Wireless Pers Commun (2012) 63:529–548
DOI 10.1007/s11277-010-0148-8

STAB-WIN: Self Organized, Topology Control Ability Backbone Node in Wireless Networks

S. Smys · G. Josemin Bala

Published online: 14 September 2010

Abstract The objective of the paper is to construct a backbone node with self organization, topology control and reconfiguration capabilities. The key issues in wireless networks are maintain a topology with minimum degree, self organization during link or node failure and reconstruction ability when the backbone changes the position. Existing research works concentrate on any one of the issues by a backbone, but nodes in wireless are battery operated. To solve the all issues separately more power is required. To overcome the existing issues we propose a localized approach namely STAB-WIN, which will solve all the issues without affecting the entire system performance using local updates. This research work focuses on multiservice ability of a node to meet the design goals of next generation networks. Our approach is witnessed by the simulation results on analyzing the parameters like scalability which includes backbone size, routing overhead, control transfer and QoS parameters.

Keywords Self organization · Topology control · Backbone construction · Wireless network · Localized approach

1 Introduction

Wireless networks are two types; namely infrastructure oriented which uses wired access points and Mobile Ad Hoc Networks (MANET), where each device in the network acts as routers. Addition of wired router in infrastructure based node performs better performance than wireless ad hoc networks [1]. MANET is a self organized network as the nodes are autonomous. Terminals are move anywhere at any time. However, the properties of frequent route breakage and unpredictable topology changes in MANET still make most of these traditional

S. Smys (✉)
ECE Department, Karunya University, Coimbatore, Tamilnadu, India
e-mail: smys@karunya.edu

G. J. Bala
Electronics and Media Technology, Karunya University, Coimbatore, Tamilnadu, India
e-mail: josemin@karunya.edu

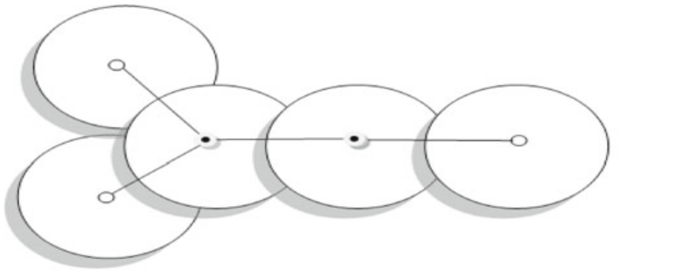

Fig. 1 Backbone formation in unit disk graph

routing protocols inherently not scalable with respect to number of nodes, control overhead, and degree of mobility. Our approach deal with four key areas in wireless domain, starting with how to construct the backbone, secondly perform reconstruction during the movement of nodes, thirdly reduce the number of connections between backbone nodes with minimum degree maintenance, and finally self organization mechanism included in the backbone nodes to maintain connectivity during link and node failure. A common source of overhead in a MANET comes from blind flooding/broadcasting, which is used in the route discovery process in several reactive routing protocols. The broadcast nature of wireless communication (i.e., when a source sends a message, all its neighbors will hear it), result in excessive redundant transmission. Redundant transmission may cause a serious problem, referred to as the broadcast storm problem [2], in which redundant messages cause communication contention and collision. To overcome this problem Virtual Backbone (VB) routing is proposed, where set of nodes act as a backbone which makes the routing decision or in other words it may define as few network nodes are selected as dominating points to access other nodes in the network. The same can be referred as connected dominating set (CDS) [3–5], which forms an interesting virtual backbone. The example network in one hop backbone in unit disk graph is given Fig. 1. In this illustration only two backbones are selected for carrying other three nodes traffic. A dominating set (DS) of a graph is a subset of nodes such that each node in the graph is either in the subset or adjacent to at least one node in that subset. A connected DS (CDS) is a DS in which all nodes are connected to at least one other node. More number of proposals in the backbone routing [6,7] concept is done earlier. Those papers are deal with how to reduce the backbone size i.e. construct Minimum Connected Dominating Set (MCDS), but none give the solution for all in one node issue, but we focus the problems and give the feasible solution with comparing existing results.

Secondly the proposed method focuses on reconstruction issues. Consider a backbone node is changing the position; reconstruction must be initiated by the network. During the movement of the normal, backbone node and when it stops travelling, three time calculations are required to update or reconstruct the backbone. It will cause extra load, which degrades the network performance in terms of latency, contention and collision. To overcome this issue we give the localized algorithm for updating the backbone locally or when there is change in nodes attach with backbone node. The localized reconstruction algorithm given [8], perform better results in terms of reconfiguration with the help of identifying new neighbors and control packet updates. But still the other issues are not addressed. Thirdly we focus on minimum degree maintenance, if more number of nodes are adding with one backbone node during the movement; the backbone is not able to support routing information between nodes, because all the nodes including the backbone nodes are connected via one or two hop distance. To reduce the number of links and make routing by the backbone easier minimum degree maintenance is used which support topology control. Finally the mechanism for self organization in backbone based wireless network is proposed to solve the link and node failure. It is also used to minimize the number of message transmission, reception and can help to conserve energy. The reconstruction of backbone and topology control will leads to the self organization capability of backbone node. The motto of our proposed method is "*ALL IN ONE NODE*".

1.1 Issues

The networking nodes in wireless networks are very limited in resources, backbone based routing approach also falls to the same category, therefore it should be designed with low control transfer support and computation cost. Several algorithms are proposed for constructing backbones, but only few of them consider the backbone maintenance and reconstruction. In addition, number of nodes attached, node and link failure analysis are also scalable as the wireless networks are typically deployed with large network size. The above issues must be satisfied by a backbone node during movement and stable conditions or in other words the node must have a self organization capability to meet the QoS requirements of the network.

1.2 Problem of the Conventional Work

Most of the protocol design procedures for mobile networks are based on the basic assumption that each node cooperates to establish a network and all the nodes involve in the active routing for better performance. Nodes are controlled by the source node which will give the routing information (virtual circuit method) or each node involved in the routing like datagram approach. Extra care required for source to find the route to destination in the former method, more control transfer required in the later case. Any way both methods suffer to process excess control packets to reach information to the destination. Conventional backbone formation algorithms suffered with lot of control transfer exchanges, i.e. control overhead required during the backbone node change the position, added with new node (when it stop travel) and movement of other nodes attached in the backbone. Three time calculations are required to update or reconstruct the backbone. It will cause extra load and degrade the network performance in terms of latency, contention and collision. To overcome this issue a localized algorithm is proposed in this paper for update the backbone locally or when there is change in nodes attach with backbone node. All the conventional routing methods only provide the route to destination, not involve in alter the links of the nodes, suffered with self organization with low energy consumption.

1.3 Motivation

In wireless domain, all the nodes forward the information to the next interface and reduce their lifetime because all nodes are battery operated. And also the processes like computing, table updations also account to the lifetime of the node. We think this problem in alternate way: each node must work with their own desire: conserve their resources and work in self organized manner. For energy consumption centralized approaches like cluster based routing is not suitable because the whole network is based on the clusters as well as their coverage. The two fundamental concepts in MANETs are selfish behavior of each node and coverage

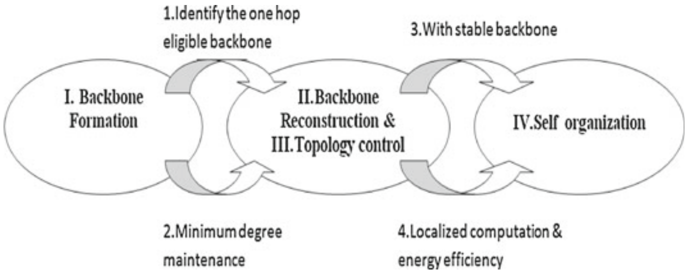

Fig. 2 Design flow of STAB-WIN

of the node. These ideas are motivated us to develop this proposed work. To implement this work all the nodes are permitted to act as a self behavior randomly, a node acting as backbone must sacrifice some characteristics like low speed node movement. Several algorithms [8,9], and [10] have been proposed to create backbone constructions. These algorithms are deals with how to reducing the backbone size and not motivate the selfish behavior of the nodes. Modeling systems based on selfish algorithm [17,18] are developed to improve the performance of the ad hoc network, and they do not guarantee the global solutions. This work is follow-up to [8], where the authors guarantee local computations for reconstruction but no support to self organization. The objective of the research work is more of practical importance and has been underscored in [17,20] which provide excellent motivation to consider self organization behavior in backbone design.

1.4 Proposed Work

The proposed methodology STAB-WIN and the entire design flow is given in Fig. 2. It deals with the self organization capability of backbone node with topology control and reconstruction ability. This is achieved by distributed localized algorithm and integrates the all tasks (construction, reconstruction, topology control and self organization) by a backbone. Initially a graph is constructed from the available nodes V and links E, i.e. say G (V, E) and one hop connection between each node is maintained. From the graph set of nodes are elected named as backbones based on the modified greedy algorithm and it will be formulated using linear integer programming.

It is mainly used to carry other node traffic, especially involved in control traffic. After the backbone construction to maintain the stability of the network second level backbones are identified based on available power, named as Eligible Nodes (EB), they are involved in alternate routing during the backbone movement or backbone failure conditions. This process is named as reconstruction of backbone. If number of connections is added to a particular backbone node, it affects the entire system performance. To avoid this scenario after the backbone creation, only limited connections are permitted for each backbone henceforth will not affect the entire network performance. Connections are also shared between one hop eligible backbone and the process is named as minimum degree maintenance, which will result in topology control. Finally the self organization characteristics of the backbone is achieved by adding reconstruction and topology control property i.e. links altered by topology control, node failure identify by the backbone and perform alternate routing is achieved by reconstructed eligible backbone node. In this way alternate routing and minimum degree maintenance play a major role in self organization.

1.5 Organization

The remainder of the paper is organized as follows: Sect. 2 describes the related work on the virtual backbone formation and self organization, Sect. 3 presents the preliminaries and notations of our approach STAB-WIN. Section 4 describes the construction of backbone with mathematical mode. Algorithm for reconstruction of virtual backbone is given in Sect. 5. Sections 6 and 7 illustrate the solution for topology control and self organization. Section 8 shows the simulation results and performance analysis. Finally Sect. 9 concludes the paper.

2 Related Work

Dai and wu in [9], a distributed localized algorithm is proposed in which local computations are used to compute the connected dominating sets. This paper state that the efficiency of dominating-set-based routing mainly depends on the overhead introduced in the formation of the dominating set and the size of the dominating set. The k-connected dominating set with energy efficient routing is given by Yang et al. in [10], in which k-dominating set can be used to disseminate topology update packets or route request packets, in which the search space is reduced to the set. In [11], Park et al. gives the idea about the backbone with various transmission ranges and this paper focus on reducing the CDS size with some QoS sacrifice. Currently, multi-hop wireless routing protocols fail to provide good quality of service, especially in the presence of a large number of connections concentrating on some nodes. These effects result in poor packet delivery, long packet latency, high signal interference and high routing overhead. To overcome this effect we first perform a connection balance between backbones and which yields to topology control. The general understanding is that dividing the traffic flow among a number of paths (instead of using a single path) results in a better balancing of load throughout the network. Afandi et al. in [12] and [13], gives a solution for on demand and distance vector protocols AODV and DSR, but these protocols suffered with optimum route selection and self organized topology control. However, Bao and Garcia in [14], informs that unless using a very large number of paths the load distribution is almost the same as single path routing. Unlike the entire above load balancing routing protocols, some of the routing protocols focus on balancing the load for the whole network without knowing the current load [15] information. Our proposed approach of routing protocol also falls into this category. To discover a simple but efficient load balancing routing technique, we propose an algorithm to balance the connection without the knowledge of load information and no additional communication overhead. Existing wireless routing protocols are based on flat architecture and some of them are based on hierarchical architecture [16,17] and some of them uses clusters [18,19]. In this method selection and maintaining the cluster heads are the major issues and these methods are not to support self organization characteristics of the node. To overcome these problems a backbone node is constructed with self organized and topology control ability is proposed in this paper.

3 Preliminaries and Notations

3.1 Preliminaries

In this section some notions and preliminaries are introduced for analysis. In STAB-WIN, backbone node must be maintained as a header for long time, to do this speed of the backbone must be maintained low and transmission power should be high compare to other nodes in the network. Normally each node has two states: (1) Normal state—N state, (2) Backbone state—B state. Hereafter we use B or N state throughout the paper. For B-state node speed reduce to one fourth of other nodes and in N-state node has no restriction. Consider a wireless network consisting of a set V of wireless nodes distributed in a unit disk graph. The maximum power of a backbone node is different from other nodes in the network. These wireless nodes form a graph, in which there is an edge between two backbone nodes if and only if two nodes are connected with one hop (i.e., these two backbone nodes can always receive the signal from each other directly). Hereafter, assume the network is a connected graph. Let G be a graph G = (V, E) where V represent the vertices (all nodes) and E contains edges (links) between nodes. In V, the set of backbone nodes is $B = \{b1, b2, b3, \ldots, b_n \epsilon V\}$ and the non backbone set is $N = \{n1, n2, n3, \ldots, n_n \epsilon V\}$ where B, N ϵ V.

3.2 Definitions and Notations

For textual representation the following short words are used.

- **CDS**: Connected Dominating Set—A connected dominating set of a graph G is a set V of vertices with the following property: any node in V can reach any other node in V by a path that stays entirely within V.
- **DS**: Dominating Set—Every vertex in G either belongs to V or is adjacent to a vertex in V.
- **VB or BN**: The Virtual Backbone or Backbone Node. It handles other node control transfer specially in routing and maintains the minimum degree.
- **NB**: Non Backbone—active member in the network.
- **EB**: Eligible Backbone—one hop neighbor to backbone having high power compare to other nodes attached in the backbone (the node identified by the following parameters: non member of CDS, one hop distance from VB node).
- **RB**: Reserved node—based on available power levels, it is a low power node (third level) compares other nodes, normally backbone nodes BN-first level, and EB-second level.
- **STAB (S or M)**—Self Organized, Topology Control Ability Backbone with nodes are in Stable(S) or Mobile (M) condition. In result analysis we use this short word.

Special symbols for **algorithm** description:

The symbol ***e, r and ***b*** indicate the eligible, reserved and backbone nodes in the algorithm.

4 Construction of Backbone

Backbone formation is formulated using linear programming approach. If a graph say G = (V, E), the dominating set can be obtained as follows; A variable d_i is a non–dominating variable, e_i—one hop eligible node for alternate routing (optional—applicable for position change of backbone).

$$d_i = \begin{cases} 1 & \text{if the node i is an non} - \text{ dominator (edge nodes in the graph)} \\ 0 & \text{otherwise} \end{cases}$$

Algorithm 1 for backbone formation process:

Construct Spanning Tree (based on power level hierarchical)
// backbone—a high transmission power node
1: INPUT: An undirected disk graph $G = (V, E)$
2: OUTPUT: A connected dominated set with one hop eligible and reserved node S
// eligible node—one hop with backbone second level transmission power
// reserved node—also one hop with backbone third level transmission power
3: select a nodes ***b, e, r*** in V
// e, r and b—eligible, reserve and backbone node to support alternate routing
4: *Construct Tree* CT $\leftarrow ((V, E), \boldsymbol{b}, \boldsymbol{r}, \boldsymbol{e})$
5: connect all edges of ***e, r*** with ***backbone b*** form a new 3 connected one hop tree
6: R, E subset of V // R is set of reserved nodes.
7: return $CT \cup$ E
8:**Subroutine ConsTree (V, E, *r, e*):**
9: color all nodes BLACK
10: backbone B $\leftarrow$ {b}:
11: **while** B contains at least one RED node **do//** Initial all nodes in BLACK
12: select one hop node r, e from CT and color it BROWN for e and GREEN for node r
// B- nodes are maintained color red
// b at least two nodes based on hierarchical power range
13: **add** each WHITE node b in B (v) with one hop b_1, e_1, r_1 etc
14: **if** BLACK node b in *movement*
15: add one hop e node in CT
// update with one hop b_1, e_1, r_1
16: e_1(or)$r_1 \leftarrow$ b
// change the colors and set the priority
17: **end if**
18: **end if**
19: return all RED and GREEN

The dominating set obtained from the following relation:

$$\mathrm{G} - \mathrm{Max} \sum_{\mathrm{i} \epsilon \mathrm{V}} \mathrm{d_i} + \mathrm{e_i}\ (\text{optional})$$

Dominating set is obtained from subtract the edge nodes from the actual graph (G).

4.1 Algorithms for Backbone Formation

The coloring scheme is used for backbone formation algorithm to identify the different nodes like backbone, eligible nodes. Initially all nodes in the unit disk graph is colored black and based on spanning distance and the power levels each nodes are colored differently (for simplification in the algorithm the nodes are named as: ***e, r*** , and ***b***—eligible, reserve and backbone node, colored as: ***b***—red, **e**—brown, **r**—green)

In order to construct the backbone a coloring scheme is used to differentiate the edge nodes are subtracted from the actual graph. We use the modified greedy algorithm for virtual backbone construction. Normally Dijsktra algorithm used to find the shortest path between nodes and kruskal algorithms used to find the minimum spanning tree (MST).The following steps are used to construct virtual backbone Calculate the spanning tree of the network; Non selection of edge node and subtract the edge nodes from the actual graph. In this way only half of the nodes will act as a virtual backbone node, which will give extra life time for the network. All the backbone nodes are connected based the property given in Chap. 4.2.

Fig. 3 Control packet format

0/1	0/1	POWER

4.2 Property-1: For Backbone Connection Procedure

Let $v_1, v_2, v_3, \ldots vk$ be the number of backbone nodes in the network, the total non backbone nodes in the network is given by n > 3NΔ, n > 2 NΔ, n > NΔ for selection of 3hop, 2hop and 1hopVB connections. We follow the one hop connection between backbone nodes. The maximum number of backbone node connectivity of the ith system is given by $V_i \geq N/\Delta$ (i + 1) for i = 0, 1, 2…6.Therefore, for m virtual backbones:

$$\sum_{i=1}^{m} V_i \geq N/\Delta\,(i+1),$$

where, 'N' is the number of nodes in the network and Δ is the degree of each node. Connectivity of a network is defined as the probability of the number of one hop backbone connecting from one non backbone to another node in the graph. All the backbone nodes are connected using the property-1 as the virtual backbone node connect with one hop neighbor.

5 Reconstruction of Backbone

5.1 Reason for Reconstruction of Backbone

The purpose of reconstruction is to identify the one hop nodes for alternate routing i.e. during information exchange if the backbone or normal nodes change their position. The following factors are consider for reconstruction (1) movement of backbone node (2) when it stop travelling (3) other node movement. In this three process updating and new link formation are needed. To solve these issues we develop a localized algorithm, which is used to update the local information when there is a change in any one the above mentioned factor. Assume that all the nodes follow any one of the following four states; backbone Stable, backbone mobile, stable nodes, mobile active nodes**.** During motion the backbone and non backbone nodes exchange the eligible backbone and battery power information to their one hop neighbor. A backbone node must perform the following function for reconstruction. Firstly exchange hand over information to one hop neighbor before change the position, identify the Eligible Backbone and their battery power from the control packet exchange. During the state change process, the backbone node exchange the routing table information to one hop nearby node, similarly non backbone node indicates the routing information to one hop backbone. To maintain a dominating set a 10 bit control packet as shown in Fig. 3 is used. This packet used for local updating between backbone and other nodes.

Packet field details: for 00- backbone Stable (enter a new state, stop travelling), 01—backbone in movement, 10—Non-backbone Stable (enter a new state, stop travelling), 11—Non-backbone in movement. Assume the following situation, if the non backbone node say C, change its state (moved from one place to another), it attaches with a new node say B. Node C exchange all the control information to node B and node B become a new backbone. In this way local updates are used to maintain MANET connectivity. At the same time service

Algorithm 2 for backbone reconstruction process:

1: function backbone reconstruction *G* = (V, E)
2: select a nodes b, ***e, r, b*** *in V*
3: *Construct Tree* CT ← ((*V*, *E*),***b, r, e***)
4: connect all edges of ***e, r*** with ***backbone*** *b* form a new 3 connected one hop tree
Backbone stable:
Functions of ***red*** and ***black:***
If (one hop node change position) {
// the leave process
Send **LEAVE_ BN** to the one hop backbone
Return
}
Update local backbone with **LEAVE_ BN**
Send **JOIN_BN or JOIN_NB** to one hop neighbor
// backbone update with one hop neighbor
Backbone in movement:
Functions of ***green*** and ***brown:***
If (backbone node changes the state) {
// during the movement of **BN**
// to maintain the connection
State (BN) — moving
EB——updated
Update local information
Return
}

provider backbone of node C is node A, delete the routing table information about node C and this information share between one hop neighbors. Our objective in reconstruction process is to create reconfigurable backbone with minimum overheads.

5.2 Algorithm for Reconstructed Backbone

During the movement of mobile nodes, topology and backbone maintenance are the important issues. When the backbone node is in movement, CDS is maintained by sending the ten bit control information to the nearby one hop eligible neighbor, in this way distributed localized computation is used to update node information locally. Local updations include the position and power level changes indicated to one hop backbones to maintain connectivity. On receiving 10 bit control message, backbone functions are described in the following algorithm. To update the power and position information **LEAVE_ BN, JOIN_BN** and **JOIN_NB** messages are used in the reconstruction algorithm. A node want to leave the backbone it uses **LEAVE_ BN** message to one hop backbone. Similarly the node wish to join to other backbone or normal node it uses the other two messages.

6 Topology Control

Topology control is a technique used mainly in wireless networks in order to reduce the initial topology of the network to save energy and extend the lifetime of the network. The main goal is to reduce the number of active nodes and active links, preserving the saved resources for future maintenance. The topology control is initiated using minimum degree maintenance is briefly outlined in this chapter.

0/1	0/1

Fig. 4 Control message format for maintaining minimum node connections. Messages from backbone: 00—not loaded, 10—connection response (indicate the one hop node for connection establishment), 11—fully loaded. Message from non backbone node: 01—connection request

6.1 Motivation of Topology Control and Minimum Degree Maintenance

Due to the nature of mobile wireless networks, node moves anywhere at any time, topology control must be included in wireless nodes. Topology control comprises link and node control issues. The number of nodes attached with a backbone is reduced in each stage with the help of minimum degree i.e. each backbone is permitted with few nodes and link, topology control is achieved. Consider more number of nodes is added with a backbone during the movement, it not able to support and maintain routing information between nodes, because all the nodes including the backbone node connected via one or two hop distance. Currently, multi-hop wireless routing protocols fail to provide good quality of service, especially in the presence of a large number of connections concentrating on some nodes. These effects result in poor packet delivery, long packet latency, high signal interference and high routing overhead. To overcome this effect we first perform the connection balancing between backbones and it will lead topology control. The Theorem 1 given below relates the minimum degree maintenance in backbone domain.

Theorem 1 *A graph G with any two backbone node 2-connected and $VB \geq 7$ non-backbone connections are permitted then a new backbone formed by $VB < 7$ connections provided new connection request.*

Proof By induction on node degrees of backbone. If any two backbone nodes are 2-connected it must connected with two internally disjoint paths and the vertices, edges lie on a common cycle. Suppose assume the situation, number of degrees allowed (i.e. connections per backbone) is 7, the backbone is able communicate all other nodes efficiently. Because two backbones are connected by two internally disjoint paths. Some new node needs the backbone is not permitted by the two backbone nodes, because of the induction hypothesis only seven connections are permitted by each node. Hence the theorem is proved.

6.1.1 Message Exchange between Backbone and Other Nodes for Connection Balanced Routing

In the proposed methodology, the network follows the local computations and updations only are accomplished by the control message. The control packet format given in Fig. 4 is used to maintain the connection balanced routing in backbone domain. To identify the connection status of the backbone only two bit field is required.

The control messages 00, 10, 11 are send periodically by the backbone node for updating the status of the network, 01 is the only message send by the non backbone node for connection request. If it receives the fully loaded message (11) and set the connection request message (01). Based on the request backbone indicate the new one hop backbone neighbor.

6.2 Scenario for Backbone Topology Control

For minimum degree maintenance backbone periodically send the number of connection information to all other nodes in the network. For topology maintenance, each backbone

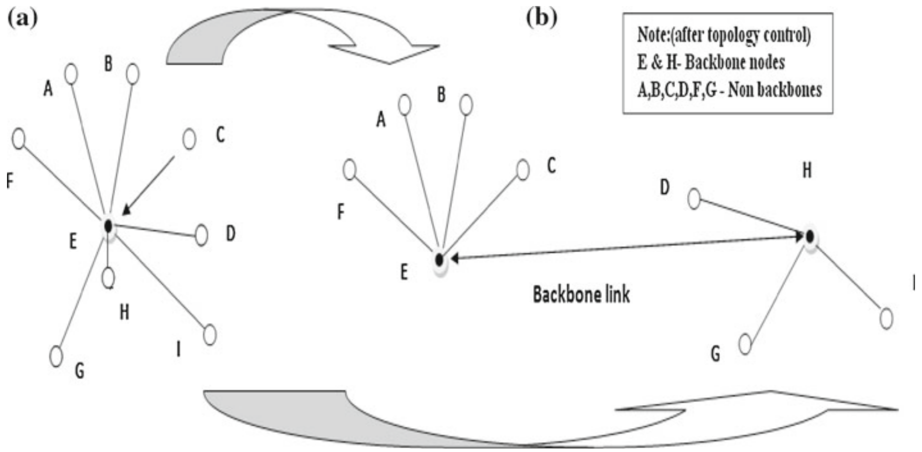

Fig. 5 **a** Seven connections per backbone, new node (C) request connection. **b** Connection request is shared between *E* (handle *A, B, C*, and *F*) and *H* (handle *G, D*, and *I*)

with seven connections is permitted, if it exceeds the limit, it will be reduced by nearby backbone. The backbone connections and their scenarios are illustrated in Fig. 5. In this figure, backbone node say E, attached with seven nodes named as A to I. Assume E is capable for routing only seven nodes, at that time a new node say C, wants to connect with E. It cannot be accommodated by E and hence the connections are divided as four and three for E and H respectively. The one hop neighbor H must be eligible backbone and its power level should be high compare to other nodes. The updation on connection balancing is done locally between the backbone and other nodes. In this way all local computation supports connection balancing routing in the network.

Topology mechanism is a key issue for large networks, which are dynamic in nature. Backbone based operation is one of the solution for topology control, because topology mechanism is required in the backbone nodes only, using the localized algorithm and local update to maintain the topology effectively, compared to the distributed algorithm.

6.2.1 The Need of Backbone with N-connection Maintenance:

The number of nodes added with the backbone cumulatively, the performance of backbone is degraded. To maintain the N-connections for each backbone, the eligible nodes are identified in earlier reconstruction stages. In this situation each node is assigned with N-connections and in our approach the number of nodes added simultaneously to a backbone and performance is evaluated based on constant bit rate traffic. Finally N = 3, 5 connections give the excellent network performance and N = 7 is accepted but the performance is not improved if N = 3 or 5 connections, the backbone with eligible nodes are identified easily and the alternate routing is done successfully. To achieve connection balanced alternate routing process, N-connection maintenance is implemented in each backbone node.

7 Self Organization

Self organization is mainly used to build a scalable network with huge number of subsystems. Wireless networks are bandwidth and energy constrained. Self organization minimizes the

Algorithm 3 for N- Connection maintenance:

1: INPUT: A undirected graph G = (V, E)
2: OUTPUT: An N- connected BN
3: Find BN, EN using algorithm 4.1(Backbone creation algorithm)
//Backbone periodically sends the number of connection information and power to all the one hop nodes.
4: chose node e ϵV such that e is maximum power
5: construct a spanning tree T_1 routed via ***b*** & ***e***.
//one hop nodes send their attachment to other backbone details to originating backbone.
6: construct a tree $T_1(b, e)$, a directed path to all other node in T and connection balanced as follows:
7: for i = 1 to n do
8: BNi is set of nodes in T_1
9: $V_i = b_i$U' e_i
10: for i = 1 to 7
11: // i.e 7 connections per backbone connections shared between BN and EB.
12: end for
13: end for
14: reserve all nodes in T to $T_1(b, e)$ for alternate route

number of message transmissions, receptions and helps energy conservation. It also support recover a CDS from a node failure. In our work, we would like to show how the low message complexity is used for energy saving in the node as well as the network and also it gives a solution to avoid the collision, contention and link failure issues. Self organization is achieved by the reconstructed, topology process of the backbone with the help of distributed localized algorithm. If the node wish to join any of the working area, its service is limited based on hierarchical way, in which the flat routing support for each node is achieved. In real time data transfer like voice over IP flat routing is not sufficient. Meanwhile hierarchical assign real time voice transfer for high priority and change the node to connect low traffic area. In the next section, the self organization scenario and the backbone selection process was briefly outlined. To reduce the routing and control overhead backbone nodes are permitted to handle other node traffic.

7.1 Self Organized Backbone Network

In the proposed method three level hierarchical structure is used to achieve the self organized backbone environment and it is based on the number of links attached with the backbones. The three levels are as follows: high traffic backbones with maximum connections, medium traffic with two or three links associated with each backbone, low traffic backbones with at least one connection per backbone. The final structure is equivalent for one hop backbone constructions with greedy algorithms. The formation of each level is summarized in Fig. 6. Backbone with high traffic consist of two backbones with four connections, medium traffic backbone has two connections. Traffic based backbone formation is given below:

- Backbone-high traffic: Routing by only one and if the backbone moves to new area eligible node perform the control traffic transfer. Stable backbones are only applicable in this type.
- Backbone medium traffic: Minimum two backbones are required to forward the request from the neighboring nodes. This level communicates with low and high traffic nodes.
- Backbone low traffic: one hop connections are required backbones and other nodes. In this case maintenance the backbone is easier during the movement compare to other two cases and more backbones are required for connectivity maintenance.

In the hierarchical architecture from the high traffic numbers of backbones are increased in each stage but the connections for each backbone reduced. This type of modeling support

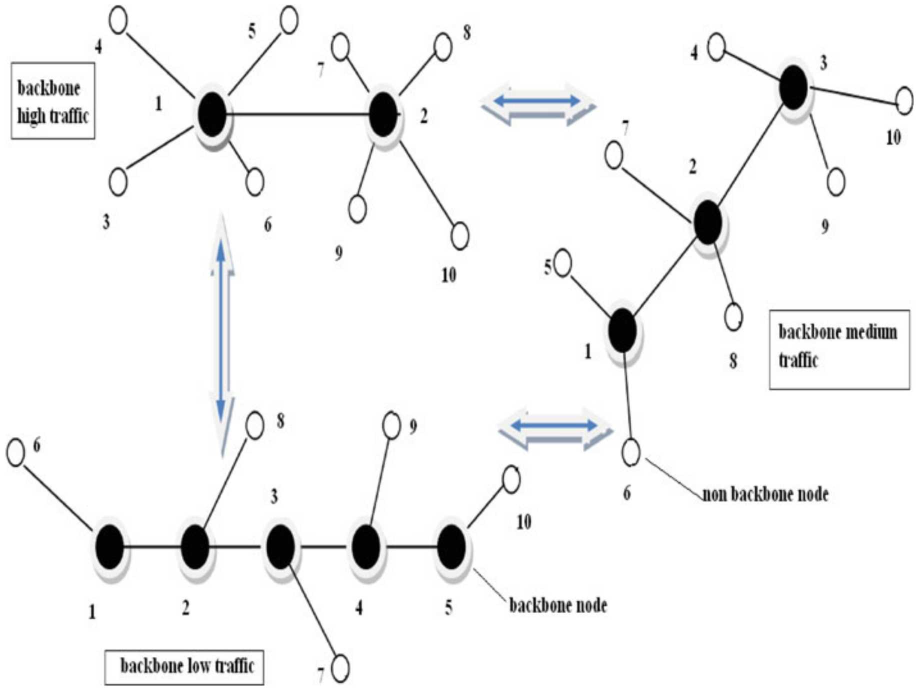

Fig. 6 Hierarchical backbones with various traffic conditions

Fig. 7 Routing packet fields

Backbone-ID	Destination-ID	Source-ID

self organization. The direct connections are not allowed between any two non backbone nodes and to maintain the connectivity, the nodes must be present in any one of the following scenario.

7.2 Selections of Backbones

Backbone selection process based on the hello packet exchanges between multiple mobile nodes. For high traffic nodes area of operation must be known by each device, and send hello request and wait for 10 ms. If it receives any response it know that some backbones are already in the operation area otherwise it will become a new backbone. Second level requires minimum two backbones based on the hello packet updates. It is similar to the high traffic hello packets, when the node receive two response from backbones it know that already sufficient backbones are in the area, otherwise it will become one of the backbone. Finally the low traffic condition simply applies the greedy procedure for backbone constructions and nodes at least connect one neighbor node based on the distance from the decision node. The various control packets and their fields for selection of backbone and updating the same during the mobility and it are displayed in Fig. 7. Normally high traffic conditions required minimum field packet compare to low and medium conditions because one node acts as a backbone.

Algorithm 4 for self organizations in backbone:

// Algorithm is executed by each backbone node b;
//let i be the integer such that $b \in B_i$; B_i - backbone set, b- backbone node
//selfish behavior – change b_i to b_{i+1}; or e_i to e_{i+1};
// where i to i + 1 be the backbone and eligible nodes with different speed;
// where B is the backbone set;
//b [i] be backbone with high power in B; // say B- node with low mobility and high coverage
Find all possible backbones in B with selfish node behavior;
if b = b (i) **then**
if 'i' odd **then** backbone support odd connection (minimum degree) **else if** 'i' is even
Connection adjusted to i + 1 backbone or eligible node.
Backbone b with $b_i < 5$ or $b_i < 7$ **then**
Become a part of B
end
else select e for connectivity with N-node //normal node with no restriction
end
else do not become part of e

Routing packets for inter and intra area has three fields namely backbone id, destination id, source id respectively.

7.2.1 Mobility Model for Maintain the Backbone with Minimum Speed

Mobility model is created based on one hop distance between all kinds of nodes. Each node in the network is operated in two modes: (1) Backbone node with slow speed (B-node), (2) Normal node (N-node), which is has no limitation for speed. In order to maintain the mobility between nodes, backbone node must have high transmission power and slow position changing speed. The low speed of backbone is not affected by the network performance because backbones are assigned with high transmission power (high coverage area), hence they able to make the one hop connection between nodes. Once a node is elected as a backbone it will switch from normal to slow speed node. The normal nodes move anywhere at any time but they must be maintain the one hop connectivity to any one backbone. If the node maintains the position it will send the control packet to the one hop nodes. The packet fields are transmission power, speed of other nodes and distance from the backbone node. The detailed information about the packet fields are given below;

(1) Transmission power: normally set as a high value (say 30 mW) for backbone node. (2) Speed: speed of the node is calculated by the position of the node in x and y co-ordinate system. (3) Node distance: If any node failure will occur in the dominating set then another one node will carry that data and it will sent the data to the destination through the dominators. The node will be selected depends on its distance. The distance will be calculated by the following formula, distance $D = N_f - D_r \cdot N_f$ is the failure node and the D_r is the dominator node. The formula used to calculate the speed of the node is given in Eq. (1).

$$\text{Speed S} = 1/\text{T} \sum_{\text{t}=0}^{\text{T}} (\text{X}_\text{i}, \text{Y}_\text{i}) + (\text{X}_{\text{i}+1}, \text{Y}_{\text{i}+1}) \tag{1}$$

where T is the current time, X_i and Y_i—current position in x and y coordinate system.

8 Performance Analysis

8.1 Performance Metrics

The proposed approach STAB-WIN, concentrate on the issues of self organization. It supports power saving for individual nodes as well as the entire network. The following parameters are used to analyze the performance of our work.

- Scalability: as the large number of nodes in wireless network increases, scalability imposes difficulties in transferring data. To overcome this issue, localized algorithm is proposed and initially 50 nodes are tested in the simulation. The updations are done by adding 50 nodes in each level, a total of 150 nodes are analyzed in the transient conditions for this purpose.
- Energy efficiency: It is measured as the number of routing packets transmitted per message packet as well as construction, reconstruction and topology formation packets used to achieve the self organization are also consider for analyzing the energy efficiency.
- System throughput: Measured as the total number data packets received at the destination during the simulation.
- End to end delay: Includes all possible delays caused by backbone construction process, topology based reconstruction and time taken for selfish node behavior during self organization is taken into account.

8.2 Simulation Results

All the simulations are conducted using ns-2 with 150 nodes initially, a $1250 \times 1250\,\mathrm{m}^2$ deployment area, 0.90 s hello interval and random way point model. We would like to observe the number of backbones versus network size, energy efficiency per nodes during self organized virtual backbone formation, throughput of the system compared with conventional routing protocols and system delay. The initial performance metrics, scalability of the proposed method was compared with Dai and Wu's method (Restricted Rule K) [9], and yang and Lin's method (D_1) [10], because these methods required minimum control overheads to maintain the backbone. Comparison is based on one hop neighbor information and the number of backbones required to achieve the self organized manner per network size is given in Fig. 8. The other methods like 2 and 3-hop connected dominating sets of yang and Lin's and Dai and Wu's method (Non-Restricted Rule K) are not compared here because it require more message overhead. Our approach is 30% efficiency than Lin cds approach and 20% efficiency than wu cds method.

In the simulation two types of STAB (Self Organized Topology Control Ability Backbone) are compared, there is (1) STAB (M)—stab with mobile nodes (2) STAB(S)—with stable nodes. The energy consumption analysis is based on number of broadcasting packets required to connect the nodes with the backbone and other nodes. In this regard broadcast costs of stab mobile and stable conditions are compared. It follows a commonly used energy model $\mathrm{e} = \alpha \mathrm{r}^k + \beta$, where e is the energy consumption is taken between 2 and 4, the device specific values α and β is assigned a value of 0.001 and 0 respectively. All the backbone nodes are permitted for different transmission power with slow mobility. The transmission power of each node is varied from 10 to 30 mW, normally the low power level assigned for eligible one hop backbone and maximum to backbone. For each simulation run, we experimented on different speed levels (5–40 m/s) by analyzing the network behavior for 200 s for STAB (M). The entire energy consumption performance in stable and during mobility is given in Fig. 9. From the result to construct a network with mobility model additional two levels of control

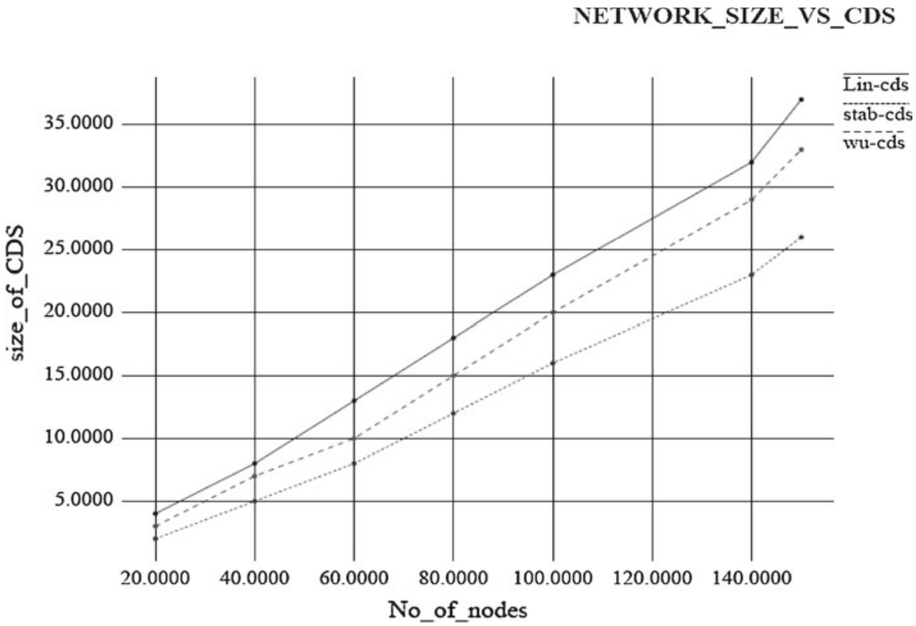

Fig. 8 Network size versus CDS (1 hop)

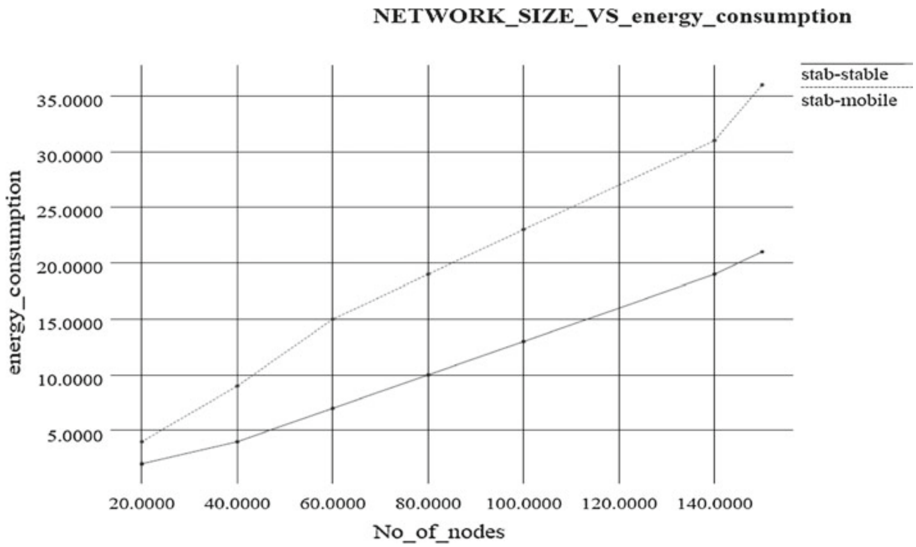

Fig. 9 Energy consumption during mobility

packets (for reconstruction and topology control) are required, in this reason only STAB (M) required higher energy consumption compare to the stab(s). The other QoS based result of STAB-WIN is compared with [12,13] conventional routing protocols AODV and DSR.

The energy efficiency performance includes the control load transfer and it is defined by the ratio of number of control packets transmitted along with the data packets. In other words, number of routing overheads as the function of packet rate. In this analysis the number of

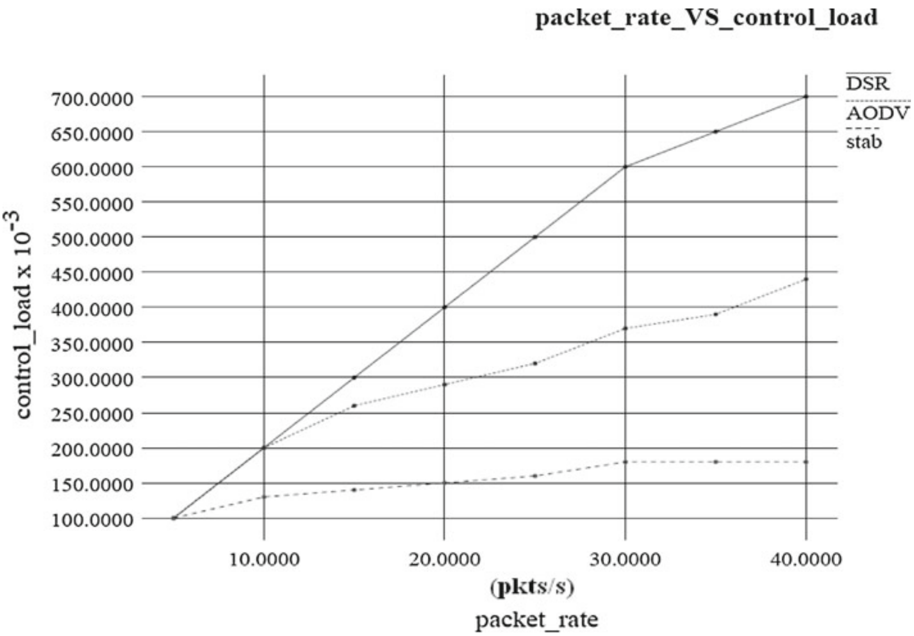

Fig. 10 Packet rate versus control field

control packets transmitted per data packet is analyzed and this should be shown in Fig. 10. The performance of routing overhead is compared with the conventional routing protocols like AODV and DSR. It is observed that, the packet rate increases the factor of 10 and the control required in the order of 0.5. Clearly, the system capacity increases depending on several factors including all local computations and updations. The control overhead rate is very low compared to the traditional routing methods.

The QoS performance in any routing protocol design is important, the performance metrics delay and throughput are analyzed in this paper. Two set of simulations are performed for this purpose, one is compare our result with [12,13] conventional routing protocols AODV and DSR. It follows the dynamic topology employing the random waypoint (RWP) model, 150 nodes with an initial power vary between 10 to 30 mW and the transmitting range 150 m are randomly positioned in 1250 m* 1250 m area. Figure 11 which show delay as a function of packet rate, in which the number of hops as metric to make routing decisions.

The 64 byte packet field is used for this purpose. The packets are sending specific data rate for backbone nodes and it will be differed for other normal nodes. Major delays caused by reconstruction process than topology process, in which the node movement and when it stops travelling are accounted to process the delay calculations. From the graph, delay performance is acceptable within the range of hierarchical routing protocols. A series of simulation experiments for the ad hoc network were conducted using the parameters outlined in the performance metrics. From Fig. 12, the delay and throughput curves which summaries the system performance and capacity. It also shows that STAB has better performance than traditional routing protocols. When the throughput is increased the delay of STAB-WIN is maintain for certain values (between1 to 1.5 s), and the performance is equal to the demand routing protocol AODV.

The simulation results summarized as follows: (1) the proposed approach support more number of self organized topology nodes than Dai and Wu's approaches, where they uses

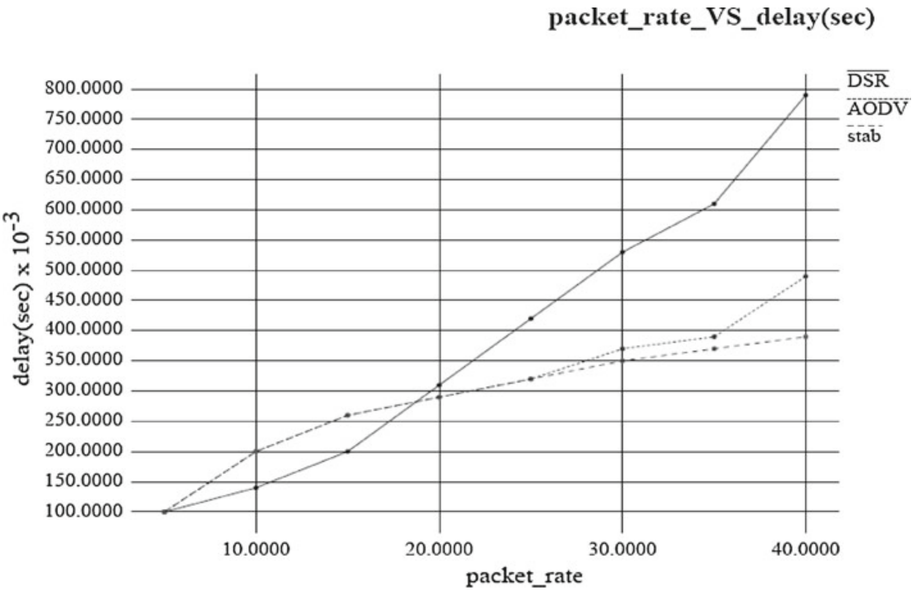

Fig. 11 Delay versus packet rate

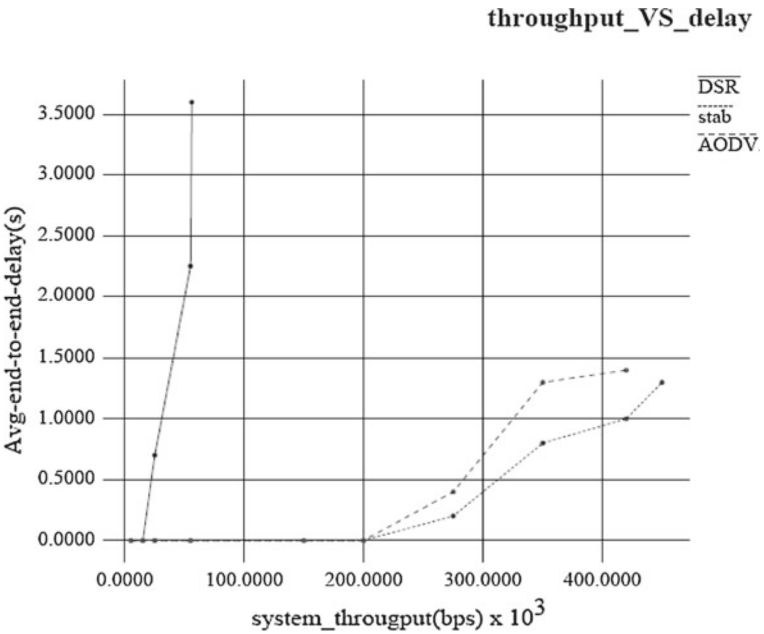

Fig. 12 Throughput under delay

only backbone nodes. (2) STAB-WIN produces efficient energy consumption in stable and mobile conditions. (3) Proposed approach has high throughput and low delay compare to the conventional routing methods.

9 Conclusion and Future Work

STAB-WIN provides all in one solution, because of this structure better performance is achieved in small working area. The main contribution of this paper is an in-depth analysis on the construction of the virtual backbone with self organization ability to support all the requirements in ad hoc networks. Since there is no fixed infrastructure most of the routing protocols mainly suffered from broadcast storm problem, to reduce the flooding virtual backbone is the optimal solution with few forwarding nodes. Although there exist a large number of algorithms to construct a backbone, there is no work on how to reconfigure, topology control and self organization by a single backbone node, in case of all nodes in movement. To overcome the existing issues we give the localized algorithm and update locally. The advantage of packet localized solution gives a clear picture about updating, reconfiguration, minimum degree maintenance and self organization with backbone during node movement, because of each node process the tiny packet efficiently. Various routing protocols have been proposed and studied in literature. Most of the existing backbone routing protocols are based on greedy algorithms and no performance guaranteed in real time. We define a model for a mobile ad hoc network and propose stable connection maintenance between nodes to support topology control. Even though the backbone sacrifices the speed, it will be tolerated by high coverage area (due to high transmission power). We show that our performance in terms of energy efficiency, throughput and delay. This paper does not cover few important aspects like multihop backbone construction and link bandwidth base routing. For future work, we would like to include these parameters and extend this approach to multihop wireless networks with fault tolerant routing.

Acknowledgments The authors would like to acknowledge the management of Karunya University, India for sponsored the initial stage of this work. Grant file no: KU/HR-2/4078/09.

References

1. Sinha, P., Sivakumar, R., & Bhargavan, V. (2001). Enhancing ad hoc routing with dynamic virtual infrastructures. In *Proceedings of IEEE INFOCOM '01* (pp. 1763–1772).
2. Tseng, Y. C., Ni, S. Y., Chen, Y. S., & Sheu, J. P (2002). The broadcast storm problem in a mobile ad hoc network. *Wireless Networks, 8*(2-3), 153–167.
3. Lee, S., Levin, D., Gopalakrishnan, V., & Bhattacharjee, B. (2007). Backbone construction in selfish wireless networks. In *Proceedings of ACM international conference on measurement and modeling of computer systems (Sigmetrics '07)* (pp. 121–132).
4. Blum, J., Ding, M., Thaeler, A., & Cheng, X. (2004). *Connected dominating set in sensor networks and MANETs* (pp. 329–369). Dordrecht: Kluwer.
5. Ruan, L., Du, H., Jia, X., Wu, W., Li, Y., & Ko, K. (2004). A greedy approximation for minimum connected dominating sets. *Theoretical Computer Science, 329*, 325–330.
6. Cheng, X., Ding, M., Du, D.H., & Jia, X. (2006). Virtual backbone construction in multi-hop ad hoc wireless networks. *Wireless Communications and Mobile Computing, 6*, 183–190.
7. Wan, P. J., Alzoubi, K. M., & Frieder, O. (2005). Distributed construction of connected dominating sets in wireless ad hoc networks. ACM Kluwer *Mobile Networks and Applications (MANET), 9*(2), 141–149.
8. Teymoori, P., & Yazdani, N. (2008). Local reconstruction of virtual backbone to support mobility in wireless ad hoc networks. In *IEEE international symposium on telecommunication* (pp. 382–387).
9. Dai, F., & Wu, J. (2004). An extended localized algorithm for connected dominating set formation in ad hoc wireless networks. *IEEE Transaction on Parallel and Distributed Systems, 15*(10), 908–920.
10. Yang, H. Y., Lin, C. H., & Tsai, M. J. (2008). Distributed algorithm for efficient construction and maintenance of connected K-hop dominating sets in mobile ad-hoc networks. *IEEE Transactions on Mobile Computing, 7*(4), 444–457.
11. Park, M., Wang, C., Wilson, J., Thai, M. T., Wu, W., & Farrago, A. (2007). A dominating and absorbent set in wireless ad-hoc networks with different transmission range. In *Proceedings of ACM MobiHoc* (pp. 22–31).

12. Hameed El-Afandi, H., hosseini, H., & Vairvan, K. (2006). A wireless ad hoc protocols comparison study: DSR, AODV and IWAR. *Journal of Computational Methods in Sciences and Engineering, 6*(5, 6 Suppl. 1), 181–190.
13. Yu, W., & Lee, J. (2002). DSR based energy-aware routing protocols in ad hoc networks. In *Proceedings of 2002 international conference on wireless networks*.
14. Bao, L., & Garcia-Luns Aceves, J. (2003). Topology management in ad hoc networks. In *Proceedings of ACM MobiHoc'03* (pp. 129–140).
15. Chen, B., Jamieson, K., Balakrishnan, H., & Morris, R. (2002). SPAN: An energy-efficient coordination algorithm for topology maintenance in ad hoc wireless networks. *ACM Wireless Networks Journal, 8*(5), 481–494.
16. wattenhofer, R., Li, L., Bahl, P., & Wang, Y. M. (2001). Distributed topology control for power efficient operation. In *Multihop wireless ad hoc networks. Infocom* (pp. 1388–1397).
17. Hedetniemi, S. M., Hedetniemi, S. T., Jacobs, D. P., & Srimani, P. K. (2003). Self-stabilizing algorithms for minimal dominating sets and maximal independent sets. *Computers and Mathematics with Applications, 46*(5–6), 805–811.
18. Wu, J., & Lou, W. (2003). Forward-node-set-based broadcast in clustered mobile ad hoc networks. *Wireless Networks and Mobile Computing, 3*(2), 155–173.
19. Yang, S., Wu, J., & Cao, J. (2005). Connected k-hop clustering in ad hoc networks. In *Proceedings of 34th international conference parallel processing (ICPP'05)* (pp. 373–380).
20. Smys, S., Bala, G. J., & Raj, J. S. (2009). Construction of virtual backbone to support mobility in MANET—A less overhead approach. In *3rd IEEE conference on application of information and communication technology, AICT 2009* (pp. 1–4). Baku, Azerbaijan.

Author Biographies

S. Smys received the Masters in Engineering degree from ANNA University, India. Currently he is a research scholar in Karunya University, Coimbatore, India. He is a member of IEEE. His research interests are self organization and topology control in mobile networks.

G. Josemin Bala received the Doctorate degree from the Anna University, India. Currently, she is a professor at Karunya University, Coimbatore, India. She is a member of IEEE. Her research interests include routing algorithms in mobile communication and complexity analysis in VLSI.