCO), 2014, pp. 81-86: IEEE.

[3] J. Janáček and M. Kvet, "Sequential approximate approach to the p-median problem," Computers & Industrial Engineering, vol. 94, pp. 83-92, 2016.

[4] C. Öztürk, G. Tuzkaya, and S. Bulkan, "Centrality based solution approaches for median-type incomplete hub location problems," Computers & Industrial Engineering, vol. 156, p. 107275, 2021.

[5] H. M. Rabie, "Particle swarm optimization and grey wolf optimizer to solve continuous p-median location problems," in Machine Learning and Big Data Analytics Paradigms: Analysis, Applications and Challenges: Springer, 2021, pp. 415-435.

[6] S. L. Hakimi, "Optimum Locations of Switching Centers and the Absolute Centers and Medians of a Graph," Operations Research, vol. 12, no. 3, pp. 450-459, 1964.

[7] S. L. Hakimi, "Optimum Distribution of Switching Centers in a Communication Network and Some Related Graph Theoretic Problems," Operations Research, vol. 13, no. 3, pp. 462-475, 1965.

[8] O. Kariv and S. L. Hakimi, "An algorithmic approach to network location problems. Part 2:The p-Medians," SIAM Journal of Applied Mathematics, vol. 37, no. 3, pp. 539–560, 1979.

[9] C. A. Irawan, S. Salhi, and M. P. Scaparra, "An adaptive multiphase approach for large unconditional and conditional p-median problems," European Journal of Operational Research, vol. 237, pp. 590–605, 2014.

[10] H. M. Rabie, "Particle Swarm Optimization and Grey Wolf Optimizer to Solve Continuous p-Median Location Problems," Cham, 2020, pp. 136-146: Springer International Publishing.

[11] R. Z. Farahani and M. Hekmatfar, Facility Location Concepts, Models, Algorithms and Case Studies. Springer-Verlag Berlin Heidelberg, 2009.

[12] Z. Drezner, J. Brimberg, N. Mladenović, and S. Salhi, "New Heuristic Algorithms for Solving the Planar p-Median Problem," Computers & Operations Research, 2014.

[13] R. Z. Farahani, N. Asgari, N. Heidari, M. Hosseininia, and M. Goh, "Covering problems in facility location: A review," Computers & Industrial Engineering, vol. 62, pp. 368–407, 2012.

[14] C. S. ReVelle and R. W. Swain, "Central Facilities Location," Geographical Analysis, vol. 2, no. 1, pp. 30-42, 1970.

[15] K. Rosing, C. ReVelle, and H. Rosing-Vogelaar, "The p-median model and its linear programming relaxation: an approach to large problems," The Journal of the Operational Research Society, vol. 30, no. 9, pp. 815–823, 1979.

[16] P. Avella, A. Sassano, and I. Vasil'ev, "Computational study of large-scale p-median problems," Math. Program, vol. 109, pp. 89–114, 2007.

[17] H. M. Rabie, I. A. El-Khodary, and A. A. Tharwat, "A particle swarm optimization algorithm for the continuous absolute p-center location problem with Euclidean distance," International Journal of Advanced Computer Science and Applications (IJACSA), vol. 4, no. 12, pp. 101-106, 2013.

[18] S. Salhi, The Palgrave Handbook of Operations Research. Springer Nature, 2022.

[19] G. Laporte, S. Nickel, and F. Saldanha-da-Gama, "Introduction to location science," in Location science: Springer, 2019, pp. 1-21.

[20] G. Laporte, S. Nickel, and F. S. d. Gama, Location Science. Springer 2015.

[21] G. Reinelt, "TSLIB—a traveling salesman library," ORSA J. Comput, vol. 3, pp. 376–384, 1991.

[22] P. Hansen, J. Brimberg, D. Urošević, and N. Mladenović, "Solving large p-median clustering problems by primal–dual variable neighborhood search," Data Mining Knowledge Discovery, vol. 19, no. 3, pp. 351-375, 2009.

[23] T. Zhang, R. Ramakrishnan, and M. Livny, "BIRCH: A new data clustering algorithm and its applications," Data Mining Knowledge Discovery, vol. 1, no. 2, pp. 141-182, 1997.

[24] P. Avella, M. Boccia, S. Salerno, and I. Vasilyev, "An aggregation heuristic for large scale p-median problem," Computers & Operations Research, vol. 39, pp. 1625–1632, 2012.

[25] C. A. Irawan, S. Salhi, and M. P. Scaparra, "An adaptive multiphase approach for large unconditional and conditional p-median problems," European Journal of Operational Research, vol. 237, no. 2, pp. 590-605, 2014.

[26] C. A. Irawan and S. Salhi, "Solving large p-median problems by a multistage hybrid approach using demand points aggregation and variable neighbourhood search," Journal of Global Optimization, vol. 63, no. 3, pp. 537-554, 2015.

[27] K. E. Rosing and C. ReVelle, "Heuristic concentration: Two stage solution construction," European Journal of Operational Research, vol. 97, no. 1, pp. 75-86, 1997.

[28] I. Vasilyev and A. Ushakov, "A shared memory parallel heuristic algorithm for the large-scale p-median problem," in International Conference on Optimization and Decision Science, 2017, pp. 295-302: Springer.

[29] I. Vasilyev, A. V. Ushakov, N. Maltugueva, and A. Sforza, "An effective heuristic for large-scale fault-tolerant k-median problem," Soft Computing, vol. 23, no. 9, pp. 2959-2967, 2019.

[30] M. S. Daskin, Network and Discrete Location: Models, Algorithms, 2nd Edition ed. John Wiley and Sons, Inc., New York., 2015.

[31] L. Cooper, "Heuristic methods for location-allocation problems," SIAM review, vol. 6, no. 1, pp. 37-53, 1964.

[32] S. Salhi and R. Atkinson, "Subdrop: A modified drop heuristic for location problems," Location Science, vol. 3, no. 4, pp. 267-273, 1995.

APPENDIX A

For the first median (p=1) Set $X=\emptyset$, Table 7 shows the calculations of the total enumeration by summing the entries in each column in Table 1, to obtain the values of $Z(J,X)$. Finding i^* =argmin$_{i\epsilon I}$ { $Z(J,X)$}. The smallest value is 4772, i^*=9 and X={9}.

TABLE VII. THE TOTAL ENUMERATION (P=1)

I	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**
Z	**7816**	**7913**	**5855**	**6457**	**5784**	**5760**	**6936**	**5971**	**4772**	**6886**	**5576**	**6371**

For the second median (p=2), (1) applying Myopic approach by finding i^* =argmin$_{i\epsilon I}$ {Z(J, X U{i})}. Table 8 shows the results of this computation, to obtain the values of $Z(J,X)$. The minimum sum value is 3145, corresponds i^*=7 and X={9,7}.

TABLE VIII. THE RESULTS OF THE MYOPIC APPROACH (P=2)

X	9	9	9	9	9	9	**9**	9	9	9	9	9
i	1	2	3	4	5	6	**7**	8	9	10	11	12
Z	3845	3725	3943	4296	3696	4268	**3145**	3655	4772	3563	3858	4392

According to the Myopic, we should proceed by adding a new third point (vertex/median) in the same manner, however, the new Hybrid Myopic-Neighborhood approach proposed applying the Neighborhood approach. By finding N_1 and N_2, where N_i is the set of demand nodes which i median is the closest open median facility to it. Here $N1$ ={3,4,5,6,9,11,12} and $N2$ ={1,2,7,8,10}. Then find k^* =argmin$_{k\epsilon Ni}${$Z(Ni$,{k})}; k^*=9,7 and X^{new}={9,7}. No median facility changed; Go to Next median (p).

For the third median (p=3), (1) applying Myopic, by finding i^* =argminiϵI {$Z(J$, X U{i})}. Table 9 shows the results of this computation to obtain the values of $Z(J,X)$. The minimum value is 2641, corresponds i^*=6 and X={9,7,6}.

TABLE IX. THE RESULTS OF THE MYOPIC APPROACH (P=3)

X	9	9	9	9	9	**9**	9	9	9	9	9	9
	7	7	7	7	7	**7**	7	7	7	7	7	7
i	1	2	3	4	5	**6**	7	8	9	10	11	12
Z	2695	2770	2647	2689	2929	**2641**	3145	2845	3145	2661	2718	2765

(2) applying Neighborhood, by finding N_1, N_2 and N_3. Here N_1 ={3,5,9,11,12} N_2 ={1,2,7,8,10}, and N_3 ={4,6}. Then find k* =argmin$_{k\epsilon Ni}${$Z(N_i$,{k})}; k^*=11,7,6 and X^{new}={11,7,6}, the new value of Z is 2535. Median facility changed; go to step 4.

finding N_1 , N_2 and N_3. N_1 ={8,10,11,12}, N_2 ={1,2,5,7} and N_3 ={3,4,6,9}. Then find k^* =argmin$_{k\epsilon Ni}${$Z(N_i$,{k})}; k^*=11,1,6 and X^{new}={11,1,6} the new value of Z is 2438. Median facility changed; go to step 4.

finding N_1 ,N_2 and N_3. N_1 ={8,10,11,12} N_2 ={1,2,5,7} and N_3 ={3,4,6,9}. Then find k^* =argmin$_{k\epsilon Ni}${$Z(N_i$,{k})}; k^*=11,1,6 and X^{new}={11,1,6} No median facility changed; Go to Next p median and so on.