International Journal on Applications in Information and Communication Engineering
Volume 5 : Issue 2: December 2019, pp 10 – 18 www.aetsjournal.com ISSN (Online) : 2394 - 6237

DIVERSITY CHANNEL ALLOCATION AND ROUTING FOR WIRELESS MESH NETWORKS

J.JEEVAREKHA , E.HEMALATHA , R.GOWRISHANKAR

Abstract— Multi-radio mesh refers to a unique pair of dedicated radios on each end of the link. This means there is a unique frequency used for each wireless hop, where you can achieve maximum performance without bandwidth degradation. But, multi-hop wireless mesh networks (WMNs) experience frequent link failures caused by channel interference, dynamic obstacles, and/or applications' bandwidth demands. These failures cause severe performance degradation in WMNs or require expensive manual network management for their real-time recovery. This paper presents an Autonomous Link Recovery System (ALR) that makes a Multi-radio WMN autonomously reconfigure its local network settings channel, radio, and route assignment for real-time recovery from link failures and also used dynamics channel allocation routing protocol it provides the change over the channels very quickly. ALR has been implemented on ns2 based simulation and its result evaluation shows an improvement of 90% channel efficiency.

Index Terms— IEEE 802.11, Multi-radio wireless mesh networks (mr-WMNs), link failures.

I. INTRODUCTION

Wireless mesh network (WMN) is a communications network made up of radio nodes organized in a mesh topology. Wireless mesh networks often consist of mesh clients, mesh routers and gateways. The mesh clients are often laptops, cell phones and other wireless devices while the mesh routers forward traffic to and from the gateways. (WMNs) are being developed actively and deployed widely for a variety of applications, such as public safety, environment monitoring, and citywide wireless Internet services [1]–[3]. They have also been evolving in various forms (e.g., using multiradio/channel systems [4]–[7]) to meet the increasing capacity demands by the above-mentioned and other emerging applications. However, due to heterogeneous and fluctuating wireless link conditions [8]–[10], preserving the required performance of such WMNs is still a challenging problem.

Even though many solutions for WMNs to recover from wireless link failures have been proposed, they still have several limitations as follows. First, resource-allocation algorithms [12]–[14] 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 *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 rerouting [16] or multipath routing [17], can be adopted to use network-level path diversity for avoiding the faulty links. However, they rely on detour paths or redundant transmissions, which may require more network resources than link-level network reconfiguration.

To overcome the above limitations, we propose an *autonomous link recovery system* (ALR) that enables a multiradio WMN to autonomously recover from local link failures to preserve network

J.Jeevarekha, Assistant Professor, ECE Department, Muthayammal Engineering College, Rasipuram,Tamil Nadu .
(Mail-id: j.jeevarekha@gmail.com)
E.Hemalatha, Assistant Professor, IT Department, Muthayammal Engineering College,Rasipuram,Tamil Nadu .
(Mail id.=: hema.eit@gmail.com)
R.GowriShankar, Assistant Professor, IT Department, Muthayammal Engineering College, Rasipuram,Tamil Nadu .
(Mail id : rgowrishankarnit@gmail.com)

performance. By using channel and radio diversities in WMNs, ALR generates necessary changes in local radio and channel assignments in order to recover from failures. Next, based on the thus-generated configuration changes, the system cooperatively reconfigures network settings among *local* mesh routers. Briefly, ALR 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 ALR identifies reconfiguration plans that require the minimum number of changes for the healthy network settings.

ALR 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 .To ensure that the faulty link needs to be fixed via reconfiguration. To this end, ALR considers three primitive link changes to fix a faulty link ALR can use:

1) A channel-switch where both end-radios of link AB can simultaneously change their tuned channel

2) A radio-switch where one radio in node A can switch its channel and associate with another radio in node B;

3) A route-switch where all traffic over the faulty link can use a detour path instead of the faulty link.

ALR 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 hop reconfiguration parameter. Starting from a faulty link(s), ALR considers link changes within the first hops and generates feasible plans. If ALR cannot find a local solution, it increases the number of hops so that ALR may explore a broad range of link changes. The total number of reconfiguration changes is determined on the basis of existing configurations around the faulty.

First, ALR's planning algorithm effectively identifies reconfiguration plans that maximally satisfy the applications' QoS demands, accommodating twice more flows than static assignment. Next, ALR avoids the ripple effect via QoS-aware reconfiguration planning, unlike the greedy approach. Third, ALR's local reconfiguration improves network throughput and channel efficiency by more than 26% and 92%, respectively, over the local rerouting scheme. The rest of this paper is organized as follows. Section II provides the design rationale and algorithms of ALR. Section III shows in-depth simulation results of ALR. Section IV concludes the paper.

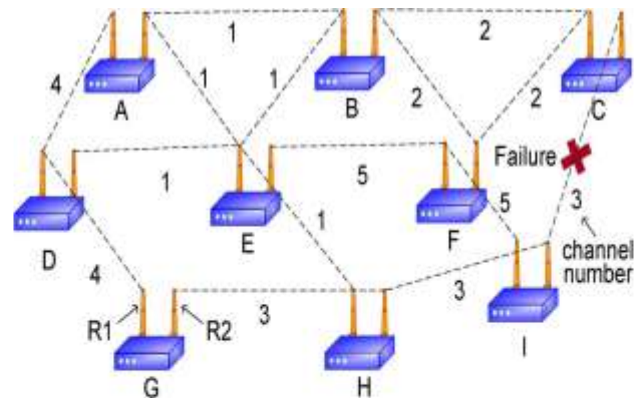


Figure 1. Multiradio WMN.AWMNhas an initial assignment of frequency channels as shown. The network often experiences wireless link failure and needs to reconfigure its settings.

II. ALR ARCHITECTURE

We first present the design rationale and overall algorithm of ALR. Then, we detail ALR's reconfiguration algorithms. Finally, we discuss the complexity of ALR.

A. Overview

ALR is a distributed system that is easily deployable in IEEE 802.11-based mr-WMNs. Running in every mesh node, ALR supports self-reconfigurability via the following distinct features.

• *Localized reconfiguration*: Based on multiple channels and radio associations available, ALR 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*: ALR effectively identifies QoS-satisfiable reconfiguration plans by: 1) estimating the QoS-satisfiability of generated

International Journal on Applications in Information and Communication Engineering
Volume 5 : Issue 2: December 2019, pp 10 – 18 www.aetsjournal.com ISSN (Online) : 2394 - 6237

reconfiguration plans; and 2) deriving their expected benefits in channel utilization.

Algorithm 1: ALR Operation at mesh node: *i*

(1) Monitoring period (t_m)
for every link *j* **do**
measure link-quality (*lq*) using passive monitoring;
end for
send monitoring results to a gateway *g*;
(2) Failure detection and group formation period (t_f)
if link *l* violates link requirements *r* **then**
request a group formation on channel *c* of link *l* ;
end if
participate in a leader election if a request is received;
(3) Planning period (M, t_p)
if node *i* is elected as a leader **then**
send a planning request message (*c*, *M*) to a gateway;
else if node *i* is a gateway **then**
synchronize requests from reconfiguration groups M_n
generate a reconfiguration plan *(p)*for M_i;
send a reconfiguration plan *(p)* to a leader of M_i ;
end if
(4) Reconfiguration period (p, t_r)
if *p* includes changes of node *i* **then**
apply the changes to links at *t*;
end if
relay *p* to neighboring members, if any

• *Autonomous reconfiguration via link-quality monitoring*:

ALR accurately monitors the quality of links of each node in a distributed manner. Furthermore, based on the measurements and given links' QoS constraints, ALR detects local link failures and autonomously initiates network reconfiguration.

• *Cross-layer interaction*:

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

Algorithm 1 describes the operation of ALR. First, ALR in every mesh node monitors the quality of its outgoing wireless links at every t_m (e.g., 10 s) and reports the results to a gateway via a management message. Second, once it detects a link failure(s), ALR 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 [29] for coordinating the reconfiguration. Third, the leader node sends a planning-request message to a gateway. Then, the gateway synchronizes the planning requests—if there are multiple 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.

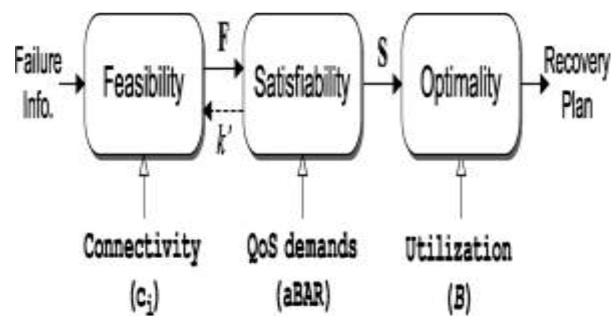


Figure 2. Localized reconfiguration planning in ALR. ALR generates a reconfiguration plan by breaking down the planning process into three processes with different constraints.

We assume that during the formation and reconfiguration, all messages are reliably delivered via a routing protocol and per-hop retransmission timer. In what follows, we will detail each of these operations, including how to generate reconfiguration plans, how to monitor link conditions such as bandwidth (Section II-B), and how much overhead ALR generates for the monitoring and for maintaining a reconfiguration group (Section II-C).

B. Planning for Localized Network Reconfiguration :

The core function of ALR 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. Existing channel-assignment and scheduling algorithms [5], [12], [13] seek "optimal" solutions by considering tight QoS constraints on all links,

International Journal on Applications in Information and Communication Engineering
Volume 5 : Issue 2: December 2019, pp 10 – 18 www.aetsjournal.com ISSN (Online) : 2394 - 6237

thus requiring a large configuration space to be searched and hence making the planning often an NP-complete problem [5]. In addition, change in a link's requirement may lead to completely different network configurations. By contrast, ALR 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. As depicted in Fig. 2, ALR first applies connectivity constraints to generate a *set* of feasible reconfiguration plans that enumerate feasible channel, link, and route changes around the faulty areas, given connectivity and link-failure constraints. Then, within the set, ALR applies strict constraints (i.e., QoS and network utilization) to identify a reconfiguration plan that satisfies the QoS demands and that improves network utilization most.

• *Feasible Plan Generation:*

Generating feasible plans is essentially to search all legitimate changes in links' configurations and their combinations around the faulty area. Given multiple radios, channels, and routes, ALR identifies feasible changes that help avoid a local link failure but maintain existing network connectivity as much as possible. However, in generating such plans, ALR has to address the following challenges.

• *Avoiding a faulty channel*:

ALR first has to ensure that the faulty link needs to be fixed via reconfiguration. To this end, ALR considers three primitive link changes. Specifically, to fix a faulty link(s), ALR can use: 1) a channel-switch *S* where both end-radios of link AB can simultaneously change their tuned channel; 2) a radio-switch R where one radio in node A can switch its channel and associate with another radio in node B; and 3) a route-switch *D* where all traffic over the faulty link can use a detour path instead of the faulty link.

• *Maintaining network connectivity and utilization*:

While avoiding the use of the faulty channel, ALR 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. To coordinate such propagation, ALR takes a *two-step* approach.

ALR 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. Furthermore, for the combination, ALR 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*:

ALR has to limit 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, ALR uses a *k*-hop reconfiguration parameter. Starting from a faulty link(s), ALR considers link changes within the first *k* hops and generates feasible plans. If ALR cannot find a local solution, it increases the number of hops *(k)* so that ALR 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*.

Let us consider an illustrative example in Fig.3. Given the failure in link CI, ALR first generates feasible and desirable changes per link (gray columns) using the primitives. Here, the changes must not include the use of a faulty or redundant channel. Next, ALR combines the generated per-link primitives of neighboring links to generate a set of feasible plans. During the combination, ALR has to preserve link and/or radio connectivities. For example, plans $S(C,I)_{3\rightarrow 6}$ and $S(H,I)_{3\rightarrow 3}$ in Fig. 3 cannot be connected because each change requires the same radio of node *I* to set up different channels. After the two steps, ALR has 11 feasible reconfiguration plans *(F)* by traversing connected changes of all links considered in the planning. Note that we set *k* to 2 in this example.

• *QoS-Satisfiability Evaluation:*

Among a set of feasible plans *F*, ALR 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 nonfaulty channels and maintain its connectivity, some plans might not satisfy the QoS constraints or may even cause cascaded QoS failures on neighboring links. To

filter out such plans, ALR has to solve the following challenges.

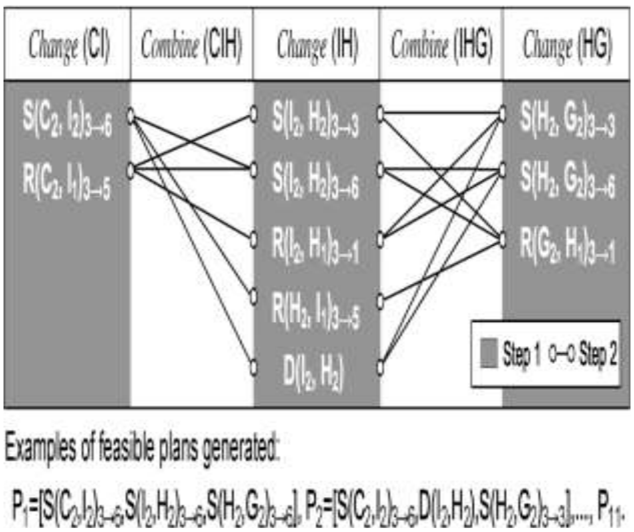


Figure 3. Example of network planning. ALR generates per-link changes (gray columns) and then connects them for feasible reconfiguration plans (white columns) for recovery of the failure in Fig.1.

• *Per-link bandwidth estimation*:

For each feasible plan,ALR has to check whether each link's configuration change satisfies its bandwidth requirement, so it must estimate link bandwidth. To estimate link bandwidth, ALR accurately measures each link's capacity and its available channel airtime. In multihop wireless networks equipped with a CSMA-like MAC, each link's achievable bandwidth (or throughput) can be affected by both link capacity and activities of other links that share the channel airtime. Even though numerous bandwidth-estimation techniques have been proposed, they focus on the average bandwidth of each node in a network [23], [30] or the end-to-end throughput of flows [17], which cannot be used to calculate the impact of per-link configuration changes. By contrast, ALR 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 [31]. Here, we assume that ALR is assumed to cache link-quality information for other channels and use the cached information to generate reconfiguration plans. If the information becomes obsolete, ALR detects link failures and triggers another reconfiguration to find QoS-satisfiable plans via lazy monitoring.

• *Examining per-link bandwidth satisfiability*:

Given measured bandwidth and bandwidth requirements, ALR has to check if the new link change(s) satisfies QoS requirements. ALR defines and uses the expected busy airtime ratio of each link to check the link's QoS satisfiability. Assuming that a link's bandwidth requirement *q* is given, the link's busy airtime ratio (BAR) can be defined as $BAR = q/C$ and must not exceed 1.0 (i.e., $BAR < 1.0$) for a link to satisfy its bandwidth requirement. If multiple links share the airtime of one channel, ALR calculates aggregate BAR *aBAR* of end-radios of a link, which is defined as $aBAR(k) = \sum_{l \in L(k)}(q_l/C_l)$, where *k* is a radio ID, a link associated with radio *k* , and *L(k)* the set of directed links within and across radio *k* 's transmission range.

• *Avoiding cascaded link failures*:

Besides the link change, ALR needs to check whether neighboring links are affected by local changes (i.e., cascaded link failures). To identify such adverse effect from a plan, ALR also estimates the QoS-satisfiability of links one hop away from member nodes whose links' capacity can be affected by the plan. If these 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.

Let us consider an example in Fig.4. Assuming BAR of each directed link (l_i) is 0.2 (e.g., 2 Mb/s 10 Mb/s) in a tuned channel, a *BAR* of each radio tuned to channel 1 does not exceed 1.0, satisfying each link's QoS requirement. In addition, assuming that $BAR(l_1)$increases from 0.2 to 0.4 in Fig.4. To accommodate this increase, reconfiguration plans that have a detour path through node Q do not affect the QoS-satisfiability of the neighboring nodes. On the other hand, plans with radio switches (e.g.,$R(L_2,M_1)_{1\rightarrow 2}$) satisfy the QoS of link MN but cause a $BAR(l_1)$ to exceed 1.0, resulting in cascaded QoS failures of links beyond node O.

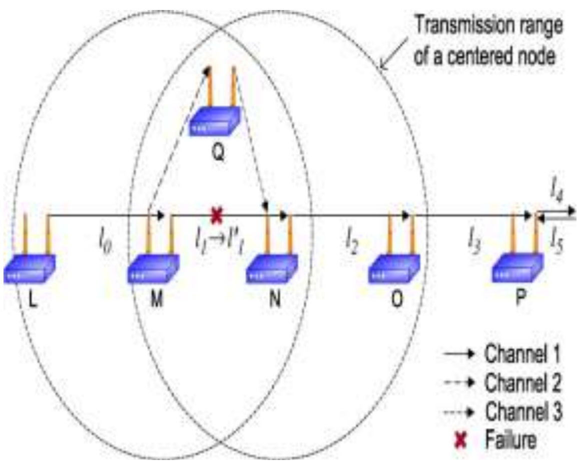


Figure 4. Busy airtime ratio (BAR) of a directed link. BAR (e.g.,l_1) is affected by activities of neighboring links (l_0 ,l_2 , and l_3) in channel 1 and is used to evaluate QoS satisfiability of a link.

• *Choosing the Best Plan:*

ALR 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. However, to incorporate the notion of fair share into the planning, ALR needs to address the following challenges.

• *Quantifying the fairness of a plan*:

ALR has to quantify the potential changes in link-capacity distribution from a plan. To this end, ALR 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 $B(p) = (1/n) \sum_{k=1}^{n} \beta(k)$, where $\beta(k)$ is the relative improvement in the airtime usage of radio k , and n the number of radios whose $\beta(k)$ has changed from the plan. This definition allows the benefit function to quantify the overall change in airtime usage, resulting from the reconfiguration plan. Here, $\beta(k)$ is considered as a fairness index on the usage of channel airtime, and it is defined as follows:

$$\beta(k)=\begin{cases} \epsilon_1(k)-\epsilon_2(k), & \text{if } \epsilon_1(k), \epsilon_2(k) > \delta \\ \epsilon_2(k)-\epsilon_1(k), & \text{if } \epsilon_1(k),\epsilon_2(k) < \delta \\ \epsilon_1(k)+\epsilon_2(k)-2\delta, & \text{if } \epsilon_1(k) > \delta > \epsilon_2(k) \\ 2\delta-\epsilon_1(k)-\epsilon_2(k), & \text{if } \epsilon_1(k) > \delta > \epsilon_1(k) \end{cases}$$

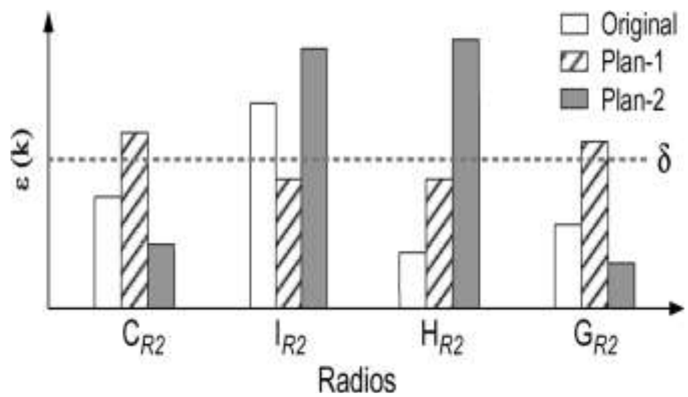


Figure 5. Benefit function. Prefers a reconfiguration plan that improves overall channel utilization close to the desired parameter

Where $\epsilon_1(k)$ and $\epsilon_2(k)$are estimated a *BAR* 's of a radio k in existing configurations and in new configurations, respectively, and the desired channel utilization.

• *Breaking a tie among multiple plans*:

Multiple reconfiguration plans can have the same benefit, and ALR needs to break a tie among them. ALR 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.

Suppose δ is 0.5 as shown in Fig. 5. ALR favors a plan that reconfigures links to have 50% available channel airtime (e.g., plan 1 in the figure). If a plan reconfigures a WMN to make the links heavily utilized while idling others (e.g., plan 2), then the benefit function considers the plan ineffective, placing the plan in a lowly ranked position. The effectiveness of β and δ will be evaluated and discussed further in Section V-B2. Equation (1)

implies that if a reconfiguration plan makes overall links' channel utilization closer to the desired utilization δ , then $\beta(k)$ gives a positive value, while giving a negative value otherwise.

C. Complexity of ALR

Thanks to its distributed and localized design, ALR 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 kb/s probing bandwidth (i.e., one packet per second) for each radio. In addition, the group formation requires only *O(n)* message overhead (in forming a spanning tree), where *n* is the number of nodes in the group. Next, the computational overhead in ALR 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. Next, 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.

III.PERFORMANCE EVALUATION

We have also evaluated ALR in large-scale network settings via simulation. We first describe our simulation methodology, and then present the evaluation results on ALR.

A. Simulation Model and Methods ns-2 [37] is used in our simulation study.

Throughout the simulation, we use a grid topology with 25 nodes in an area of 1x1 km^2 , as shown in Fig. 6(a). In the topology, adjacent nodes are separated by 180 m, and each node is equipped with a different number of radios, depending on its proximity to a gateway. The gateway is equipped with four radios, one-hopaway nodes from a gateway have three radios, and other nodes have two radios.

For each node in this topology, we use the following network protocol stacks. First, the shadowing propagation model [38] is used to simulate varying channel quality and multipath effects. Next, CMU 802.11 wireless extension is used for the MAC protocolwith a fixed data rate (i.e., 11 Mb/s) and is further modified to support multiple radios and multiple channels. Finally, a link-state routing protocol, a modification of DSDV [32], and multiradio-aware routing metric (WCETT [6]) are implemented and used for routing.

In these settings, ALR is implemented as an agent in both the MAC layer and a routing protocol as explained in Sections II. It periodically collects channel information from MAC and requests channel switching or link-association changes based on its decision. At the same time, it informs the routing protocol of network failures or a routing table update.

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 kb/s 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 s, and the results of 10 runs are averaged unless specified otherwise.

B. Evaluation Results

1) Effectiveness of QoS-Aware Planning:

We measured the effectiveness of ALR 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 ALR can generate via reconfiguration. As shown in the tables of Fig. 6(b), ALR reconfigures a wireless mesh network to meet different QoS requirements. Before each reconfiguration, the gray areas can only accept 1–9 UDP flows. On the other hand, after reconfiguration, the network in the areas can admit 4–15 additional flows, improving the average network capacity of the gray areas by 3.5 times.

2)Impact of the Benefit Function:

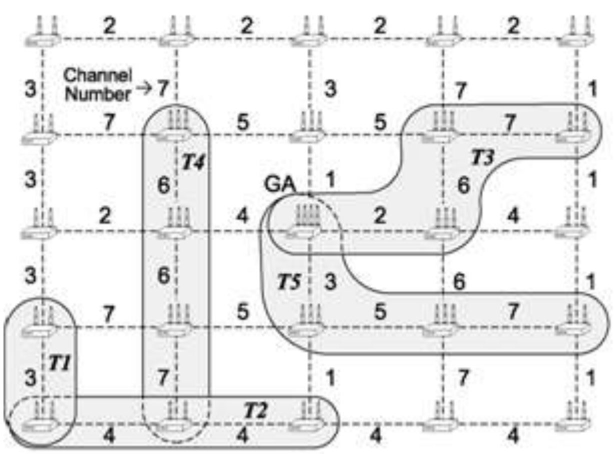
We also studied the impact of the benefit function on the ALR's planning algorithm. We conducted the same experiment as the previous one with different values of δ in the benefit function. As shown in Fig. 6(b), a high value (0.8) of δ allows ALR to keep local channel efficiency high. By contrast, a low value (0.4) can deliver more available bandwidth (on average, 1.2 Mb/s) than

International Journal on Applications in Information and Communication Engineering
Volume 5 : Issue 2: December 2019, pp 10 – 18 www.aetsjournal.com ISSN (Online) : 2394 - 6237

when the high value is used since ALR tries to reserve more capacity.

3)Impact of the Reconfiguration Range:

We evaluated the impact of the reconfiguration range. We used the same experiment settings as the previous one and focused on reconfiguration requests at T1 . As we increase the hop count *k* from a faulty link(s), we measure the capacity improvement achieved by the reconfiguration plans. In addition, we calculate the capacity gain per change as the cost-effectiveness of reconfiguration planning with different *k* values. Plot the available capacity of the faulty area after reconfigurations. ALR can improve the available links' capacity by increasing the reconfiguration range. However, its improvement becomes marginal as the range increases. This saturation results mainly from the fixed number of radios of each node. In other words, the improvement is essentially bounded by the total capacity of physical radios. Furthermore, because reconfiguration plans with a larger range are required to incur more changes in network settings, the bandwidth gain per change significantly degrades (e.g., capacity gain per change at the hop count of 4 as in Fig. 7.) We also observed the similar results in other reconfiguration requests (T2,T3,T4), but omitted them for brevity.



(a)

Table-1: $\delta = 0.8$

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: $\delta = 0.4$

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)

Figure 6. Satisfying varying QoS constraints. (a) Requests with different QoS requirements. (b) Improved (or changed) network capability (i) before and (ii) after reconfiguration.

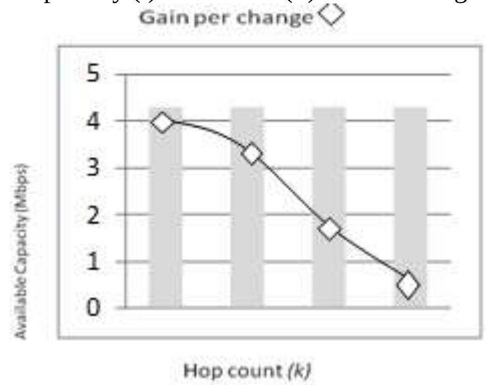


Figure 7. Impact of reconfiguration range. The hop length can help ARS search for reconfiguration plans. However, the benefit from the increased length is small, whereas the number of total changes for the reconfiguration adversely increases.

IV. CONCLUSION

We first make concluding remarks and then discuss some future work.

A. Concluding Remarks

This paper presented an autonomous link recovery system (ALR) that enables a multiradio WMN to autonomously recover from wireless link failures. ALR generates an effective reconfiguration plan that requires only local network configuration changes by exploiting channel, radio, and path diversity. Furthermore, ALR effectively identifies reconfiguration plans that satisfy applications' QoS constraints, admitting up to two times more flows than static assignment, through QoS-aware planning. Next, ALR's online reconfigurability allows for real-time failure detection and network reconfiguration, thus improving channel efficiency by 92%. Our implementation on ns2-based simulation have demonstrated the effectiveness of ALR in recovering from local link-failures and in satisfying applications' diverse QoS demands.

B. Future Work

Joint Optimization With Flow Assignment and Routing: ALR decouples network reconfiguration from flow assignment and routing. Reconfiguration might be able to achieve better performance if two problems are jointly considered. Even though there have been a couple of proposals to solve this problem [5], [12], they only provide theoretical bounds without considering practical system issues. Even though its design goal is to recover from network failures as a best-effort service, ALR is the first step to solve this optimization problem, which we will address in a forthcoming paper.

Use of ALR in IEEE 802.11 b/g WMNs:

ALR is can also mainly be evaluated in IEEE 802.11a networks, where 13 orthogonal channels are available. However, ALR can also be effective in a network with a small number of orthogonal channels (e.g., three in IEEE 802.11b/g). Because ALR includes a link-association primitive, it can with idle interfaces of neighboring nodes, and it further limits the range of a reconfiguration group (e.g., nodes within 4 hops).

References

[1] Dr.E.Punarselvam ,S.Gopi " Effective and Efficient traffic scrutiny in sweet server with data privacy" in the link " http://www.aetsjournal.com/journal_issues/Effective-And-Efficient-Traffic-Scrutiny-In-Sweet-Server-With-Data-Privacy.pdf ".

[2] Channel Allocation and Routingin Hybrid Multichannel Multiradio Wireless Mesh NetworksYong Ding, Kanthakumar ieee feb 2013.

[3] I. Akyildiz, X. Wang, and W. Wang, "Wireless mesh networks: survey," *Comput. Netw.*, vol. 47, no. 4, pp. 445–487, Mar. 2005.

[4] P. Kyasanur and N. Vaidya, "Capacity of multi-channel wireless networks: Impact of number of channels and interfaces," in *Proc. ACM MobiCom*, Cologne, Germany, Aug. 2005, pp. 43–57.

[5] K. Ramanchandran, E. Belding-Royer, and M. Buddhikot, "Interference- aware channel assignment in multi-radio wireless mesh networks," in *Proc. IEEE INFOCOM*, Barcelona, Spain, Apr. 2006, pp. 1–12.

[6] R. Draves, J. Padhye, and B. Zill, "Routing in multi-radio, multi-hop wireless mesh networks," in *Proc. ACM MobiCom*, Philadelphia, PA,Sep. 2004, pp. 114–128.

[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 *Proc. ACM MobiCom*, Philadelphia, PA, Sep. 2004, pp. 216–230.

[8] D. Aguayo, J. Bicket, S. Biswas, G. Judd, and R. Morris, "Link-level measurements from an 802.11b mesh network," in *Proc. ACM SIGCOMM*, Portland, OR, Aug. 2004, pp. 121–132.

[9] A. Akella, G. Judd, S. Seshan, and P. Steenkiste, "Self-management in chaotic wireless deployments," in *Proc. ACM MobiCom*, Cologne,Germany, Sep. 2005, pp. 185–199.

[10] J. Zhao, H. Zheng, and G.-H. Yang, "Distributed coordination in dynamic spectrum allocation networks," in *Proc. IEEE DySPAN*, Baltimore, MD, Nov. 2005, pp. 259–268.

[11] M. J. Marcus, "Real time spectrum markets and interruptible spectrum: New concepts of spectrum use enabled by cognitive radio," in *Proc.IEEE DySPAN*, Baltimore, MD, Nov. 2005, pp. 512–517.