Intelligent Clustering in Vehicular ad hoc Networks

Farhan Aadil[1, 2], Salabat Khan[2], Khalid Bashir Bajwa[1], Muhammad Fahad Khan[2] and Asad Ali[2]
[1] Department of Computer Engineering, University of Engineering and Technology
Taxila, 45020 – Pakistan
[2] Department of Computer Science, COMSATS Institute of Information Technology,
Attock, 43600 – Pakistan
[e-mail: farhan.aadil@uettaxila.edu.pk (F. Aadil), salabat.khan@ciit-attock.edu.pk (S. Khan),
khalid.bashir@uettaxila.edu.pk (K.B. Bajwa), fahadmuhammad.khan@gmail.com (M.F. Khan)
assaddali000@gmail.com (A. Ali)]
*Corresponding author: Farhan Aadil

Received January 14, 2016; revised May 5, 2016; accepted June 2, 2016; published August 31, 2016

Abstract

A network with high mobility nodes or vehicles is vehicular ad hoc Network (VANET). For improvement in communication efficiency of VANET, many techniques have been proposed; one of these techniques is vehicular node clustering. Cluster nodes (CNs) and Cluster Heads (CHs) are elected or selected in the process of clustering. The longer the lifetime of clusters and the lesser the number of CHs attributes to efficient networking in VANETs. In this paper, a novel Clustering algorithm is proposed based on Ant Colony Optimization (ACO) for VANET named ACONET. This algorithm forms optimized clusters to offer robust communication for VANETs. For optimized clustering, parameters of transmission range, direction, speed of the nodes and load balance factor (LBF) are considered. The ACONET is compared empirically with state of the art methods, including Multi-Objective Particle Swarm Optimization (MOPSO) and Comprehensive Learning Particle Swarm Optimization (CLPSO) based clustering techniques. An extensive set of experiments is performed by varying the grid size of the network, the transmission range of nodes, and total number of nodes in network to evaluate the effectiveness of the algorithms in comparison. The results indicate that the ACONET has significantly outperformed the competitors.

Keywords: Vehicular ad hoc network (VANET), Ant Colony optimization (ACO), Intelligent Transportation Systems (ITS), clustering

A preliminary version of this paper appeared in PLOS ONE, May 05, 2016. This version includes a concrete analysis and supporting implementation results on Load Balance Factors (LBF). The computational complexity of our proposed algorithm is also incorporated in this version. LBF is used as an evaluation criteria to compare the methods. The computational complexity can be calculated for individual steps and then these can be aggregated to represents the overall complexity. The authors are grateful for useful inputs provided by anonymous reviewers.

http://dx.doi.org/10.3837/tiis.2016.08.005 **ISSN : 1976-7277**

1. Introduction

VANET is actually a mobile ad hoc network (MANET), which transforms the automobiles on the roads into network nodes. With the help of VANET, the automobiles on the road are able to create a dispersed network and by doing so vehicles can correspond and exchange information with each other [1]. A wide array of applications can make the most from such technology, e.g. safety, comfort and infotainment related applications [2]. One type of VANET communication is vehicle to vehicle communication (V2V), also referred as ad hoc mode, the second type is vehicle to infrastructure communication (V2I), and the third is hybrid communication. The topology of VANET keeps changing rapidly due the very high mobility pattern of nodes/vehicles. Despite the fact that we can predict the mobility pattern of vehicular nodes in VANET, increasing the lifetime of the network is a relatively difficult task. The topic of scalability, in VANETs, is a vital issue for designers of this network. Among the remedies for the scalability problem, one remedy is clustering, which is important for load balancing and efficient resource utilization. Clustering is a process of grouping the vehicles which are in the same vicinity, helping to make the network more scalable and optimized. [3, 4]. In other words clustering is about segregating the entire network into small logical groups and a technique for increasing the lifetime of network among so many other techniques. Clustering remains an interesting topic of research for optimized throughput of the network in terms of communication. MOBIC [5] is one of the most often mentioned clustering algorithm, which only focuses on MANETs. VANET has been a recently proposed network as compared to MANET and sensor ad hoc networks. Due to this reason, VANET is kind of under explored area of research and needs extensive efforts to enrich the related research work. Some [6-8] research studies are tailored to explain the differences among three types of networks (i.e. VANET, MANET and sensor ad hoc networks) and their relative challenges.

Clustering can be described as a technique of assembling the group of nodes (mobile gadgets, devices, automobiles and many others) inside geographical locality according to certain regulations or protocols. Such regulations vary from one particular algorithm to a new one, and therefore are the crucial aspect to create dependable clusters [9]. Clusters tend to be a kind of virtual sets which are established by using a clustering algorithm. There is one CH, in every cluster, which is nominated or simply elected by many other cluster nodes (CN) of that particular cluster. In most cases, every single CN could be elected as a CH, however, in several algorithms, various kinds of nodes possess more effective properties to become a CH. E.g. a CN with supplemental 3G network connection is often more desirable as CH than its non-3G peers [4, 5, 10]. Cluster size depends on the node's transmission range, and due to this transmission range cluster size varies from cluster to cluster [4, 11, 12].

Vehicular node clustering is a method for improvement in communication efficiency of VANETs. CNs and CHs are elected in the process of clustering. The longer the lifetime of clusters and the lesser the number of CHs, attributes to efficient networking in VANETs. However, clustering of network is an NP hard problem [14] and thus swarm based optimization can be used to find near optimal solutions. This is the basic motivation of proposed work, where clustering is performed using Ant Colony Optimization (ACO).

Cluster stability is a fundamental objective which clustering algorithms endeavor to accomplish and is also regarded as a way of measuring effectiveness of the clustering algorithm. Stability is important for the upper as well as lower communication layers as their performances will raise apparently by using clusters [13]. This simplifies routing, permits spatial reuse of resources and helps makes the network turn up further stable in the view of every CN. Cluster stability can be explained in various ways, yet most often used parameters are i) the number of CH changes and ii) number of a CNs switching their CH. By diligently picking the CH along with CNs that form a specific cluster, their stability can improve considerably [9].

Formation of cluster, maintenance of network topology and distributing resources to all the nodes in the cluster, are all the tasks performed by CH. Due to the dynamic nature of VANET, the topology changes very fast and therefor CH's configuration changes frequently. In this scenario, it is necessary to minimize the number of CHs. The group of nodes which exists within transmission range of CH is called its neighborhood.

According to our best knowledge, the proposed method is a novel method that uses ACO for the very first time for clustering in VANET environment. The method is tailored to handle multi-objective based optimization, moreover, each objective can be assigned with weights as per user requirements. Constraints are added for the construction of a valid solution in ACO. Furthermore, each component of the proposed work is mathematically modeled for a precise description. Comparative analysis is performed based on several evaluation measures to show the significance of the proposed work as compared to recent methods related to optimize clustering.

2. Literature Review

A highest connectivity clustering algorithm is proposed by Gerla and Tsai [10], this algorithm is multi-cluster, multi-hop packet radio network architecture for wireless adaptive mobile information systems. Initially neighbors of a given node are identified in this scheme by calculating the degree of that node. Each node announces its identifier for the election procedure. Once the degree is computed the node with the maximum degree becomes the CH. S.K. Das et al proposed a clustering algorithm, for ad hoc network to optimize the number of clusters, based on genetic algorithm [11]. Weighted clustering algorithm (WCA) where each objective is assigned a weight by the user is presented in 2002 by Chatterjee et al. [7]. It was the first WCA for mobile ad hoc networks, in this algorithm CHs are elected according to their weights. The weights are calculated by combining different parameters. The process of the CH selection process was non-parodic to reduce the communication and computation costs. The call to CH selection procedure was on demand. The diameter of basic network was directly proportional to the time required to identify the CHs. The non-periodic procedure for the selection of CHs is invoked on demand to reduce the computation and communication costs.

For MANETs Shahzad et al. [15] proposed a CLPSO based clustering algorithm. This MANET algorithm efficiently manages the network resources by finding the optimal number of clusters, inter-cluster and/or intra-cluster transmission of data chunks is completed by CHs. CLPSO assigns weight to all network parameters i.e. battery power consumptions, transmission power, node mobility and ideal degree. The information regarding cluster members and CH of each cluster is contained by each particle.

Beneath the umbrella of swarm intelligence there are a couple of key techniques, like ACO and particle swarm optimization. Comparatively few ACO based techniques for multi-objective optimization problems (MOP) are suggested up until now. Furthermore, most of the proposed algorithms are just appropriate towards problems where a lexicographical ordering of the objectives is provided, for instance, where the objectives can belistedt down in accordance with their significance [16, 17]. We talk about swarming behavior regarding ants, social insects, wasps, bees and termites. A lot of individual insects are contained in an insect nest. Itself individual insect is not very smart living being, but the communal entity can make a collective intelligence, i.e., the temperature in the hive is maintained by bees, local stimuli is responded by every reactive agent (Insect) without any reasoning in a simple way. An algorithm inspired by the maneuvering of a bird flock termed as particle swarm optimization (PSO) is proposed in 1995 by Kennedy and Eberhart [18]. Every individual in the flock is guided with the help of best personal and best global behavior in this algorithm. Every individual is converged to the near optimal geographical positions due to these behaviors.In MOPs evolutionary algorithms are already trusted for obtaining multiple solutions. These kind of algorithms are designed for getting several solutions at a time rather than just one solution. Many evolutionary algorithms are developed that work with different mechanisms in order to acquire the solutions, for instance, genetic algorithm [19], differential evolution, artificial immune system and swarm intelligence [15, 18, 20-25], and so forth.

One of the best metaheuristics is ACO which builds the graph of optimization problem, this graph is then explored by artificial ants for the best possible solution of the given problem. [26]. Initially, each ant finds its local solution and then lay pheromone trails over the search space to encourage other ants to further explore the surroundings of the best solutions found. As other evolutionary algorithms are implemented for optimized clustering [8, 11, 15, 27] so this encouraged us to employee ACO based algorithm named ACONET.

3. Optimizing the problem of dominant set problem

Optimization challenges possess excellent significance within scientific engineering model along with decision making applications. Optimization is the term for discovering several remedies of an issue, which will correspond to extreme values connected with more than one objectives. When an optimization problem possesses just one objective, the task of choosing the best possible solution is named single-objective problem. Typically, in a single-objective problem, the focus is on obtaining just a single solution with the exception of multimodal functions. If the optimization problem comes with several objective functions, the optimization problem is referred as MOP. The majority of the real world problems belong to MOPs, as they encompasses a variety of objectives which have to be optimized concurrently. Clustering in VANET also belongs to the set of problems in MOPs [27]. Many conventional mathematical programing approaches produce a single solution for MOPs. For that reason, such approaches may not be appropriate in order to optimize MOPs. The evolutionary algorithms paradigm is rather desirable to fix MOPs as they are population based which enables them to produce a group of solutions in a single iteration. [20].

$$f = W1\big(f_1(d)\big) + W2\big(f_2(d)\big) + W3\big(f_3(d)\big) + \cdots Wn\big(f_n(d)\big) \tag{1}$$

MOPs contain numerous desired goals which are minimized or perhaps maximized at the same time [28]. Such problems possess numerous limitations that a solution must satisfy. The search space is multidimensional in multi-objective optimization. Suppose, there are '*n*' number of objective functions: f_1(d), f_2(d),..., f_n(d), the final evaluation '*f*' of a solution is based on the weighted summation of these objective functions as given in equation (1). Where '*Wi*' represents the weight assigned to i^{th} objective function in the range 0 to 1 and '*d*' represents the decision variables. As an example, decision variables '*d*' for clustering in VANET are: 1) distance of neighboring nodes from CH (the lesser the better), 2) movement speed of CH (the similar the better), and 3) direction of CH and CNs in a cluster (the similar the better). It is possible that more than one optimal solutions are found based on the same values of '*f*'. d^* is called Pareto optimal solution (decision variables) when there is no possible vector of decision variables $d \in D$ which will reduce a few objective value(s) while not creating any increase in any other objective value(s) at the same time (means the final '*f*' value remains the same). This specific strategy in most cases offers not just one solution, but instead a group of solutions known as the Pareto optimal solutions. The vector joining all Pareto optimal solutions is called non-dominated vector, which is when plotted is called a Pareto front. By joining all these non-dominated solutions with a curve, then this curve is called as the Pareto optimal front [21], and all solutions spread on Pareto optimal front are labeled as Pareto optimal solutions. For instance, if there are two objective functions which are contradictory with one another. As multi-objective clustering is the focus of the proposed technique, two objective functions (delta difference and distance of CH from other cluster members) of VANET environment with equal weights are utilized in equation (1) for this purpose.

In a nutshell, objective of the proposed algorithm is to perform clustering such that the number of CHs are minimized and the load distribution in the clusters is balanced (known as load balancing factor, see e.g. Section 4.7). However, there are some constraints e.g.

1) b = The nodes cannot move out from the boundary of grid defined for experiments.
2) t = Node/vehicle transmission range cannot exceed the defined range.
3) n = The number of nodes remain the same as defined at the start of the algorithm.

The objective can be represented mathematically as below:

Min (CHs, LBF)
Subject to: b, t, n

There are two search spaces in MOPs, one is decision variable space and another is objective space. Assortment might be specified within these spaces. Multiple Pareto optimal solutions could be found only if there exists contradictory objectives in '*f*'. There will be just one search space (decision variable space) if the objectives are not contradictory with one another. However, there are two search spaces in MOPs and for this reason the MOPs are considered very challenging.

4. Proposed Technique

ACO meta-heuristic usually models the real world environment of ants in the form of a graph. The vertices of the graph represent the components of a candidate solution. The edges are traversed by ants to create the trails. While traversing different paths, ants put a chemical substance called pheromone to mark the route taken. The artificial pheromone values are associated with the edges and updated based on the quality of a trail. The more the quality of a trail the more the concentration of pheromone is carried out and the more the trail become attractive for the ants. An artificial ant constructs a candidate solution to the problem by adding solution components one by one. Before the construction of complete candidate solution, usually a problem dependent heuristic is used in collaboration with pheromone values to guide the ants moves. Subsequently, as time passed, ants construct their solutions one by one and guide each other to find better and more better solutions. The components with higher pheromone concentrations are thus identified as contributing to a good solution and repeatedly appear in the solutions. Usually, after sufficient iterations, the ants converge towards a good, if not the optimal, solution.

For the application of ACO to a problem we have the following requirements [28]:

- The ability to represent a complete solution as a combination of different components.
- There should be a method to determine the fitness or quality of the solution.
- A heuristic measure for the solution's components (this is desirable, but not necessary).

The pseudo code of ACONET is presented in **Table 1** and the major stages of the proposed algorithm are discussed in follows:

Table 1. Proposed algorithm ACONET

Pseudo code of proposed algorithm ACONET

1: Initialize all vehicles' positions randomly on the highway
2: Initialize the speed/velocity of each vehicle
3: Randomly initialize each vehicle's direction
4: Create a mesh topology among nodes/vertices where each vertex represents the vehicle id
5: Initialize same pheromone values for each edge for the above mesh topology
6: Calculate distance of each vehicle with others, normalize and associate these distance values with the corresponding edges in the above mesh topology
7: **WHILE** (Iteration == Total Iterations OR Stall iteration ==20)
8: { **FOR** Ant_i =1 to Swarm size
9: Ant_i.tour ==empty and cost==infinity
10: Vertices or Nodes – Available for clustering = {All Nodes}
a. **WHILE** (Nodes available for clustering!=empty)
b. {
i. Calculate probability of selection of each node in (Nodes Available for clustering)
ii. CH= Roulette Wheel selection [Probability of all the available for clustering]
iii. Ant.tour.append (CH)
iv. Neighbors of CH = find Neighbors (CH)
v. (Nodes Available for clustering) = (Nodes Available for clustering) –CH
vi. (Nodes Available for clustering)= (Nodes Available for clustering)-Neighbors of CH
c. } **END WHILE**
d. Ant_i.cost=evaluation (Ant_i.tour)
IF (Ant_i.cost < Best Ant.cost)
Best Ant = Ant_i
e. Ant_i++
f. END FOR
11: **FOR** Ant_i =1 to Swarm size
i. Update Pheromone (Ant_i.tour, Ant_i.cost)
a. Evaporate
b. **END FOR**
c. **IF** (BestAnt.cost== Last iteration Best.Ant.cost)
ii. Stall Iteration ++;
d. **ELSE**
iii. Stall Iteration=0;
e. **END IF**
f. Iteration++;
12: END WHILE
13: CHs =Best Ant.tour;

4.1 Search space creation

The solution to a particular problem based on the ACO algorithm starts with designing a problem search space in which the ants conduct the search to find the candidate solutions. The search space of ACONET is a mesh topology based graph as described in Table 1. The label of the vertices in the graph represents the ID's of vehicles/ nodes in the VANET. For example, to perform clustering of a VANET environment with 30 vehicles, the search space will consists of 30 vertices each connected via mesh topology. The edges between the vertices are associated with two values: 1) pheromone value, and 2) heuristic value. In the subsequent subsections, more detail about these two values is provided for concrete understating.

4.2 Pheromone initialization

The edges in the search graph are initialized with the small pheromone values. The initial pheromone τ_{ij} over the edge between two vertices i and j is laid down based on the following equation:

$$\tau_{ij}\,(iter = 1) = \frac{1}{|Vehicles|} \tag{2}$$

Where |Vehicles| represents the total number of the vehicles in the network.

4.3 Solution creation

In each iteration of the FOR loop (line # 8) of the algorithm in **Table 1**, each ant constructs its solution. An ant starts its tour by selecting a vertex in the search space, randomly. Later, the ant selects and adds more vertices in its tour keeping in view pheromone values and heuristic values over the edges subject to some constraints. It may please be noted that the vertices in the tour of an ant are the CHs for clustering. So, each ant tour is a collection of CHs for the given VANET environment. The constraints for selection of a vertex to be added in the tour of an ant are given as:

i. A vertex can only be added in the tour if it is not already present in tour. This constraint makes it sure that a vehicle cannot be selected as CH more than once in a tour/ solution. The tour consists of unique labeled vertices and which represents the CH vehicles in the VANET.

ii. A vertex cannot be added in the tour if it is in the transmission range of a vertex already present in the tour. Once a CH is selected, all the vehicles in the transmission range of the CH become a member of the cluster. This constraint makes it sure that a cluster is controlled by only one CH.

In the proposed algorithm, the probability of next vertex selection (from search space) to be added into the tour of current ant is calculated using equation (3).

$$P_{i,j} = \frac{Pheromone_{i,j} \times Heuristic_{i,j}}{\sum_{k\in S} Pheromone_{i,k} \times Heuristic_{i,k}} \tag{3}$$

Where i is the label of vertex last added into the tour of the current ant, j is the label of next candidate vertex which can be selected by the ant, $P_{i,j}$ is the selection probability of edge between vertex i and j. S is the set of all vertices available for selection subject to above two constraints. $Pheromone_{i,j}$ and $Heuristic_{i,j}$ is pheromone and heuristic values associated with edge between vertex *i* and *j*, respectively. The selection probability of an edge is divided by the summation of the selection probabilities of all the edges available for traversal. The higher the values of pheromone and heuristic over an edge, the better is the chances of its selection. In order to make it sure that the algorithm doesn't stuck in local optima, the selection of an edge is performed by roulette wheel selection [29]. In other words, the edge with lowest selection probability still has the chance of selection and the selection of edge is not based on greed. Once an edge is selected, the correct ant moves over the edge and reaches to a new vertex in the search space. So, the selection of an edge is actually the selection of next vertex to be added to the tour of current ant.

The tour of an ant is completed when there is no more vertex available to be added in the tour due to above mentioned constraints. It is important to note that the length of tours of ants is variable. A tour with less number of CHs or clusters is usually more preferred due to lowest communication overhead as compared to flooding.

4.4 Evaluation of solution and heuristic value calculation

The tour/ solution of an ant is evaluated to determine its worth. Due to multi-objectives nature of VANET clustering, following modified version of Equation (1) is used to evaluate the tour of ant 't':

$$f_t = W1(f_1) + W2(f_2) \tag{4}$$

Where W1=W2=0.5 represents the equivalent weights assigned to two objective functions f_1 and f_2, respectively. For ACONET, f_1 is the delta difference value of the clusters in *t* and f_2 is the summation of distance values of all the CHs from their cluster members. Delta difference value '*d*' of the clusters in a tour can be calculated by employing equation (5):

$$d = \sum_{i=1}^{|t|} ABS(D - |CN_i|) \tag{5}$$

Where 'D' is a constant value and represents the ideal degree of clusters. The value of 'D' depends on user choice. For example, if user needs highly densed clusters, the 'D' may be assigned with a high value and vice versa. $|t|$ is the length of tour or in other words, total number of clusters formed. $|CN_i|$ is the total number of vehicles in cluster '*i*' excluding CH. ABS function returns the absolute value of the given value. The lowest value of '*d*' represents the formation of clusters nearly equivalents to specified ideal degree by user. If value of '*d*' is zero, the clustering is optimal in terms of ideal degree requirements of user.

The value for objective function f_2, can be calculated based on Euclidean distance (ED) between the cluster members and CHs for all the clusters. Distance between CH and all of its member nodes can be calculated using equation (6):

$$dist_{CH\_i} = \sum_{j=1}^{|CN_i|} \mathrm{ED}(\mathrm{CH_i}, CN_{j,i}) \tag{6}$$

Here $\mathrm{CH_i}$ represents the coordinate position of 'i^{th}' CH. CN{j, i} is the coordinate position of "j^{th}" CN and which is the member of cluster '*i*'. Similarly, f_2 objective value is calculated using equation (7):

$$f2 = \sum_{i=1}^{|t|} dist_{CH\_i} \tag{7}$$

Again, $|t|$ is the length of tour or in other words, total number of clusters. Similar to f_1, the lowest value of f_2 is more preferred. The lowest the distance between CH and its cluster members, the lowest value of energy will be required to transfer the data.

Having discussed solution/ tour construction, yet the discussion about heuristic value calculation over an edge is required. Suppose, the ant is over vertex *i* and it has to calculate the heuristic value over the edge between vertex *i* and *j*, Equation (4) can be used for this purpose. Equation (4) is used for evaluating the tour when it is completed, whereas the same equation is used for heuristic calculation when the tour is incomplete (i.e. still the vertices are available that can be added in tour). For incomplete tours, every single available vertex is added in the tour, one by one at a time, and its worth is calculated using Equation (4). In this way, the available vertices are assigned with heuristic values in accordance with their worth determined by Equation (4).

4.5 Update Pheromone in search space

Pheromone values on the edges are an important learning dynamic for the ACONET. The quality of the ant tours/ trails is used to make an efficient use of the pheromone values. The pheromone values on the edges constituting the trails are updated in proportion to the quality of the trails and thus define the learning directions for the subsequent transitions of the entire swarm. Equation (8) is used to update the pheromone values over the edges between the vertices in the trails constructed by ants.

$$\tau_{ik}\,(t+1) = (1-\rho)\tau_{ik}\,(t) + (1 - \frac{1}{1+f_n}\tau_{ik}\,(t)) \tag{8}$$

Where $\tau_{ik}(t)$ is the pheromone value encountered in iteration '*t*' (of the outer most WHILE loop, line # 7, **Table 1**) between $vertex_i$ and $vertex_k$. The pheromone evaporation rate is represented by ρ and f_t is the worth of the tour of n^{th} ant.

The equation (8) updates pheromones by first evaporating a percentage of the previously seen pheromone and then adding a percentage of the pheromone dependent on the quality/ worth of the trail constructed by n^{th} ant. Update of pheromone is carried out for all tours constructed by all the ants. If the tour is well representative to the clustering requirement (based on Equation (4)), the pheromone added in quantity is greater than the pheromone evaporated and the vertices found in the tour become more attractive for the ants in the subsequent iterations. The evaporation in the equation improves exploration, otherwise in the presence of a static heuristic function the ants tend to converge quickly to the terms selected by the entire swarm during the first few iterations of the first inner repeat loop [30].

4.6 Stopping criterions

In this section, different criterions to stop the execution of ACONET algorithm are discussed. The first criteria to stop the execution of ACONET is when the total number of iterations specified by the user are completed (line # 7, **Table 1**). The second criteria to stop the execution is when the count of stall iteration reaches to 20 (initially started from 0). An iteration is considered stall if there is no improvement in the quality of best trail found in outermost WHILE loop as compared to the quality of best trail found in previous iteration of outermost WHILE loop. Finally, after stopping the execution of ACONET, the best tour found so far is used for clustering of the VANET.

4.7 Computational Complexity of ACONET:

Following symbols are used in calculations:

n= total number of vehicles/nodes
r=total number of iterations executed
z=number of ants
k = Average number of CHs in a solution constructed by ant

The computational complexity of ACONET can be calculated for individual steps and then these can be aggregated to represent the overall complexity

Solution construction by a single ant:
To decide about a CH to be added into a solution, in the worst case, O (n) time is required for ACONET. It may please be noted that for this decision, probability calculation is performed over pre-computed values of heuristic and pheromone. For a solution, the above decision is done 'k' times. So, the solution construction takes **O(k.n)**.

Solution Quality / Fitness:
For a solution with 'k' cluster heads, it takes **O(k.n)** time to calculate the fitness of the solution.

Pheromone update:
ACONET takes O(k) time to increase the amount of pheromone on the links between the 'k' cluster heads related to the solution. It takes O(n) time to decrease the amount of pheromone on unused cluster heads. Since k <= n with tendency to less, this adds upto **O(n)** for ACONET. ACONET requires **O(n)**$^{2-}$operations to increase the pheromone and perform evaporation.

Complexity of while loop (i.e. batch of ants):
ACONET takes O(k.n) + O(k.n) + O(n) for single ant which collapses to: O(k.n) and for 'z' ants, it becomes O(z .(k.n))

For 'r' rules creations in WHILE loop:
So the overall complexity of ACONET is O(r.(z .(k.n)) + (n^2)), where n^2 represents pheromone evaporate operation.5. Implementation, results and discussions

5. Implementation, results and discussions

Experimental setup is described in this section along with the all the result comparisons of our performed experiments. The results of our proposed algorithm ACONET were compared with other popular clustering algorithms i.e. MOPSO [8], and CLPSO based clustering [15]. The experimental results demonstrate the fact that the proposed technique addresses the entire network with the bare minimum number of clusters which can reduce the routing cost of the network. It will allow to decrease the number of hops and packet delays in the cluster-based routing. Typically, there will be more clusters when the transmission ranges of nodes are small. The final results indicate the fact that the proposed clustering technique is effective and adaptable in comparison to other techniques and works much better than the other algorithms in a VANET environment. The algorithm can optimize the parameters associated with the vehicular nodes for seeking the optimal solution. The parameters used in the simulations are expressed in **Table 2** .

Table 2. Simulation Parameters for algorithms

Parameters	**Values for MOPSO and CLPSO**	**Values for ACONET**
Population Size (Particles)	100	100
Maximum iterations	150	150
Evaporation Rate	--	0.05
Inertia weight w	0.694	--
C1[1]	2	2
C2[1]	2	2
Vehicle's velocity range	22 m/s - 30 m/s	22 m/s - 30 m/s
Simulation area	1x1 Km^2, 2x2 Km^2, 3x3 $Km^{2,}$ 4x4 Km^2	1x1 Km^2, 2x2 Km^2, 3x3 $Km^{2,}$ 4x4 Km^2
Maximum acceleration m/s^2	1.5	1.5
Minimum distance B/W Vehicles	2m	2m
Maximum distance B/W Vehicles	5m	5m
Lane width	50m	50m
Total lanes	8	8
Transmission Range	100 m – 600m	100 m – 600m
Mobility model	Freeway mobility model	Freeway mobility model
Simulation Runs	10	10
W1 (wieght of first objective function)	0.5	0.5
W2 (wieght of second objective function)	0.5	0.5

[1]Learning Factor

5.1 Experimental Setup

MATLAB Version 8.5.0 is used for implementation purposes. The experiments are conducted on a machine with 8 GB of RAM and 2.5 GHz core i5 processor. The experiments are performed by varying number of nodes between 10 and 60. There were four possible sizes of road segments for performing these experiments 1km×1km grid, 2km×2km grid, 3km×3km grid and 4km×4km grid. The movement of all nodes were in two directions along with the X-axis with velocity varying uniformly between 80 km/hour (22m/sec) and 120 km/hour (30 m/sec). For each node the transmission range was also varied from 100m to 600m. For load balancing in the ad hoc network the value of degree difference is set to 10. Ten simulations are performed for each algorithm and their average is taken which is presented in results/graphs.

5.2 Transmission Range vs Number of Clusters vs Network Nodes

The transmission range of each node is varied from 100m to 600m and number of nodes were made static against the transmission range to find the number of clusters. Number of Nodes were varied to 30, 40, 50, and 60 as a result four diverse solutions were produced. Results were generated by varying the size of road segment (grid size) to 1km x 1km, 2km x 2km, 3km x 3km and 4km x 4km. The proposed algorithm finds the optimized solutions against each transmission range which is exhibited in **Fig. 1**, these solutions covers the entire network in comparison with CLPSO and MOPSO. Average number of clusters were used as performance metric, shown in **Fig. 1**. In the same scenario, i.e. 1km x 1km, our proposed algorithm produces for each transmission range to cover the whole network as compare to the other algorithms CLPSO and MOPSO. The number of clusters produced by ACONET are less than CLPSO and MOPSO in most cases, moreover, we varied the number of nodes from 10 to 60 to conduct these experiments. Although MOPSO produce multiple solutions, which makes the user more powerful by empowering him to choose among the solutions according to the current scenario of the network, but the number of clusters generated by ACONET are more optimized than MOPSO.

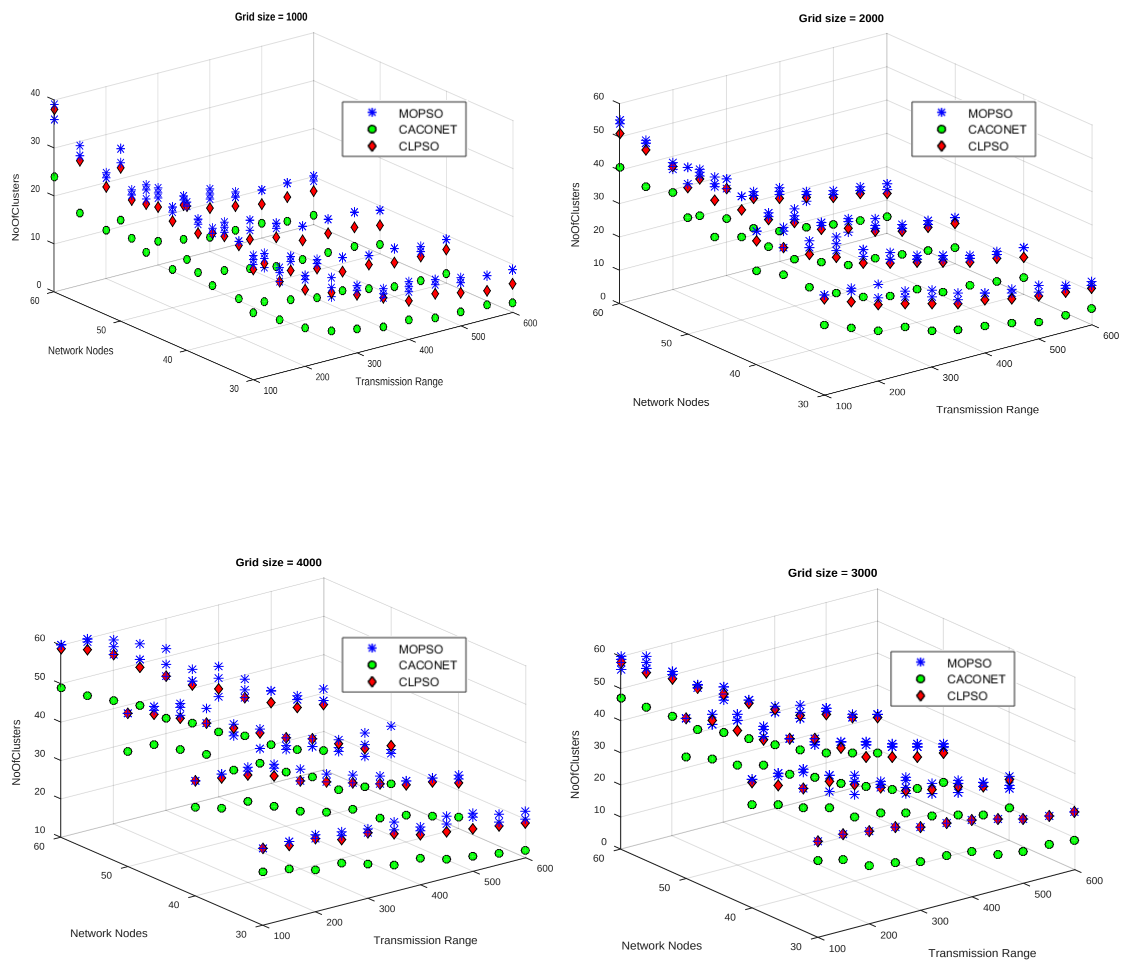


Fig. 1. Transmission range vs. number of clusters vs number of nodes in MOPSO, CLPSO and ACONET by fixing nodes from 30 to 60. And varying the grid size to 1km, 2km, 3km and 4km

After these initial experiments we changed the size of road segment to 2km x 2km, the results of this setup are displayed in **Fig. 1 (b)**. We can observe in results that with less transmission range there are more number of clusters because due to small transmission range nodes are inaccessible from each other, so there are less number of nodes in each cluster. Whereas if the transmission range of nodes rises the number of cluster in each solution decreases, moreover in case of ACONET there are more optimized solution as compared to CLPSO and MOPSO. ACONET also out performs CLPSO in all experiments with improved solutions.

At this point we change the grid size to 3km x 3km as shown in **Fig. 1(c)** Now we change the grid size to 4km x 4km. In **Fig. 1 (d)** MOPSO show the same clusters as number of nodes due to small range of transmission and its decreases gradually downward up to 29 as we increase the transmission range. In CLPSO we also have almost the same trend as with MOPSO. In ACONET graph shows 49 clusters initially, which lead downwards up to 15 at the end when we increase the transmission range, it is because the network area is very large and the transmission range of nodes is comparatively small. So we can say that there is a direct relation between node transmission range and road segment size. It can also depicted that the number of solutions increase as the transmission range increases in case of MOPSO.

5.3 Number of Clusters vs Grid Sizes vs Transmission Ranges

In **Fig. 2** we demonstrate the relationship between different grid sizes, number of clusters and transmission ranges. We kept the number of nodes fixed to 40 and vary the transmissionrangee from 300metersr to 600metersr. **Fig. 2** exhibit that the grid size is inversely proportional to the number of clusters, this is evident that in large grid size the nodes will more scattered which will cause more number of clusters required to cover the entire network and vice versa. By comparing these results we conclude this section, ACONET provides less number of clusters as compare to other algorithms which leads to efficient clustering, moreover we can determine that ACONET performs better in case of dense environment.

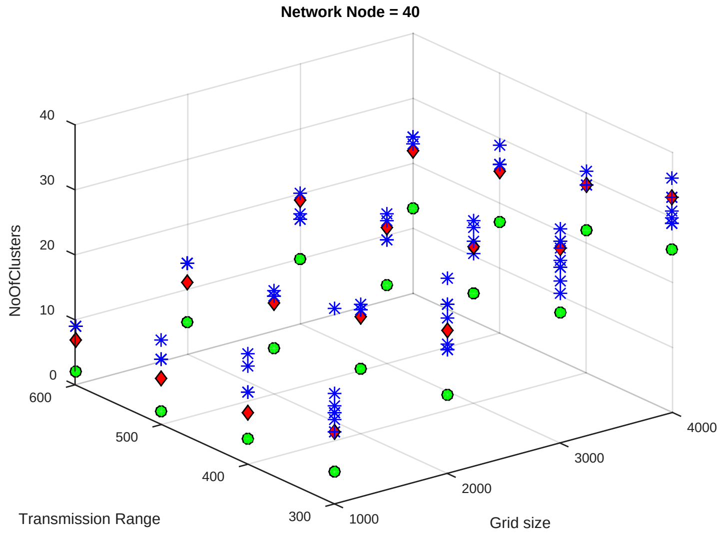


Fig. 2. Number of clusters vs. Grid size vs. Transmission range in case of CLPSO, MOPSO and ACONET when node = 40 and transmission range varying from 300 to 600.

5.4 Load Balance Factor (LBF)

To quantify the load on each CH, load balance factor is used as an evaluation criteria to compare the methods. In an ideal case, every CH must handle an equal number of CNs, but it is very difficult to maintain a perfectly load-balanced system at all times. The main reason is the frequent detachment and attachment of neighbors from the CHs. The cardinality of the cluster size represents the load of a CH. In [8], the LBF is defined as,

$$Load\ Balance\ Factor = \frac{1}{n_c \times \sum_i (x_i - \mu)^2} \quad (9)$$

where n_c is the number of CHs, x_i is the cardinality of cluster i, and _ $\mu = N - n_c/n_c$ is the average number of neighbours of a CH (being the total number of nodes in the system).

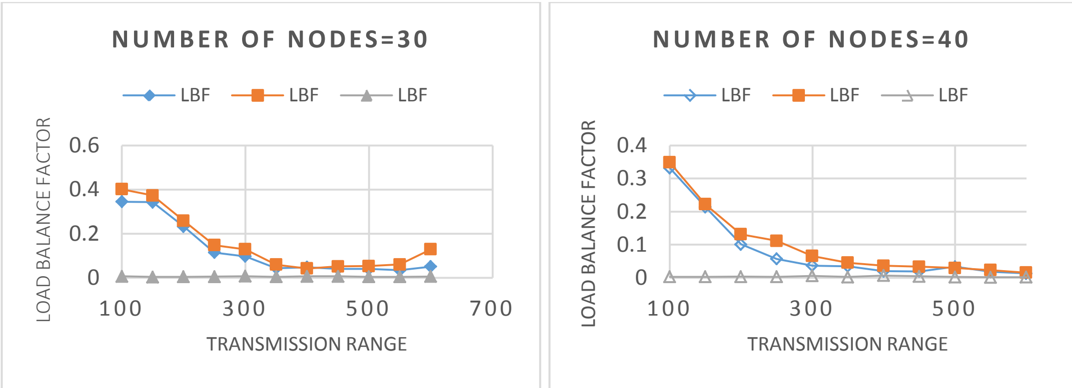


Fig. 3. Load Balance Factor in case of CLPSO, MOPSO and ACONET when grid size is 1km×1km and transmission range varying from 100 to 600 and number of nodes are 30–40.

Fig. 3 shows the load balance factor in case of CLPSO, MOPSO and ACONET. The LBF is calculated by varying the transmission range from 100m to 600m while the grid size is 1km×1km and the number of nodes are 30 and 40. The ACONET gives more balanced clusters than the CLPSO and MOPSO as we increase the transmission range as well as it gives a variety of solutions.

Both graphs in **Fig. 3** exhibits that ACONET is more effective as the number of neighbors reaches the threshold value and performs better than CLPSO and MOPSO in terms of balancing the load in the network.

6. Conclusion

A detailed analysis of multi-objective evolutionary algorithms in VANETs is presented in this paper. In the proposed scheme the node clusters are prepared efficiently, moreover near optimal solutions are generated by the proposed algorithm which are best among the three algorithms employed for VANET clustering in these experiments. The packet routing cost is minimized by minimizing the total number of clusters in the entire network. Due to the evolutionary capability of these algorithms larger search space can be searched as well as objective function values can be adjusted dynamically. The flexibility and effectiveness of the approach are exhibited with the help of simulation results. Result comparisons with other famous algorithms (MOPSO and CLPSO) are also presented in this paper. Optimal number of clusters are found with the help of proposed scheme ACONET. Researchers can enhance list of objectives and make number of nodes, dynamic in future to extend this work, other evolutionary algorithms can also be implemented, e.g. Gray Wolf Optimizer, moth flame optimizer etc. for further extensive comparison.

References

[1] M. Fathian, G. R. Shiran, and A. R. Jafarian-Moghaddam, "Two New Clustering Algorithms for Vehicular Ad-Hoc Network Based on Ant Colony System," *Wireless Personal Communications*, pp. 1-19. Article (CrossRef Link).

[2] K. Jothi and A. E. Jeyakumar, "Optimization and Quality-of-Service Protocols in VANETs: A Review," *Artificial Intelligence and Evolutionary Algorithms in Engineering Systems*, ed: Springer, pp. 275-284, 2015. Article (CrossRef Link).

[3] A. Daeinabi, A. G. Pour Rahbar, and A. Khademzadeh, "VWCA: An efficient clustering algorithm in vehicular ad hoc networks," *Journal of Network and Computer Applications*, vol. 34, pp. 207-222, 2011. Article (CrossRef Link).

[4] N. Kumar, N. Chilamkurti, and J. H. Park, "ALCA: agent learning-based clustering algorithm in vehicular ad hoc networks," *Personal and Ubiquitous Computing*, vol. 17, pp. 1683-1692, Dec 2013. Article (CrossRef Link).

[5] P. Basu, N. Khan, and T. D. Little, "A mobility based metric for clustering in mobile ad hoc networks," in *Proc. of Distributed computing systems workshop, 2001 international conference on*, pp. 413-418, 2011. Article (CrossRef Link).

[6] Y.-C. Tseng, S.-Y. Ni, Y.-S. Chen, and J.-P. Sheu, "The broadcast storm problem in a mobile ad hoc network," *Wireless networks*, vol. 8, pp. 153-167, 2002. Article (CrossRef Link).

[7] M. Chatterjee, S. K. Das, and D. Turgut, "WCA: A weighted clustering algorithm for mobile ad hoc networks," *Cluster Computing*, vol. 5, pp. 193-204, 2002. Article (CrossRef Link).

[8] H. Ali, W. Shahzad, and F. A. Khan, "Energy-efficient clustering in mobile ad-hoc networks using multi-objective particle swarm optimization," *Applied Soft Computing*, vol. 12, pp. 1913-1928, Jul 2012. Article (CrossRef Link).

[9] Z. Y. Rawashdeh and S. M. Mahmud, "A novel algorithm to form stable clusters in vehicular ad hoc networks on highways," *EURASIP Journal on Wireless Communications and Networking*, vol. 2012, pp. 1-13, 2012. Article (CrossRef Link).

[10] M. Gerla and J. T.-C. Tsai, "Multicluster, mobile, multimedia radio network," *Wireless networks*, vol. 1, pp. 255-265, 1995. Article (CrossRef Link).

[11] D. Turgut, S. K. Das, R. Elmasri, and B. Turgut, "Optimizing clustering algorithm in mobile ad hoc networks using genetic algorithmic approach," in *Proc. of Global Telecommunications Conference, 2002. GLOBECOM'02. IEEE*, pp. 62-66, 2002. Article (CrossRef Link).

[12] I. I. Er and W. K. Seah, "Mobility-based d-hop clustering algorithm for mobile ad hoc networks," in *Proc. of Wireless Communications and Networking Conference, 2004. WCNC. 2004 IEEE*, pp. 2359-2364, 2004. Article (CrossRef Link).

[13] A. A. Abbasi and M. Younis, "A survey on clustering algorithms for wireless sensor networks," *Computer communications*, vol. 30, pp. 2826-2841, 2007. Article (CrossRef Link).

[14] D. Aloise, A. Deshpande, P. Hansen, and P. Popat, "NP-hardness of Euclidean sum-of-squares clustering," *Machine learning*, vol. 75, pp. 245-248, 2009. Article (CrossRef Link).

[15] W. Shahzad, F. A. Khan, and A. B. Siddiqui, "Clustering in mobile ad hoc networks using comprehensive learning particle swarm optimization (CLPSO)," in *Communication and Networking*, ed: Springer, pp. 342-349, 2009. Article (CrossRef Link).

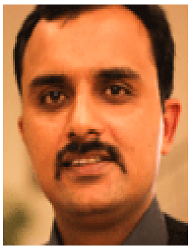
[16] L. M. Gambardella, É. Taillard, and G. Agazzi, "Macs-vrptw: A multiple colony system for vehicle routing problems with time windows," *New ideas in optimization*, 1999. Article (CrossRef Link).

[17] M. Gravel, W. L. Price, and C. Gagné, "Scheduling continuous casting of aluminum using a

multiple objective ant colony optimization metaheuristic," *European Journal of Operational Research*, vol. 143, pp. 218-229, 2002. Article (CrossRef Link).
[18] J. Kennedy, "The particle swarm: social adaptation of knowledge," in *Proc. of Evolutionary Computation, 1997., IEEE International Conference on*, pp. 303-308, 1997. Article (CrossRef Link).
[19] R. Dewri, N. Poolsappasit, I. Ray, and D. Whitley, "Optimal security hardening using multi-objective optimization on attack tree models of networks," in *Proc. of the 14th ACM conference on Computer and communications security*, 2007, pp. 204-213. Article (CrossRef Link).
[20] J. J. Liang, A. K. Qin, P. N. Suganthan, and S. Baskar, "Comprehensive learning particle swarm optimizer for global optimization of multimodal functions," *Evolutionary Computation, IEEE Transactions on*, vol. 10, pp. 281-295, 2006. Article (CrossRef Link).
[21] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *Evolutionary Computation, IEEE Transactions on*, vol. 6, pp. 182-197, 2002. Article (CrossRef Link).
[22] Y. Del Valle, G. K. Venayagamoorthy, S. Mohagheghi, J.-C. Hernandez, and R. G. Harley, "Particle swarm optimization: basic concepts, variants and applications in power systems," *Evolutionary Computation, IEEE Transactions on*, vol. 12, pp. 171-195, 2008. Article (CrossRef Link).
[23] J. Kennedy, "The behavior of particles," *Evolutionary programming VII*, 1998, pp. 579-589. Article (CrossRef Link).
[24] W.-F. Leong and G. G. Yen, "Dynamic population size in PSO-based multiobjective optimization," *Evolutionary Computation, 2006. CEC 2006. IEEE Congress on*, pp. 1718-1725, 2006. Article (CrossRef Link).
[25] S. Mostaghim and J. Teich, "Covering pareto-optimal fronts by subswarms in multi-objective particle swarm optimization," *Evolutionary Computation, 2004. CEC2004. Congress on*, pp. 1404-1411, 2004. Article (CrossRef Link).
[26] M. Dorigo, M. Birattari, and T. Stützle, "Ant colony optimization," *Computational Intelligence Magazine, IEEE*, vol. 1, pp. 28-39, 2006. Article (CrossRef Link).
[27] M. Hadded, R. Zagrouba, A. Laouiti, P. Muhlethaler, and L. A. Saidane, "A multi-objective genetic algorithm-based adaptive weighted clustering protocol in vanet," in *Proc. of Evolutionary Computation (CEC), 2015 IEEE Congress on*, pp. 994-1002, 2015. Article (CrossRef Link).
[28] J. E. Alvarez-Benitez, R. M. Everson, and J. E. Fieldsend, "A MOPSO algorithm based exclusively on pareto dominance concepts," *Evolutionary Multi-Criterion Optimization*, pp. 459-473, 2005. Article (CrossRef Link).
[29] A. Lipowski and D. Lipowska, "Roulette-wheel selection via stochastic acceptance," *Physica A: Statistical Mechanics and its Applications*, vol. 391, pp. 2193-2196, 2012. Article (CrossRef Link).
[30] S. Khan, A. R. Baig, and W. Shahzad, "A novel ant colony optimization based single path hierarchical classification algorithm for predicting gene ontology," *Applied Soft Computing*, vol. 16, pp. 34-49, 2014. Article (CrossRef Link).

Farhan Aadil received his B.S. degree in Computer Science from Allama Iqbal Open University, Pakistan in 2005. He pursued a career in the computer science for 4 years (2005 to 2009). He received his M.S. degree in Software Engineering from University of Engineering and Technology, Taxila, Pakistan in 2011. Now he is a Ph.D. candidate at University of Engineering and Technology, Taxila, Pakistan. His research interests include Vehicular ad hoc Networks, Machine Learning, and Evolutionary algorithms.

Salabat Khan received the B.S. degree in Computer Science from Virtual University, Lahore, Pakistan, in 2007, and the M.S. and PhD degree in computer science from the National University of Computer and Emerging Sciences, Islamabad, Pakistan, in 2009 and 2015 respectively. He is currently working as assistant professor in COMSATS Institute of Information Technology, Attock, Pakistan. His current research interests include data mining, pattern analysis, medical image processing, bio-informatics, and bio-inspired algorithms.



Khalid Bashir Bajwa received BS Computer Engineering from National University of Sciences and Technology (NUST). He pursued a career in the computer engineering for 4 years (June 2002 to Sep 2006). He received MS from Queen Mary University of London, UK in 2007 and PhD from the same university in 2010. He has worked in signal processing and computer vision. He is working as assistant professor at University of Engineering and Technology (UET), Taxila, Pakistan.

Muhammad Fahad Khan received the B.S. degree in Telecommunication & Networks from COMSATS Institute of Information Technology, Attock, Pakistan, in 2016. Currently he is pursuing his MS in Computer Science from COMSATS Institute of Information Technology, Attock, Pakistan. He is also working as Research Associate in the same department. His current research interests include data mining, bio-informatics, and bio-inspired algorithms.

Asad Ali received the B.S. degree in telecommunication & Networks from COMSATS Institute of Information Technology, Attock, Pakistan, in 2016. His current research interests include ad hoc networks, bio-informatics, and bio-inspired algorithms.