Advanced QoS Routing Algorithm in Wireless Ad-Hoc Networks

Hussein El-Attar, Mohamed Ibrahim El-Emary
Arab Academy for Science, Technology and Maritime Transport
Egypt
hattar@aast.edu
Mohamed.i.emary@gmail.com

ABSTRACT: *QoS configuration is needed to maintain the global network needs as Maintaining Max Network Through put, Maintaining Minimum Network Consumption, Assigning Specific Profiles upon Organization's Request and Targeting Prioritizing Service Delivery Based On The Constraints Faced.*

This paper will introduce an optimization solution for multi-constrained routing problem to assess the potential benefit of QoS routing. Multi Constrained Routing Algorithm (OMCR) will be analyzed throughout the paper including simulation results by Network Simulation program (NS2). Comparing its advantages over other routing algorithms. OMCR main disadvantage is that it deals with constraints separately while fetching best path to destination, This barrier prevents network to deal with mutli constraint behavior effectively. In this paper, Advanced OMCR (A-OMCR) protocol is proposed to optimize the quality of service constraints such as energy, delay and hop count. Main aim of AOMCR is fetching feasible path from source to destination while optimizing multiple QoS constraints simultaneously to overcome the main disadvantage of OMCR Algorithm.

Keywords: QoS routing, Quality of Service, Traffic classes, Performance evaluation, Packet delivery guarantee, Multi constraint problem.

Received: 14 May 2017, Revised 18 June 2017, Accepted 25 June 2017

1. Introduction

Ad hoc network methodology is always concerned with perfect QoS achievement. Several barriers can stand against succeeding this target, Such as [1] :

• Difficulty of updating the global network state at appropriate time at each node due to the dynamic nature of network.

• Addressing QoS routing subject to only a single constraint such as delay or bandwidth.

• Considering either Multi constrained path computation or optimization even though they are much related to each other.

• Computing only shortest paths without satisfying multiple constraints simultaneously.

Constraints are categorized upon their act of calculation into additive, multiplicative and concave constraints [1].

All the above constraints always affect feasible path computation process. Time behavior for computing paths is bounded by NP problem.

QoS Architecture consists of routing protocols and QoS algorithms with which QoS routing is mainly achieved and processed. Routing protocols main role is to save the network state information and delegate it throughout the network nodes.

QoS Routing Protocols are classified into [1]:

• Bandwidth Estimation Based Routing

• Position based QoS Routing, Backbone Based QoS Routing

• Multi Path QoS Routing, Multi Cast QoS Routing

• QoS Routing with Resource Allocation

• Constraint Based QoS Routing.

In this paper, we will focus on Constrained based QoS routing on which our proposed QoS algorithm is built. Constraint Based Routing is based on dealing with multiple constraints while fetching feasible path. Several challenges are faced based on constraint type and its polynomial time consumed fetching feasible path.

Example for Constraint Based Routing is in [2] which proposed an algorithm for bandwidth-delay based QoS routing. The algorithm first removes all links which do not satisfy the bandwidth constraint and then finds a shortest path in terms of delays. The Main Constraint based routing protocols used in ad- hoc networks are OLSR, AODV, DSDV and DSR protocols. [3]. AODV protocol characteristics analyzed in [4][5] showed that Constrained Based Routing Protocol (AODV) had provided a set of “All Best Paths” satisfying multiple constraints. Now, QoS Algorithm comes finally to combine all those paths to conclude one “Feasible Path” that satisfies multiple constraints required by the system. QoS algorithms use the output information from routing protocols to compute feasible paths. QoS Algorithms main challenge is to solve the Multi Constraint Optimal Path (MCOP) problem. Multi constraint optimal path problem is the problem of finding best feasible path among set of paths facing multiple constraints [1]. In other words, it is the second phase of feasible path computation while first phase was done by AODV protocol. While implementing QoS algorithm, Classification for MCOP problem should be considered. MCOP classification is as follows:

Path Restricted (PR) Problem:

QoS Algorithm best behavior is determined upon the constraint targeted to be achieved within the path. For example if bandwidth is our constraint to be achieved by the QoS algorithm used, Hence our problem is called Bandwidth Restricted Path (BPR) that need to be considered while building the algorithm. So by taking bandwidth as the priority constraint, Algorithms solving the BPR problem are two types defined as Widest- Shortest Path Algorithm (WSP) and Shortest-Widest Path Algorithm (SWP) [6].

All Hops Optimal Path (AHOP) is a variant of SWP algorithm that tries to reduce network cost while achieving the load balancing [7].A variant of SWP algorithm known as Maximally Disjoint Shortest Widest Path (MADSWP) is proposed in [8]. Optimized multi constrained Routing Algorithm (OMCR) is classified as an SWP algorithm that aims to fetch least hop count path while achieving best bandwidth assignment.

OMCR algorithm is an SWP algorithm that deals with Hop count and delay on separate basis in calculation. For example, if shortest path is the main constraint, OMCR first search for shortest paths then on later iteration it chooses least delay path among the set of shortest paths. This methodology doesn’t help the dynamic nature of ad-hoc network which can’t guarantee

the shortest path set to be static until least delay is calculated. Proposed A-OMCR algorithms calculates least hop count, delay and energy consumption simultaneously at one time on same set of feasible paths. A-OMCR algorithm enhancement is with dealing with multi constraints simultaneously and not separately as dealt by OMCR algorithm. Conclusively, A-OMCR treated preserving ad-hoc network dynamicity that OMCR failed to achieve and increased the number of constraints that algorithm can handle reaching three constraints: hop count, energy and path delay.

In section 2, Related work and studies are mentioned. In Section 3, The Proposed A-OMCR algorithm will be introduced, main contribution will be described in details showing A-OMCR control messages flow, New algorithm for selecting minimum hops feasible path and energy consumption optimization equations. In Section 4, OMCR and Proposed A-OMCR algorithms were compiled and their results will be compared. Last section 5 contains the conclusion.

2. Related Work

Several comparisons and compilations were carried to compare various routing protocols like DEAR, DSR, AODV, AOMDV, TORA, DSDV, LAR, PAODV, SPF, LEACH, ZRP, OLSR and WRP protocols with respect to system throughput , code running time and maximum number of constraints each protocol can deal with efficiently. [9], [4], [5], [10], [11], [12], [13].

The algorithm proposed in this paper was based on the idea of the QoS algorithm proposed in [1]. *OMCR* Algorithm can be described as follows: Consider a network that is represented by a directed graph $G=(V,E)$. Where V is the set of nodes and E is the set of links. Each link (i,j) is associated with a primary cost (metric) parameter $c(i,j)$ till $K(i,j)$ parameters on all link.

The Variables can be defined as follows: D_{ij} is the Tentative (hamming) distance. K is the Neighbor node for node i. $D_{jk}(i)$ is the Shortest distance to j reported by neighbor k to node i. SFD_{ij} is the Shortest Feasible distance. S_{ij} is the Successor set chosen for node i towards j. N_i is the array containing All Neighbor nodes to node i, where $k \subseteq N_i$. $w(i, k)$ is the link state weight between i and node k. $L(i, k)$ is the Link between i and k. sd (i,k) is the Shortest Distance of the adjacent link $L_{i,k}$.

Code initialization begins with Node "i" maintaining each routing entry for destination "j" which includes SFD_{ij}, D_{ij} and the successor set chosen for "j" and denoted by S_{ij}. Then Node i maintains neighbor table that records the shortest distance $D_{jk}(i)$ reported by each node k in its neighbor set N_i for each destination j; and a link table that reflects the link state weight $w(i,k)$ for each adjacent link $L(i,k)$.

Hence the role of node being i, k and j is rotating all over all nodes so that all the nodes do the above set of actions to set their routing tables. When a node is "Active", SFD is set to infinity and all the other entries are set to empty indicating a convergence happened in the network. By this, The active node prepares itself for receiving new D_{jk} data. When node "i" receives D_{jk} from neighbor "k", either updates the estimates $D_{jk}(i)$ and without affecting other estimates or node "i" updates $S_{ij}(d)$ for destination "j" based on equations (1) and (2) For all D_{jk} reported by each neighbor "k" and overall neighbors in N_i.

$$S_{ij}(t) = K \mid D^i_{jk}(t) < SFD_{ij} \tag{1}$$

$$SFD_{ij}(t) = \min\left(D^i_{jk}(t) \mid\mid \mathrm{sd}\,(i,k)\,(t)\right) \tag{2}$$

Equation (1) sets the successor set for node i array through time. All neighbor nodes should always report distances to destination node j smaller than the recorded SFD at node i routing table. If and only of this condition is met, S_{ij} array is updated with new k neighbors.

Equation (2) calculates the updated SFD value at node i through time. SFD is calculated based on the minimum value output from optimizing both $D^i_{jk}(t)$ and sd (i,k) (t). sd is computed based on the constraint needed to be calculated beside hop count. Links between nodes are programmed to compute the shortest distance based on secondary constraint. Finally, Node "i" refreshes the shortest distance of each feasible path maintained for "j" and sends neighbors updates if any change occurs, otherwise if node "i" doesn't receive any control messages hence node it remains "Idle. Previous statistics conclude that (1) and (2) make OMCR a loop-free QoS routing algorithm.

The total number of routing entries for node "j" maintained at each node form a directed graph rooted at "j", which is sub graph

of network G denoted by SGj. If routing converges correctly, *SGj* is an acyclic rotating graph in which each node has multiple successors for node "*j*". At any point of time, multiple of SGj can exist for destination "*j*". To achieve routing optimization, *OMCR* constructs *SGj* in such a way that path with shortest distance to destination "*j*" is always maintained according to (1) and (2).

OMCR sends shortest distance only amongst neighboring nodes, like distributed Bellman-Ford (DBF) algorithm, and eliminating expensive routing overhead by distributing link-state information throughout the network. The optimization function used is a combination that considers each link-weight component equally, which is defined by (3)

$$F(p) = \sum \frac{w_i}{k} \quad k = \textit{number of constraints} \tag{3}$$

Where W is the weight component including the metric targeted by the QoS . For example, if hop count is the primary constraint then W_1 Will be an array containing all hop counts for all feasible paths. And if Bandwidth is the secondary constraint then W_2 will contain the bandwidth for all links towards destination node and so on. Finally $F(p)$ is substituted as the optimization function in (3).

3. Proposed advanced optimized multi constrained routing algorithm (A-OMCR)

This section will discuss the proposed A-OMCR Control messages flow, A-OMCR routing then comparing the results compared to previous OMCR Algorithm.

A-OMCR is considered an ad-hoc On-Demand Distance Vector (AODV) routing protocol. On demand routing is always providing the network with Quick adaptation to dynamic link conditions, Low processing, low memory overhead and low utilization rate.

OMCR control messages flow is when source node *S* consults its routing table requesting to send data to a given destination *D*. If valid entry found, towards destination *D*, it uses it immediately. Else, it launches a route discovery procedure which consists in broadcasting, by the source node *S*, a route request (RREQ) message (containing amongst other information: destination's address, destination's sequence number) towards neighbors.

When RREQ is received by an neighbor node, this last consults its routing table to find a new route (the route is new if the sequence number of this route is larger than that of RREQ) towards the requested destination in RREQ.

If such a route is found, a route reply (RREP) message is sent through reverse route (established when RREQ pass through intermediate nodes) towards the source *S*.

In case new route is not found, it updates its routing table and sends RREQ to these neighbors. This process is repeated until RREQ reaches the destination node *D*. The destination node *D* sends RREP to *S* by using the reverse route. It should be noted that the source S can receive several RREP, it will choose that whose destination's sequence number is larger, if destination's sequence numbers of several RREP are equal, that of which the smallest hop counter will be selected.

A-OMCR uses Hello messages to maintain the connectivity between nodes. Each node periodically sends a Hello message to these neighbors and awaits Hello replies. If Hello messages are exchanged in the two directions, a symmetrical link between nodes is always maintained if no convergence occurred. The broken link can be repaired locally by the node upstream, else a route error (RERR) message is sent to the source *S*. This last can launch again, if necessary, the route discovery procedure. It should be noted that the link interrupt is the consequence of the mobility or the breakdown of nodes.

Fig. 1 illustrates the network topology towards destination node 8. Each path between nodes is called arc (i, j) carries a number indicating path delay. To achieve routing, Each node should be marked with both Label and Status marks. The label is a 2 entry output $[u_i, k] = [u_i + u_{ij}, {}_k]$, Where u_i is the least delay count between a node and its next hop. u_{ij} is accumulative counter for delay along the path to *j*. where *i* is the source node and *j* is the upcoming destination node in next iteration. *k* is last hop before reaching *j*. If node is used as a neighbor to reach node 8, hence node's status is visited. Otherwise node is unvisited.

Iterations to fetch shortest distance and Least delay path can be described ad follows:

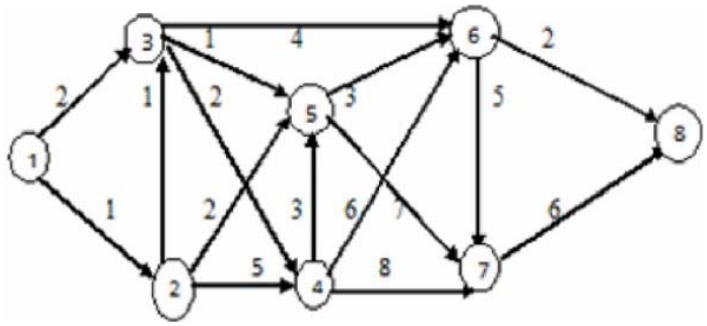

Figure 1. Network Topology towards Node 8

Based on Fig. 1, we can calculate each node label and status passing through iterations. If there are two paths towards same destination, smallest delay and number of hops is chosen.

Iteration 0: Set the first node as visited label [0, -]
Iteration 1: Nodes 2 and 3 can be covered from node 1.

Thus, the list of labeled nodes (Unvisited and visited) becomes between the two unvisited labels [1, 1] and [2, 1], node 2 get the smaller distance (u2=1). So the status of node 2 is changed as visited.

Node	Label	Status
1	[0, -1]	Visited
2	[0+1, 1]=[1, 1]	Unvisited
3	[0+2, 1]=[2, 1]	Unvisited

Table 1. First iteration results

Iteration 8: The iteration will be completed when all the node status as visited. If any unvisited node are there the process to identify shortest path algorithm is not completed properly. The result of final iteration given in table 2.

Node	Label	Status
1	[0, -1]	Visited
2	[1, 1]	Visited
3	[2, 1]	Visited
4	[4, 3]	Visited
5	[3, 2], [3,3]	Visited
6	[6, 3] or [6, 5]	Visited
7	[10, 5]	Visited
8	[8, 6]	Visited

Table 2. Last iteration of shortest path selection

So the shortest path to reach the base station 8 from node 1 through hop nodes gives the following sequence (8) → [8,6] → (6) → [6,5] → (5) → [3,3] → (3) → [2,1] → (1), So the shortest route to reach the base station is 1→ 3 → 5 → 6 → 8.

The individual constituent can be a state-based constituent, because every unit has different energy level consumption in different states. In addition, this constituent involves two different types of transitions: transitions between units and transition between states of a single unit. The overall energy consumption in individual constituents is expressed as follows:

$$E_{individual,\,i}\,(\Delta t) = \Sigma_{u=1}^{Nu}\Sigma_{seS}\Sigma_{wew}(e_{u,x},\, e_{u,w},\, t_{u,x}) \tag{4}$$

Since most of energy minimization methodologies use idle and sleep states for avoid of wasting energy in idle states, the above constraint states that the total energy consumed for switching among states should be smaller than the total energy consumption of states. Energy consumption in an active state for each unit depends on several factors as illustrated in [15].

We suppose a continuous time between t1 and t2 for the energy consumption measurement. Residual energy in time t is defined by omitting consumed energy in Δt from the initial battery power in t-Δt. Thus, the energy consumption will be determined in ”t. The residual energy is calculated using [16]:

$$E_{residual,\,i}\,(t) = E_{initial,\,i}\,(t - \Delta t) - E_{Consumed,\,i}\,(\Delta t)$$

$$E\,(\Delta t) = dE\,/\,dt\,\Delta t \tag{5}$$

$$\Delta t = t_2 - t_1$$

Where The total energy consumption of node i in the interval Δt as follows:

$$E_{Consumed,\,i}\,(\Delta t) = \lambda_1 E_{individual,\,i}\,(\Delta t) - \lambda_2 E_{local,\,i}\,(\Delta t) + \lambda_3 E_{global,\,i}\,(\Delta t) + \lambda_4 E_{battery,\,i}\,(\Delta t) + \lambda_5 E_{snk,\,i}\,(\Delta t) \tag{6}$$

Subject to :

1. $E_{local,\,i} > 0$
2. $E_{global,\,i} > 0$
3. $E_{individual,\,i}\,(\Delta t) + \lambda_2 E_{local,\,i}\,(\Delta t) + \lambda_3 E_{global,\,i}\,(\Delta t) + \lambda_4 E_{battery,\,i}\,(\Delta t) + \lambda_5 E_{snk,\,i}\,(\Delta t)$

The first constrain expresses condition for necessity to establish a collaboration connection. The second constrain shows the necessary and sufficient condition for accessibility of the node in the network. The third constrain means a node should have enough energy to do network tasks otherwise it is not active and should be removed from the network calculations. Each constituent is expressed in terms of key parameters (or factors). These key factors are determined based on application requirements. On the other hand, these parameters may influence more than a single constituent; hence energy constituents may partially overlap.

Consequently, the interplay among energy constituents must be taken into account in evaluating the overall energy consumption of the entire setup. For example, the number of neighbors determined by topology in the global constituent has direct influence in energy consumption of the local constituent. The node which sensed the object using sleep wake up method [16] , routes the information packet based on the shortest path algorithm mentioned above. During the routing the energy level of hop nodes calculated and it compare with other nodes in the network. That is shortest path algorithm executes based on the energy level and shortest path. The nodes which are having maximum energy will be included in the routing path. The result will be the best feasible path towards the destination node.

Once the shortest path, Nodes with maximum energy level has been identified, based on that the information will be routed. This is called primary path routing. Hence data packets will pass through that route normally to destination.

A-OMCR keep tracking the energy level of hop nodes it compare with other nodes in the network as the mathematics model discussed above. If the energy of primary path nodes reduced at particular level (20 percent of energy remaining), the “low energy” nodes send control messages to the source node indicating to not consider these nodes as neighbor nodes , hence the routing path change and priority given to energy level path then, shortest path. The nodes which are having maximum energy will be included in the routing path. Thus secondary path is generated.

4. Simulation results

Table 3 provides the network setup configuration, NS2 simulator was used to compile both OMCR and A-OMCR algorithms.

Parameter	Value
Simulator	NS-2 (version 2.35)
Channel Type	WirelessChannel
Network Interface Type	Phy/WirelessPhy
Mac Type	Mac/802.11
Interface Queue Type	DropTail/PriQueue
Link Layer Type	LL
Antenna	OmniAntenna
Maximum Packet in ifq	256
Area	500 X 400
Radio-Propagation Model	TwoRayGround
Number of Mobile Nodes	50
Simulation Time	120 sec
Initial energy in Joules	40
Routing Protocols	AODV

Table 3. Network Setup

Fig. 2.shows hop count vs time. A better performance of A-OMCR algorithm than OMCR algorithm with respect to hop count from source to destination node is observed. Those results were based on the frequent and successive control messages used by A-OMCR that makes all network components in a meshing format exchanging important messages leading to perfect hop count. Average Hop count ration between A-OMCR and OMCR is 0.5 : 3 hops respectively. The decrease in OMCR hop count graph (at point 20 msec) is due to that OMCR had picked it's primary path and stabilized on 4 hops path , while on point 40 msec we can see that hop count stabilization is lost as OMCR algorithm started to pick secondary path searching for best delay path.

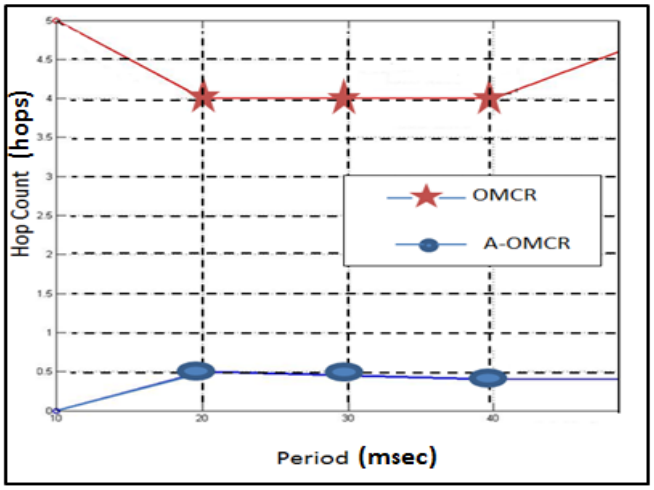

Figure 2. Hop Count vs Time

Node energy consumption is calculated in both protocols during the running period, as shown in Fig. 3, A-OMCR nodes are less in energy consumption than the OMCR Nodes with average ratio of 97 : 43 Joules respectively. Delivery delay between source and destination node is shown in Figure 4 showing success for A-OMCR algorithm on OMCR algorithm by decreasing packet delivery delay from 12 msec to 1.8 msec for A-OMCR.

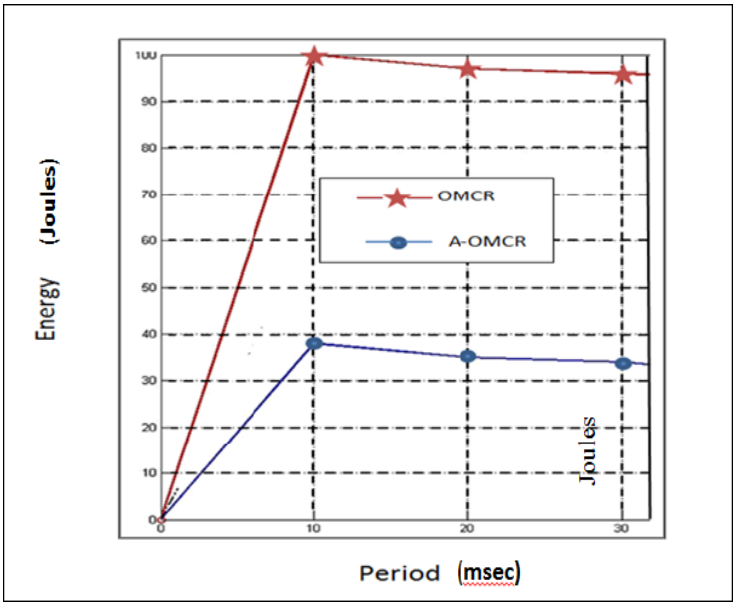

Figure 3. Energy Consumption vs Time Period

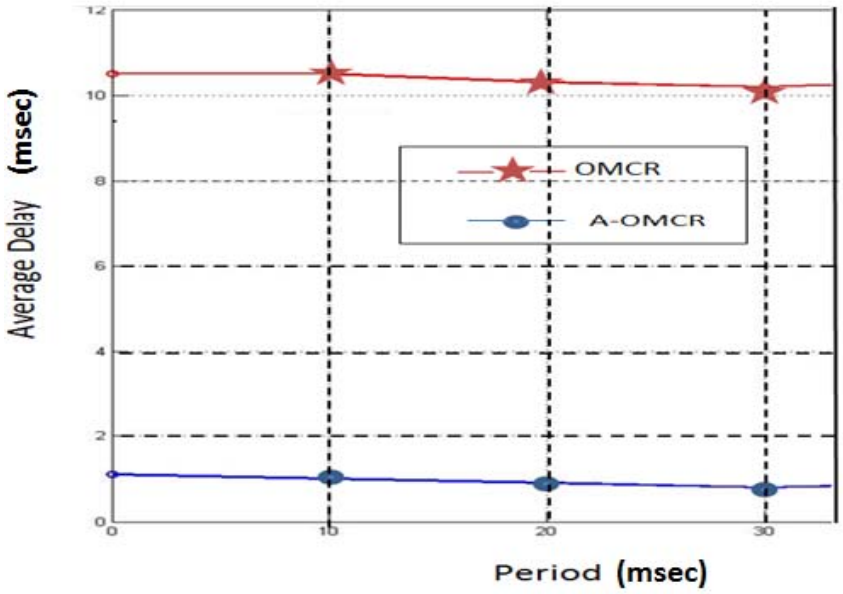

Figure 4. Average Delay vs Time

Based on previous gathered information related to both latest routing algorithms OMCR and A-OMCR , we can conduct an informative statistical table as shown in table 4 indicating the main differences between the two protocols from the mechanism view and results enhancement perspective.

	Proposed A-OMCR	OMCR
Routing Protocol Type Used	Table driven and Source routing	Source Routing
Route Maintained in	Routing Table	Route cache
Route discovery	On demand	On demand
Multiple route discovery	Yes	Yes
Control messages broad cast	No , Reducing algorithm overhead	Yes
Multicast	Yes , Only to successor nodes	Yes , To all nodes
Reuse of routing data	Yes , Unless it is reported from energy consumed node	Yes
Metrics in use while computing feasible paths	Hop count , Node energy Consumption and Link Delay	Hop Count and Link delay
Feasible path computation technique	Using shortest path algorithm	Using Sequential filtering
Metric computation efficiency	Gives efficient results when computing 3 metrics simultaneously, efficiency decreases when increasing number of metrics	Good when using one metric, Decreases by increasing more metrics in computation
Topology Flexibility	Flexible (position distributed)	Fixed Square grid

Table 4. Comparison between OMCR and A-OMCR

5. Conclusion

A-OMCR is an on demand table driven protocol. Which gives the protocol flexibility in dealing with route indexing.

In case of network convergence, A-OMCR using AODV protocol distinguish itself over other protocols by "Erasing route then source notification or local route repair" Method. This Method enables the protocol to erase the affected route from routing table then notifies the source node at once with the failure.

A-OMCR also is characterized by it's high adaptability to dynamic topologies, which is the exact need for ad-hoc networks. Reusing routing data is disabled by default in A-OMCR. Meaning , if a route is dropped due to node failure , it is directly flushed from the routing table. This behavior results in higher system throughput and processing speed serving A-OMCR algorithm success.

A-OMCR keep tracking all network nodes information, messages exchanged between them and changes of convergence ongoing throw the network while updating all nodes. As conclusion, A-OMCR algorithm had succeeded in maintaining dynamic network stability by fetching best feasible path while reserving overall energy consumption and hop count.

References

[1] Prakash, P.S. (2010). Investigations On Additive Multi Constrained Quality Of Service Routing, p. 26 – 30.

[2] Wang., Crowcroft. (1995). Bandwidth-delay based routing algorithms, *In:* Global telecommunications Conference, GLOBECOM 95' IEEE.

[3] Narsimaha, V.B. The Performance Comparison of an AODV, DSR, DSDV and OLSR Routing Protocols in Mobile Ad-Hoc Networks.

[4] Shah, S., Khandre, A., Shirole, M., Bhole, G. (2008). Performance Evaluation of Ad Hoc Routing Protocols Using NS2 Simulation, *In:* Mobile and Pervasive Computing (CoMPC-2008), p. 167-171.

[5] Azad, S., Uddin, M., Anwar, F., Rahman, A. (2008). Performance Evaluation of Wireless Routing Protocols in Mobile WiMAX

Environment.

[6] Qingmin., Steenkiste. (1997). On path selection for traffic with bandwidth guarantees.

[7] Roch, Gurein., Ariel, Orda. Qos routing in networks with inaccurate information Theory and algorithms.

[8] Taft-Plotika, N., Bellur, B., Ogeir, R. (1999). Quality of Routing Using Maximally Disjoint Paths July 2003.

[9] Hong-ryeol, Gil., Joon, Yoo., Jong-won, Lee, (2002). An On-demand Energy-efficient Routing Algorithm for Wireless Ad hoc Networks.

[10] Singh, R., Singh, D.K., Kumar, L. (2011). Performance Evaluation of DSR and DSDV Routing Protocols for Wireless Ad Hoc Networks.

[11] Fernandez, L., Pedro, A. (2008). Routing Protocol Performance Evaluation for Mobile Ad-Hoc Networks, UNF Thesis and Dissertations, p. 293.

[12] Das, S.R., Perkins, C.E., Royer, E.M. (2001). Performance Comparison of Two On-Demand Routing Protocols for Ad Hoc Networks. *IEEE Personal Communications*, 8, p. 16-28.

[13] Usop, N.S.M., Abdullah, A., Abidin, A.F.A. (2009). Performance Evaluation of AODV, DSDV & DSR Routing Protocol.

[14] Arbunar, B.C., Grama, A., Vitek, J., arbunar, O.C. (2010). Redundancy and Coverage Detection in Sensor Networks.

[15] Najmeh, Kamyabpour., Doan, B.Hoang. Modeling overall energy consumption in Wireless Sensor networks.

[16] Nagarajan, M. (2012). "A New Approach to Improve Life Time Using Energy Based Routing in Wireless Sensor Network. *International Journal of Science and Research* (IJSR).