International Journal of Ethics in Engineering & Management Education
Website: www.ijeee.in (ISSN: 2348-4748, Volume 1, Issue 4, April 2014)

LINK FAILURE RECOVERY IN WIRELESS MESH NETWORKS USING ARS

Lubna kauser
P.G. Student, Dept. CSE
Appa institute of Engg & Tech, Gulbarga, Karanataka
lubna.kauser12@gmail.com

Syeda Asra
Assistant Professor , Dept. CSE
Appa institute of Engg & Tech, Gulbarga, Karanataka
asrascholar@gmail.com

Abstract--- **This paper describes the technique to recover from the failures that occurs in Wireless Mesh networks(WMN) like node failure, link failures etc , due to channel interference, dynamic obstacles or application bandwidth demands. This paper present an Autonomous network Reconfiguration System (ARS) that enables a multi-radio WMN to autonomously recover from local link failures to preserve network performance. ARS has been implemented and evaluated extensively on our IEEE 802.11-based WMN test-bed as well as through ns-2-based simulation . By using channel and radio diversities in WMNs, ARS generates necessary changes in local radio and channel assignments in order to recover from failures. Next ARS's on-line reconfigurability allows for real-time failure detection and network reconfiguration, thus improving channel-efficiency by 92%.. Our evaluation results demonstrated the effectiveness of ARS in recovering from local link-failures and in satisfying application's diverse QoS demands.**

Keywords: **wsn, ars**

1. INTRODUCTION:

A Wireless Mesh network (WMN) is dynamically self-organized and self-configured, with the nodes in the network automatically establishing and maintaining mesh connectivity among themselves . WMNs will deliver wireless services for a large variety of applications in personal, local, campus, and metropolitan areas networks . Still it is a challenging problem for preserving the required performance of such WMNs , due to heterogeneous and fluctuating wireless link conditions. For example, some links of a WMN may experience significant channel interference. Links in a certain area (e.g., a hospital or police station) might not be able to use some frequency channels because of spectrum etiquette or regulation.

To overcome from the wireless link failures many solutions has been proposed in WMNs, but still they have several limitations as follows. First, resource-allocation algorithms , even though their approach provides a comprehensive and optimal network configuration plan, they often require "global" configuration changes, which are undesirable in case of frequent local link failures. Second, a *greedy* channel-assignment algorithm can reduce the requirement of network changes by changing settings of only the faulty link(s),this greedy change might not be able to realize full improvements. Third, fault-tolerant routing protocols, i.e, local re-routing or multi-path routing , they rely on detour paths or redundant transmissions, which may require more network resources than link-level network reconfiguration.

We propose an *Autonomous network Reconfiguration System* (ARS) to overcome from above limitations, this technique allows a multi-radio WMN (mr-WMN) to autonomously reconfigure its local network settings for real-time recovery from link failures. ARS first searches for feasible local configuration changes available around a faulty area, based on current channel and radio associations. Then, by imposing current network settings as constraints, ARS identifies reconfiguration plans that require the minimum number of changes for the healthy network settings. Next, ARS also includes a monitoring protocol that enables a WMN to perform real-time failure recovery in conjunction with the planning algorithm. The accurate link-quality information from the monitoring protocol is used to identify network changes that satisfy applications' new QoS demands or that avoid propagation of QoS failures to neighboring links. Based on the measurement information, ARS detects link failures and/or generates QoS-aware network reconfiguration plans upon detection of a link failure. ARS can be implemented and evaluated extensively via experimentation on our multi-radio WMN test-bed as well as via ns2-based simulation.

First, ARS's planning algorithm effectively identifies reconfiguration plans that maximally satisfy the applications' QoS demands, accommodating twice more flows than static assignment. Next, ARS avoids the ripple effect via QoS-aware reconfiguration planning, unlike the greedy approach. Third, ARS's local reconfiguration improves network throughput and channel-efficiency by more than 26% and 92%, respectively, over the local re-routing scheme.

2. EXISTING SYSTEM:

First, resource-allocation algorithms can provide (theoretical) guidelines for initial network resource planning. However, even though their approach provides a comprehensive and optimal network configuration plan, they often require "global" configuration changes, which are undesirable in case of frequent local link failures. Next, a

International Journal of Ethics in Engineering & Management Education
Website: www.ijeee.in (ISSN: 2348-4748, Volume 1, Issue 4, April 2014)

greedy channel-assignment algorithm can reduce the requirement of network changes by changing settings of only the faulty link(s). However, this greedy change might not be able to realize full improvements, which can only be achieved by considering configurations of neighboring mesh routers in addition to the faulty link(s). Third, fault-tolerant routing protocols, such as local re-routing or multi-path routing, can be adopted to use network-level path diversity for avoiding the faulty links.

2.1 Localized reconfiguration: Network reconfiguration needs a planning algorithm that keeps necessary network changes (to recover from link failures) as local as possible, as opposed to changing the entire network settings. Even though these algorithms are suitable for static or periodic network planning, they may cause network service disruption and thus are unsuitable for dynamic network reconfiguration that has to deal with frequent local link failures. Next, the *greedy* channel-assignment algorithm, which considers only local areas in channel assignments might do better in reducing the scope of network changes. Finally, interference-aware channel-assignment algorithms can minimize interference by assigning orthogonal channels as closely as possible geographically. For example, in Fig. 2, if channel 5 is lightly-loaded in a faulty area, the second radio of node *C* can re-associate itself with the first radio of node *I*, avoiding configuration changes of other links.

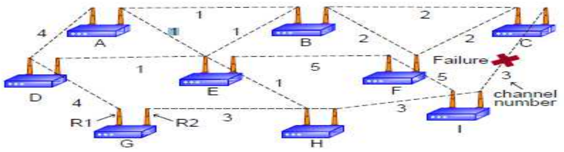


Fig. 1. Multi-radio WMN:

A WMN has an initial assignment of frequency channels as shown above. The network often experiences wireless link failure and needs to reconfigure its settings. While this approach can improve overall network capacity by using additional channels, the algorithm could further improve its flexibility by considering both radio diversity (i.e., link association) and local traffic information.

2.2 QoS-Constraints: Reconfiguration has to satisfy QoS constraints on each link as much as possible. First, given each link's bandwidth constraints, existing channel-assignment and scheduling algorithms can provide approximately optimal network configurations. However, these algorithms may require global network configuration changes from changing local QoS demands, thus causing network disruptions. We need instead a reconfiguration algorithm that incurs only local changes while maximizing the chance of meeting the QoS demands. For example, if link EH in Fig.1 experiences a QoS failure on channel 1, then one simple reconfiguration plan would be to re-associate R1 of node H to R2 of node E in channel 5, which has enough bandwidth. The greedy algorithm might be able to satisfy particular links' QoS demands by replacing a faulty channel with a new channel.

2.3 Cross-layer interaction: Network reconfiguration has to jointly consider network settings across multiple layers. In fault-tolerant routing protocols, such as local re-routing or multi-path routing , allow for flow reconfiguration to meet the QoS constraints by exploiting path diversity, they consume more network resources than link reconfiguration, because of their reliance on detour paths or redundant transmissions.

2.4 Limitations of Existing System:

1. Cannot avoid propagation of QoS failures to neighboring links.
2. Unsuitable for dynamic network reconfiguration.

3. PROPOSED SYSTEM:

To overcome the above limitations, we propose an *Autonomous Network Reconfiguration System* (ARS) that allows a multi-radio WMN to autonomously reconfigure its local network settings—channel, radio, and route assignment—for real-time recovery from link failures. In its core, ARS is equipped with a reconfiguration planning algorithm that identifies local configuration changes for the recovery, while minimizing changes of healthy network settings. Briefly, ARS first searches for feasible local configuration changes available around a faulty area, based on current channel and radio associations. Then, by imposing current network settings as constraints, ARS identifies reconfiguration plans that require the minimum number of changes for the healthy network settings. It detects a long-term (lasting for weeks or months) failures, network-wide planning algorithms can be used. Note that hardware failures or broadband-channel failures .

3.1 The ARS Architecture:

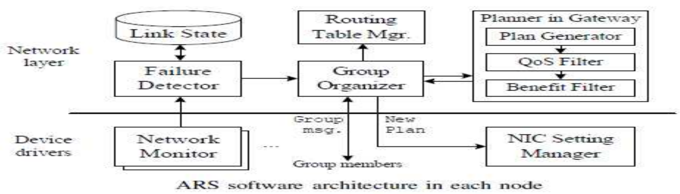


Fig 2: ARS is implemented across network and link layers as a loadable module of Linux 2.6 kernel.

The above figure shows the software architecture of ARS. First, ARS in the network layer is implemented using netfilter , which provides ARS with a hook to capture and send ARS-related packets such as group-formation messages. In addition, this module includes several important algorithms and protocols of ARS: (i) *network planner*, which generates reconfiguration plans only in a gateway node; (ii) *group organizer*, which forms a local group among mesh routers;

International Journal of Ethics in Engineering & Management Education
Website: www.ijeee.in (ISSN: 2348-4748, Volume 1, Issue 4, April 2014)

(iii) *failure detector*, which periodically interacts with a network monitor in the device driver and maintains an up-to-date link-state table; and (iv) *routing table manager*, through which ARS obtains or updates states of a system routing table. Next, ARS components in the device driver are implemented in an open source MADWiFi device driver . This driver is designed for Atheros chipset-based 802.11 NICs and allows for accessing various control and management registers (e.g., longretry, txrate) in the MAC layer, making network monitoring accurate. The module in this driver includes
(i) *network monitor*, which efficiently monitors link-quality and is extensible to support as many multiple radios as possible ; and (ii) *NIC manager*, which effectively reconfigures NIC's settings based on a reconfiguration plan from the group organizer.

3.2 Multi-radio WMN (mr-WMNs):

A network is assumed to consist of mesh nodes, IEEE 802.11-based wireless links, and one control gateway. Each mesh node is equipped with n radios, and each radio's channel and link assignments are initially made by using global channel/link assignment algorithms. ARS is a distributed system that is easily deployable in IEEE802.11-based mr-WMNs.

ARS self-reconfigurability has following distinct features:

• *Localized reconfiguration*: On multiple channels and radio associations , ARS generates reconfiguration plans that allow for changes of network configurations only in the vicinity where link failures occurred, while retaining configurations in areas remote from failure locations.

• *QoS-aware planning*: QoSsatisfiable reconfiguration plans ARS identifies by (i) estimating the QoSsatisfiability of generated reconfiguration plans and (ii) deriving their expected benefits in channel utilization.

• *Autonomous reconfiguration via link-quality monitoring*: ARS accurately monitors the quality4 of links of each node in a distributed manner. Furthermore, based on the measurements and given links' QoS constraints, ARS detects local link failures and autonomously initiates network reconfiguration.

• *Cross-layer interaction*: ARS actively interacts across the network and link layers for planning. This interaction enables ARS to include a re-routing for reconfiguration planning in addition to link-layer reconfiguration. ARS can also maintain connectivity during recovery period with the help of a routing protocol.

Algorithm.1 ARS Operation at Mesh node i

(1) Monitoring period (t_m)

1: for every link j do
2: measure link-quality (lq) using passive monitoring;
3: end for
4: send monitoring results to a gateway g;

(2) Failure detection and group formation period (t_f)

5: if link l violates link requirements r then
6: request a group formation on channel c of link l;
7: end if
8: participate in a leader election if a request is received;

(3) planning period (M , t_p)

9: if node i is elected as a leader then
10: send a planning request message (c, M) to a gateway;
11: else if node i is a gateway then
12: synchronize requests from reconfiguration groups M_n
13: generate a reconfiguration plan (p) for M_i;
14: send a reconfiguration plan p to a leader of M_i;
15: end if

(4) Reconfiguration period (p , t_r)

16: if p includes changes of node i then
17: apply the changes to links at t;
18: end if
19: relay p to neighbouring members, if any

ARS operation has define in above algorithm as follows:
First, ARS in every mesh node monitors the quality of its outgoing wireless links at every tm (e.g., 10 sec) and reports the results to a gateway via a management message. Second, once it detects a link failure(s), ARS in the detector node(s) triggers the formation of a group among local mesh routers that use a faulty channel, and one of the group members is elected as a leader using the well-known bully algorithm, for coordinating the reconfiguration.
Third, the leader node sends a planning-request message to a gateway. Then, the gateway synchronizes the planning request and generates a reconfiguration plan for the request. Fourth, the gateway sends a reconfiguration plan to the leader node and the group members. Finally, all nodes in the group execute the corresponding configuration changes, if any, and resolve the group.

3.3 Localized Network Reconfiguration:

The ARS function is to *systematically* generate localized reconfiguration plans. A *reconfiguration plan* is defined as a set of links' configuration changes e.g., channel switch, link association necessary for a network to recover from a link(s) failure on a channel and there are usually multiple reconfiguration plans for each link failure. By contrast, ARS systematically generates reconfiguration plans that localize network changes by dividing the reconfiguration planning into three processes—feasibility, QoS-satisfiability, and optimality—and applying different levels of constraints.

Fig. 3. Localized reconfiguration planning in ARS: ARS generates a reconfiguration plan by breaking down the planning process into three processes with different constraints.

As depicted in Figure 3, ARS first applies connectivity constraints to generate a *set* of feasible reconfiguration plans that enumerate feasible channel, link, and route changes

International Journal of Ethics in Engineering & Management Education

Website: www.ijeee.in (ISSN: 2348-4748, Volume 1, Issue 4, April 2014)

around the faulty areas, given connectivity and link-failure constraints., ARS then applies within set strict constraints (i.e., QoS and network utilization) to identify a reconfiguration plan that satisfies the QoS demands and that improves network utilization most.

3.3.1 Generating feasible plans : Given multiple radios, channels, and routes, ARS identifies feasible changes that help avoid a local link failure but maintain existing network connectivity as much as possible. However, in generating such plans, ARS has to address the following challenges.:

• *Avoiding a faulty channel*: First ARS must has to ensure that the faulty link needs to be fixed via reconfiguration. ARS considers three primitive link changes, as explained in Table I. Specifically, to fix a faulty link(s), ARS can use (i) a channel-switch (S) where both end-radios of link AB can simultaneously change their tuned channel, (ii) a radioswitch (R) where one radio in node A can switch its channel and associate with another radio in node B, and (iii) a routeswitch (D) where all traffic over the faulty link can use a detour path, instead of the faulty link.

TABLE I

Definition of link-change in ARS. EACH CHANGE REPRESENTS A PRIMITIVE LINK CHANGE IN CHANNEL, ASSOCIATION, OR ROUTE. MULTIPLE CHANGES CAN BE JOINTLY USED TO REPRESENT CHANGES OF MULTIPLE LINKS.

Primitive changes	Description
Channel switch $(S(A_i, B_j)_{\alpha \to \beta})$	Radios A_i and B_j of link AB switch their channel (α) to other channel (β).
Radio switch $(R(A_i, B_j)_{\alpha \to \beta})$	Radio A_i in node A re-associates with radio B_j in node B, tuned in channel (β).
Detouring $(D(A_i, B_j))$	Both radios A_i and B_j of link AB remove their associations and use a detour path, if exists.

• *Maintaining network connectivity and utilization*: To avoiding the use of the faulty channel, ARS needs to maintain connectivity with the full utilization of radio resources. Because each radio can associate itself with multiple neighboring nodes, a change in one link triggers other neighboring links to change their settings. ARS takes a *two-step* approach. ARS first generates feasible changes of each link using the primitives, and then combines a set of feasible changes that enable a network to maintain its own connectivity. ARS maximizes the usage of network resources by making each radio of a mesh node associate itself with at least one link and by avoiding the use of same (*redundant*) channel among radios in one node.

• *Controlling the scope of reconfiguration changes*: ARS must to avoid network changes as *local* as possible, but at the same time it needs to find a locally optimal solution by considering more network changes or scope. To make this tradeoff, ARS uses a k-hop reconfiguration parameter. Starting from a faulty link(s), ARS considers link changes within the first k hops and generates feasible plans. If ARS cannot find a local solution, it increases the number of hops (k) so that ARS may explore a broad range of link changes. Thus, the total number of reconfiguration changes is determined on the basis of existing configurations around the faulty area as well as the value of k.

3.3.2 QoS-Satisfiability Evaluation: From a set of feasible plans F, ARS now needs to identify QoS- satisfying reconfiguration plans by checking if the QoS constraints are met under each plan. Although each feasible plan ensures that a faulty link(s) will use non-faulty channels and maintain its connectivity, some plans might not satisfy the QoS constraints or even cause cascaded QoS failures on neighboring links. To filter out such plans, ARS has to solve the following challenges.

• *Per-link Bandwidth Estimation*: ARS has to check whether each link's configuration change satisfies its bandwidth requirement, so it must estimate link bandwidth. To estimate link bandwidth, ARS accurately measures each link's capacity and its available channel air-time. Even though numerous bandwidth-estimation techniques have been proposed, they focus on the average bandwidth of each node in a network or the end-to-end throughput of flows , which cannot be used to calculate the impact of per-link configuration changes. By contrast, ARS estimates an individual link's capacity (C), based on measured (or cached) link-quality information—packet-delivery ratio and data-transmission rate measured by passively monitoring the transmissions of data or probing packets —and the formula back-off time (tb) and actual transmission time (ts).

$$t_b \text{ is } \frac{CWMin \times slotTime}{2}$$

Here, we assume that ARS is assumed to cache link-quality information for other channels and use the cached information to generate reconfiguration plans. If the information becomes obsolete, ARS detects link failures and triggers another reconfiguration to find QoS-satisfiable plans—lazy monitoring.

• *Examining per-link bandwidth satisfiability*: With Given measured bandwidth and bandwidth requirements, ARS has to check if the new link change(s) satisfies QoS requirements. ARS defines and uses the expected busy air-time ratio of each link to check the link's QoS satisfiability.

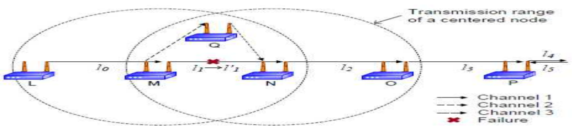


Fig. 4. *Busy Air-time Ratio (BAR) of a directed link*: BAR (e.g., l1) is affected by activities of neighboring links (l0, l2,and l3) in channel 1 and is used to evaluate QoS satisfiability of a link.

By taking that a link's bandwidth requirement (q) is given, the link's busy air-time ratio (BAR) can be defined as BAR = $\frac{q}{c}$ and must not exceed 1.0 (i.e., BAR < 1.0) for a link to satisfy its bandwidth requirement. If multiple links share the air-time of one channel, ARS calculates aggregate BAR (*aBAR*) of end-radios of a link, which is defined as

$$\text{aBAR(k)}=\sum l \in L(k) \frac{ql}{cl} ,$$

where k is a radio ID, l a link associated with radio k, L(k) the set of directed links within and across radio k's transmission range.

IJEEE
International Journal of Ethics in Engineering & Management Education
Website: www.ijeee.in (ISSN: 2348-4748, Volume 1, Issue 4, April 2014)

• *Avoiding cascaded link failures*: ARS needs to check whether neighboring links are affected by local changes (i.e., cascaded link failures). To identify such adverse effect from a plan, ARS also estimates the QoS-satisfiability of links one hop away from member nodes whose links' capacity can be affected by the plan. If one-hop-away links still meet the QoS requirement, the effects of the changes do not propagate thanks to spatial reuse of channels. Otherwise, the effects of local changes will propagate, causing cascaded QoS failures.

3.3.3 Choosing the best plan: ARS now has a set of reconfiguration plans that are QoS-satisfiable, and needs to choose a plan within the set for a local network to have evenly distributed link capacity. to support fair share into the planning, ARS needs to address the following challenges.

• *Quantifying the fairness of a plan*: ARS defines and uses a benefit function B(p) that quantifies the improvement of channel utilization that the reconfiguration plan p makes. Specifically, the benefit function is defined as

$$B(p)=\frac{1}{n}\sum_{k=1}^{n}\beta(k),$$

where β(k) is the relative improvement in the air-time usage of radio k, and n the number of radios whose β(k) has changed from the plan. Here, β(k) is considered as a fairness index on the usage of channel air-time, and it is defined as follows:

$$\beta(k)=\begin{cases}\epsilon1(k)-\epsilon2(k) & if\ \epsilon1(k),\epsilon2(k)>\delta\\ \epsilon2(k)-\epsilon1(k) & if\ \epsilon1(k),\epsilon2(k)<\delta\\ \epsilon1(k)+\epsilon2(k)-2\delta & if\ \epsilon1(k)>\delta>\epsilon2(k)\\ 2\delta-\epsilon1(k)-\epsilon2(k) & if\ \epsilon2(k)>\delta>\epsilon1(k)\end{cases}$$

where $\epsilon1(k)$ and $\epsilon2(k)$ are estimated *aBAR*s of a radio k in existing configurations and in new configurations, respectively, and δ the desired channel utilization.

• *Breaking a tie among multiple plans*: ARS needs to break a tie among them. ARS uses the number of link changes that each plan requires to break a tie. Although link configuration changes incur a small amount of flow disruption (e.g., in the order of 10 ms), the less changes in link configuration, the less network disruption.

3.4 Complexity of ARS:

ARS incurs reasonable bandwidth and computation overheads. First, the network monitoring part in the reconfiguration protocols is made highly efficient by exploiting existing data traffic and consumes less than 12 Kbps probing bandwidth (i.e., 1 packet per second) for each radio. First the group formation requires only O(n) message overhead (in forming a spanning tree), where n is the number of nodes in the group. Second, the computational overhead in ARS mainly stems from the planning algorithms. Specifically, generating its possible link plans incurs O(n+m) complexity, where n is the number of available channels and m the number of radios. Finally , a gateway node needs to generate and evaluate feasible plans, which incurs search overhead in a constraint graph that consists of O(l(n+m)) nodes, where l is the number of links that use a faulty channel in the group.

3.5 Advantages of proposed system:

1. Public safety, environment monitoring and city-wide wireless Internet services.
2. Avoid propagation of QoS failures to neighboring links(or 'ripple effects').

4. PERFORMANCE EVALUATION:

Implementation of *Au-tonomous network Reconfiguration System* (ARS) that allows a multi-radio WMN (mr-WMN) in the real world is quite hard.Hence, the preferred alternative is to use some simulation software which can mimic real-life scenarios. Though it is difficult to reproduce all the real life factors, most of the characteristics can be programmed into the scenario.

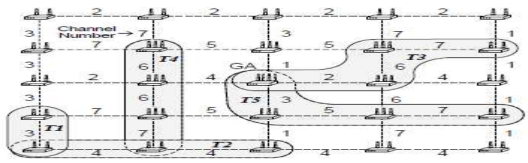
4.1 Methodology

The network in ARS(mr-WMN) runs routing protocols such as WCETT or ETX to determine the path of the admitted flows. This routing protocol is also assumed to include route discovery and recovery algorithm that can be used for maintaining alternative paths even in the presence of link failures, it is best to use identical simulation environments for their performance evaluation.

4.1.1 Simulation Environment:

Fig 5: ARS hardware prototype, ARS software is then installed in Soekris wireless routers and evaluated extensively in our multi-radio WMN testbed.

We have implemented ARS in a Linux OS and evaluated it in our testbed



(a) Scenarios for diverse QoS requests

Table-1: δ = 0.8

Time	T1		T2		T3		T4		T5	
	(i)	(ii)	(i)	(ii)	(i)	(ii)	(i)	(ii)	(i)	(ii)
Admission	no	yes	no	yes	no	yes	no	yes	no	yes
Available capacity(Mb/s)	0.5	3.5	0.5	2.0	1.0	3.0	3.0	7.5	0.5	1.5

Table-2: δ = 0.4

Time	T1		T2		T3		T4		T5	
	(i)	(ii)	(i)	(ii)	(i)	(ii)	(i)	(ii)	(i)	(ii)
Admission	no	yes	no	yes	no	yes	no	yes	no	yes
Available capacity(Mb/s)	0.5	5.5	0.5	5.5	1.0	3.0	0.5	1.5	0.5	1.5

(b) QoS benefits from reconfigurations

Fig. 6. *Satisfying varying QoS constraints*: (a) shows requests with different QoS requirements. Next, (b) shows improved (or changed) network capability (i) before and (ii) after reconfiguration

International Journal of Ethics in Engineering & Management Education
Website: www.ijeee.in (ISSN: 2348-4748, Volume 1, Issue 4, April 2014)

Ns-2 is used in our simulation study. Throughout the simulation, we use a grid topology with 25 nodes in an area of 1Km×1Km, as shown in Fig. 4(a). In the topology, adjacent nodes are separated by 180m and each node is equipped with a different number of radios, depending on its proximity to a gateway. For each node in the topology, we use the following network protocol stacks. First, the shadowing propagation model [39] is used to simulate varying channel quality and multi-path effects. Second , CMU 802.11 wireless extension is used for the MAC protocol with a fixed data rate (i.e., 11 Mbps) and is further modified to support multiple radios and multiple channels.

Finally, a link-state routing protocol, a modification of DSDV, and multi-radio-aware routing metric(WCETT) are implemented and used for routing. There are several settings to emulate real-network activities. First, to generate users' traffic, multiple UDP flows between a gateway and randomly-chosen mesh nodes are introduced. Each flow runs at 500 Kbps with a packet size of 1000 bytes. Second, to create network failures, uniformly-distributed channel faults are injected at a random time point. Random bit-error is used to emulate channel-related link failures and lasts for a given failure period. Finally, all experiments are run for 3000 seconds, and the results of 10 runs are averaged unless specified otherwise.

4.2Module Descriptions:

1. Multi-radio WMN: A network is assumed to consist of mesh nodes, IEEE 802.11-based wireless links, and one control gateway. Each mesh node is equipped with n radios, and each radio's channel and link assignments are initially made by using global channel/link assignment algorithms**.**

2. Link-Failure Detection:ARS in every mesh node monitors the quality of its outgoing wireless links at every tm (e.g., 10 sec) and reports the results to a gateway via a management message. Second, once it detects a link failure(s), ARS in the detector node(s) triggers the formation of a group among local mesh routers that use a faulty channel, and one of the group members is elected as a leader and coordinating the reconfiguration.

3. Leader Node: The leader node sends a planning-request message to a gateway. If any link is failure group members send request to the particular leader after that the leader node send request to the gateway.

4. Network Planner: It generates reconfiguration plans only in a gateway node. Network planner plans the diversity path for avoiding the faulty links. Then, the gateway synchronizes the planning requests—if there are multiples requests—and generates a reconfiguration plan for the request. Fourth, the gateway sends a reconfiguration plan to the leader node and the group members. Finally, all nodes in the group execute the corresponding configuration changes, if any, and resolve the group.

The following image shows the ARS(mr-WMN) environment with the model design:

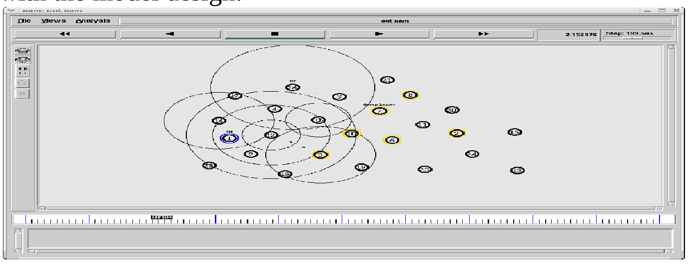


Image. ARS(mr-WSN)model design in Ns-2

5. RESULTS AND ANALYSIS:

Evaluation Results:

*1) Effectiveness of QoS-aware planning:*We measured the effectiveness of ARS in meeting the varying QoS requirements in a mr-WMN. We initially assign symmetric link capacity as shown in the channel assignment of the grid topology (fig 6(a)). Then, while changing the QoS constraints in gray areas at different times (i.e., T1, . . . , T5), we evaluate the improvement of available capacity that ARS can generate via reconfiguration. the tables of Fig. 6(b), ARS reconfigures a wireless mesh network to meet different QoS requirements. Before each reconfiguration, the gray areas can only accept 1 to 9 UDP flows. On the other hand, after reconfiguration, the network in the areas can admit 4 to 15 additional flows, improving the average network capacity of the gray areas by 3.5 times.

2) Importance of the benefit function: the impact of the benefit function on the ARS's planning algorithm. We conducted the same experiment as the previous one with different values of δ in the benefit function. As shown in Fig. 6(b), a high value (0.8) of δ allows ARS to keep local channel-efficiency high. By contrast, a low value (0.4) can deliver more available bandwidth (on average, 1.2 Mbps) than when the high value is used, since ARS tries to reserve more
capacity.

3) Impact of the reconfiguration range:

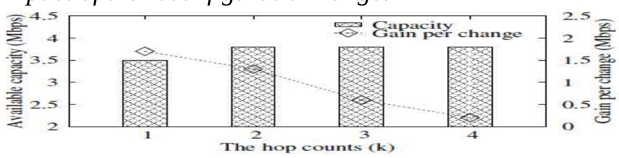
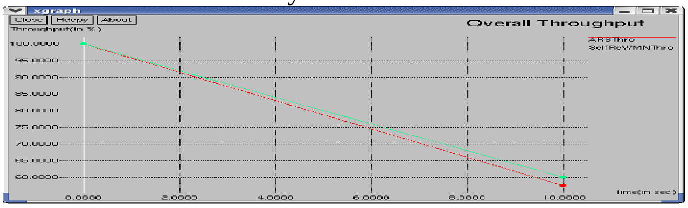


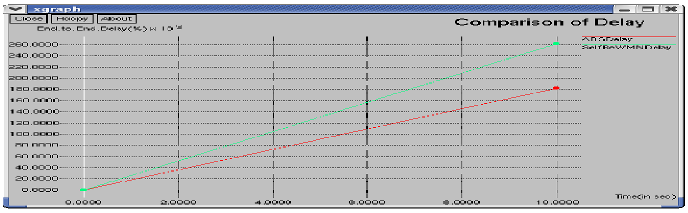
Fig.7. The impact of reconfiguration range:

The hop length can help ARS search for reconfiguration plans. However, the benefit from the increased length is small, whereas the number of total changes for the reconfiguration adversely increases. As shown in the figure, ARS can improve the available links' capacity by increasing the reconfiguration range. However, its improvement becomes marginal as the range increases. In other words, the improvement is essentially bounded by the total capacity of physical radios. Furthermore,

International Journal of Ethics in Engineering & Management Education
Website: www.ijeee.in (ISSN: 2348-4748, Volume 1, Issue 4, April 2014)

because reconfiguration plans with a larger range are required to incur more changes in network settings, the bandwidth gain per change significantly degrades . We also observed the similar results in other reconfiguration requests (T2,T3,T4), but omitted them for brevity.





***Images** .show the overall increase in throughput performance & decrease in delay of proposed system*

6. CONCLUSION:

This paper presented an Autonomous network Reconfiguration System (ARS) that enables a mr- WMN to autonomously recover from wireless link failures. ARS effectively identifies reconfiguration plans that satisfy applications' QoS constraints, admitting up to two times more flows than static assignment, through QoS-aware planning. Next, ARS's on-line reconfigurability allows for real-time failure detection and network reconfiguration, thus improving channel-efficiency by 92%. The effectiveness of ARS in recovering from local link-failures and in satisfying applications' diverse QoS demands are implemented in Linux-based implementation and *ns-2*-based simulation.

7. FUTURE WORK:

Joint optimization with flow assignment and routing: *Use of ARS in IEEE 802.11b/g WMNs*: ARS is mainly evaluated in IEEE 802.11a networks, where 13 orthogonal channels are available. However, ARS can also be effective in a network with a small number of orthogonal channels (e.g., 3 in IEEE 802.11b/g). Because ARS includes a link association primitive, it can learn available channel capacity by associating with idle interfaces of neighboring nodes, and it further limits the range of a reconfiguration group. Even though its design goal is to recover from network failures as a best-effort service, ARS is the first step to solve this optimization problem.

REFERENCES:

[1]. I. Akyildiz, X. Wang, and W. Wang, "Wireless mesh networks: A survey," *Computer Networks*, no. 47, pp. 445–487, 2005.
[2]. MIT Roofnet, http://www.pdos.lcs.mit.edu/roofnet.
[3]. Motorola Inc. Mesh Broadband, http://www.motorola.com/mesh.
[4]. P. Kyasanur and N. Vaidya, "Capacity of multi-channel wireless networks: Impact of number of channels and interfaces," in *Proceedings of ACM MobiCom*, Cologne, Germany, Aug. 2005.
[5]. K. Ramanchandran, E. Belding-Royer, and M. Buddhikot, "Interferenceaware channel assignment in multi-radio wireless mesh networks," in *Proceedings of IEEE InfoCom*, Barcelona, Spain, Apr. 2006.
[6]. R. Draves, J. Padhye, and B. Zill, "Routing in multi-radio, multi-hop wireless mesh networks," in *Proceedings of ACM MobiCom*, Philadelphia, PA, Sept. 2004.
[7]. P. Bahl, R. Chandra, and J. Dunagan, "SSCH: Slotted seeded channel hopping for capacity improvement in IEEE 802.11 ad-hoc wireless networks," in *Proceedings of ACM MobiCom*, Philadelphia, PA, Sept. 2004.
[8]. D. Aguayo, J. Bicket, S. Biswas, G. Judd, and R. Morris, "Link-level measurements from an 802.11b mesh network," in *Proceedings of ACM SIGCOMM*, Portland, OR, Aug. 2004.
[9]. A. Akella, G. Judd, S. Seshan, and P. Steenkiste, "Self-management in chaotic wireless deployments," in *Proceedings of ACM MobiCom*, Cologne, Germany, Sept. 2005.
[10]. J. Zhao, H. Zheng, and G.-H. Yang, "Distributed coordination in dynamic spectrum allocation networks," in *Proceedings of IEEE DySPAN*, Baltimore, MD, Nov. 2005.
[11]. M. J. Marcus, "Real time spectrum markets and interruptible spectrum: New concepts of spectrum use enabled by cognitive radio," in *Proceedings of IEEE DySPAN*, Baltimore, MD, Nov. 2005.