ES2: Managing Link Level Parameters for Elevating Data Rate and Stability in High Throughput WLAN

Sandip Chakraborty
Department of Computer Science and Engineering,
Indian Institute of Technology Kharagpur,
Kharagpur, India 721302
Email: sandipc@cse.iitkgp.ernet.in

Subhrendu Chattopadhyay
Department of Computer Science and Engineering,
Indian Institute of Technology Guwahati,
Guwahati, India 781039
Email: subhrendu@iitg.ernet.in

***Abstract*—High Throughput (HT) IEEE 802.11n/ac wireless networks support a large set of configuration parameters, like Multiple Input Multiple Output (MIMO) streaming, modulation and coding scheme, channel bonding, short guard interval, frame aggregation levels etc., that determine its physical data rate. However, all these parameters have an optimal performance region based on the link quality and external interference. Therefore, dynamically tuning the link parameters based on channel condition can significantly boost up the network performance. The major challenge in adapting all these parameters dynamically is that a large feature set need to be enumerated during run-time to find out the optimal configuration, which is a not feasible in real time. Therefore in this paper, we propose an estimation and sampling mechanism to filter out the non-preferable features on the fly, and then apply a learning mechanism to find out the best features dynamically. We apply a Kalman filtering mechanism to figure out the preferable feature sets from all possible feature combinations. A novel metric has been defined, called the *diffESNR*, which is used to select the best features from the sampled feature sets. The proposed scheme, *Estimate-Sample-Select* (*ES2*) is implemented and tested over a mixed wireless testbed using IEEE 802.11n and IEEE 802.11ac HT wireless routers, and the performance is analyzed and compared with other related mechanisms proposed in the literature. The analysis from the testbed shows that *ES2* results in approx 60% performance improvement compared to the standard and other related mechanisms.**

***Keywords*-High Throughput Wireless; link adaptation; IEEE 802.11n; IEEE 802.11ac**

I. Introduction

The invent and wide-spread deployments of high throughput (HT) wireless technologies, like IEEE 802.11n or IEEE 802.11ax [1]–[3], give a breakthrough to the wireless local area networking (WLAN) for commodity and community wireless usage over free bandwidth. The HT wireless technologies support high physical data rates (IEEE 802.11n supports data rates up to 600 Mbps, whereas IEEE 802.11ax supports rate in Gbps) through a number of physical and media access control (MAC) layer advancements; like the inclusion of multiple antenna technology through *Multiple Input Multiple Output* (MIMO), channel bonding and frame aggregation. The MIMO supports spatial division multiplexing (SDM) with space time block coding (STBC) for improving network capacity. The IEEE 802.11n standard supports 2 spatial multiplexed streams. The channel bonding feature provides wider channels of 40 MHz (for IEEE 802.11n) or 80 MHz (for IEEE 802.11ax) by combining multiple narrow channels of 20 MHz together. Frame aggregation combines multiple upper layer frames, or protocol data unit (PDU), to reduce channel access overhead.

Apart from HT wireless specific features, modern wireless routers also provide different modulation and coding schemes (MCS) that support different data rates. The existing studies [4], [5] have revealed that optimal selection of MCS depends on channel and network conditions, and accordingly different rate adaptation mechanisms have been proposed for legacy WLAN networks. However, HT wireless networks combine a large number of feature set combinations based on the link configuration parameters, like number of spatial streams, channel bondings, guard intervals, MCS levels, frame aggregation levels etc. For example, IEEE 802.11n supports 256 possible combinations of feature sets - 4 spatial streams, 2 channel widths (20 MHz and 40 MHz), 2 guard intervals (800 ns and 400 ns), 8 different MCS levels and 2 frame aggregation levels (ON or OFF). A number of recent studies [5]–[14] have shown that dynamic adaptation from these feature sets provides better network performance compared to static assignments. Accordingly, several link adaptation mechanisms have been proposed in the literature, and Minstrel HT [15] has been widely accepted as the default link adaptation mechanism for HT wireless networks.

The existing link adaptation mechanisms for HT wireless networks can be broadly classified into two groups - open loop link adaptation and closed loop link adaptation. Open loop link adaptation [5]–[7] uses some form of sampling to find out the best set of features from the available feature set. However, the feature sets are extensively large for online processing and therefore, the existing approaches use some kind of filtering on static assignments of few features. The closed loop approaches [8] rely on receiver feedback which incurs significant overhead and processing delay to the network. Consequently, the link adaptation mechanism for HT wireless network remains an open issue to the research community.

In summary, link adaptation in HT wireless networks faces two challenges - first, the number of features in the feature space is very large for on-line processing and optimization, and second, both capacity (or throughput) and fairness need to be

TABLE I
MCS LEVELS AND CORRESPONDING DATA RATES FOR IEEE 802.11N

MCS	Spatial Streams	Modulation Type	Coding Rate	20 MHz, 400ns	20 MHz, 800ns	40 MHz, 400ns	40 MHz, 800ns
4	1	16-QAM	3/4	39	43.3	81	90
7	1	64-QAM	5/6	65	72.2	135	150
11	2	16-QAM	1/2	52	57.8	108	120
15	2	64-QAM	5/6	130	144.4	270	300

adjusted simultaneously. This paper uses a testbed analysis to understand the impact of different link configuration features; like MIMO spatial streams, MCS levels, channel bonding, guard intervals and frame aggregation, over the performance of the HT wireless networks. Based on the analysis, we design a hybrid link adaptation mechanism to select the best set of link features and tune them adaptively based on channel condition. The proposed link adaptation mechanism works in three phases. In the first phase, we use a Kalman filter approach to predict the signal to noise ratio (SNR) at the transmitter based on receiver feedback. In the second phase, the estimated SNR value is used to design a novel metric, called *differentiated estimated SNR* (diffESNR), which is used to sample the set of features for finding out most appropriate link parameters. *diffESNR* considers performance while reserving fairness during the sampling procedure from the link configuration feature set. In the third phase, the conventional rate adaptation mechanism is applied over the sampled set of features to find out the complete link configuration parameters based on channel condition. The proposed link adaptation mechanism, called *Estimate, Sample and Select* (*ES2*), is implemented over a testbed, and the performance is analyzed and compared with other related works available in the literature. The testbed analysis reveals that *ES2* significantly improves goodput and network fairness compared to `Minstrel HT` [15] and SampleLite [6], two other popular dynamic link adaptation mechanisms, while keeps the signaling overhead minimum.

The rest of the paper is organized as follows. Section II analyzed the impact of link configuration parameters over protocol performance through a thorough testbed analysis. The proposed adaptive link management methodology has been discussed in Section III. Section IV analyzes and compares the performance of the proposed mechanism from the testbed analysis. The impact of proposed *ES2* mechanism over a cross technology testbed (IEEE 802.11n and IEEE 802.11ac mixed testbed) has been discussed in Section V. Finally, Section VI concludes the paper.

II. MOTIVATION: IMPACT OF LINK CONFIGURATION PARAMETERS

In this section, we analyze impacts of different link adaptation parameters, namely channel bonding, MCS selection (no of streams, modulation type and coding rates), guard intervals and frame aggregation (aggregation ON/OFF), over the performance of IEEE 802.11n and IEEE 802.11ac wireless networks.

A. Testbed Configuration

We use three different set of testbed configurations to observe the impact of different link parameters over network performance;

1) IEEE 802.11n only testbed
2) IEEE 802.11ac only testbed
3) IEEE 802.11n/ac mixed testbed

1) ***IEEE 802.11n only testbed:*** : The IEEE 802.11n only testbed consists 18 IEEE 802.11n supported wireless nodes - 6 access points (AP) and 12 wireless stations (STA). The network is configured as a basic service set (BSS), where we vary the number of contending nodes in the BSS. Every node is a RaLink (MediaTek) RT-3352 router-on-chip (RoC) that supports IEEE 802.11b/g/n compatibility. The RoC supports $2T2R$ MAC along with BBP/PA/RF MIMO, a high performance 400 MHz MIPS24KEc CPU core, a Gigabit Ethernet MAC, 5-ports integrated $10/100$ Ethernet Switch/PHY, 64 MB of SDRAM and 32 MB of Flash. This chip can support up to MCS 15 with a peak data rate of 300 Mbps in 40 MHz, 400 ns guard interval. The maximum MCS (MCS 15) supports 2 spatial streams, 64-QAM modulation and $5/6$ coding rate. The APs are connected with Gbps backbone network via wired connection. The RoCs are equipped with Linux Kernel version 3.18. We use `iperf` as the tool for network traffic generation.

2) ***IEEE 802.11ac only testbed:*** : The IEEE 802.11ac only testbed consists 8 IEEE 802.11ac supported wireless nodes - two APs and 8 STAs. As earlier, the network is configured as a basic service set (BSS). Every AP in this testbed is an Asus RT-AC3200 IEEE 802.11ac router, and the STAs are IEEE 802.11ac Asus USB-AC56 client boards. The APs are equipped with 3×3 multi-user MIMO (MU-MIMO) that supports a peak data rate of 1300 Mbps at 5 GHz channel with 80 MHz channel bandwidth. These routers support three channel bonding scenarios: 20 MHz, 40 MHz and 80 MHz. The routers and clients are equipped with open-source `asuswrt-merlin` firmware built over Linux Kernel version 3.18.

3) ***IEEE 802.11n/ac mixed testbed:*** : In this scenario, both the IEEE 802.11n and IEEE 802.11ac routers are used, while the IEEE 802.11ac routers are configured in n+ac mixed mode. The analysis from this testbed configuration shows the interaction and interoperability among IEEE 802.11n and IEEE 802.11ac routers.

The testbed configurations are shown in Figure 1. For IEEE 802.11n testbed, we configure two basic service sets (BSS) with 3 APs each, for IEEE 802.11ac testbed, there are two BSSes with 1 AP each, and in the mixed testbed, we have

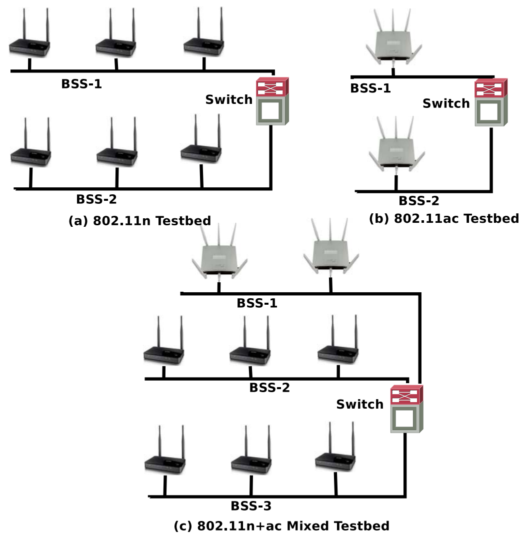

Fig. 1. Testbed Configuration

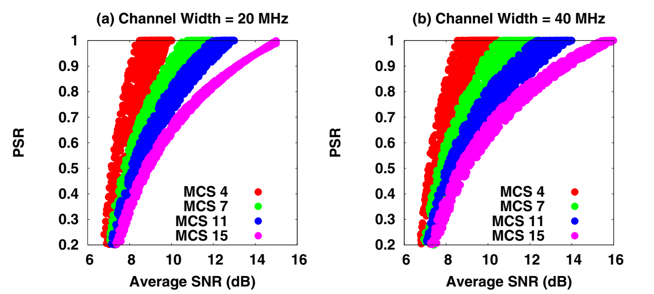

Fig. 2. Impact of MCS Selection over PSR (IEEE 802.11n)

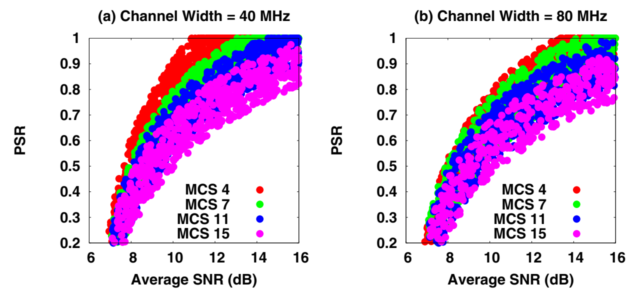

Fig. 3. Impact of MCS Selection over PSR (IEEE 802.11ac)

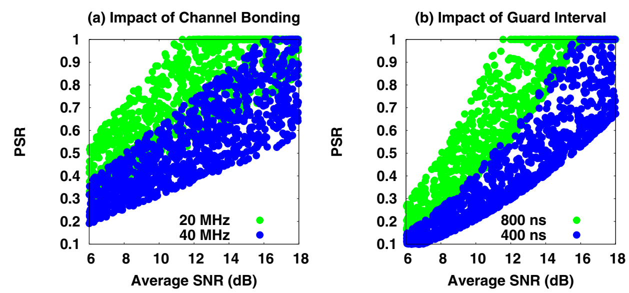

Fig. 4. Impact of Channel Bonding and Guard Intervals (IEEE 802.11n)

three BSSes for two BSSes with 3 IEEE 802.11n APs and one BSS with 2 IEEE 802.11ac APs. All the BSSes are connected to a common 10 Gbps switch.

B. Protocol and Metric Selection

We use `Minstrel` [16] as the base link adaptation protocol which is widely used in present Linux kernel network modules. `Minstrel` uses packet success rate (PSR) as the metric for rate adaptation. As discussed in several works in the literature [6], [10], [17], the PSR of a wireless link depends on the channel condition that can be measured in terms of signal to noise ratio (SNR). In these sets of experiments, we use SNR and PSR as the measurement metrics.

C. Impact of MCS Selection and MIMO Streaming

To check the impact of MCS selection and MIMO streaming, we fix the channel bonding and short guard interval. All the nodes transmit data using predefined MCS levels. We consider MCS levels 4, 7, 11 and 15. The number of spatial streams, modulation type, coding rates, and data rates corresponding to different {Channel width, Guard Interval} settings for the selected MCS levels have been shown in Table I. We consider 20 MHz, 40 MHz (for IEEE 802.11n) and 40 MHz, 80 MHz (for IEEE 802.11ac) communications at 5 GHz band, for these experiments. The data is collected from all the three testbed configurations (in both pure mode and mixed mode) and the average data points are plotted in the graphs.

Fig. 2 and Fig. 3 show the PSR for different MCS levels with respect to SNR value, for IEEE 802.11n and IEEE 802.11ac, respectively. The figures indicate that low MCS levels can sustain at low SNR region and provide better PSR compared to high MCS values. However high MCS levels provide good PSR at high SNR region. It can be observed further that PSR variation in significantly more in single stream communication compared to double stream communication. Fig. 6(b) reveals that 40 MHz and 80 MHz channels perform better at high SNR region, however it increase PSR variation significantly compared to 20 MHz channel. Such a variation in PSR impacts fairness of the network.

D. Impact of Channel Bonding and Guard Intervals

To analyze the impact of channel bonding, we have generated the SNR-PSR curves for different MCS levels at both 20 MHz and 40 MHz channels for IEEE 802.11n, and additionally 80 MHz channel for IEEE 802.11ac. Due to space constraint, only the graph for MCS 15 is discussed in this paper, as shown in Fig. 4(a) and Fig. 5(a). The figures indicate that 20 MHz performs better in low SNR region. This is also discussed in existing literatures [13], [14] that 40 MHz and 80 MHz wider channels get affected by the external noise and interference. The similar analysis has been done for guard intervals, as shown in Fig. 4(b) and Fig. 5(b). IEEE 802.11n/ac HT wireless

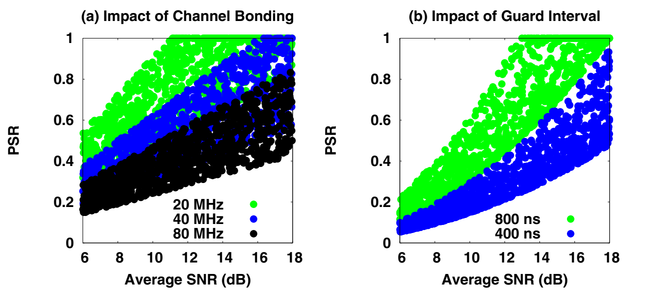

Fig. 5. Impact of Channel Bonding and Guard Intervals (IEEE 802.11ac)

supports short guard interval (400 ns) which is effective for low interference scenario. The figure also indicates that 400 ns performs good at high SNR region.

E. Observations from the Testbed Analysis

The testbed analysis shows that different link parameters, like MIMO streaming, MCS levels, channel bonding, guard interval and frame adaptation together decides the capacity of a HT wireless link. It can be noted that although we have not reported the results for frame aggregation, it has been observed that frame aggregation works better at high SNR region. Therefore the existing works [5], [7], [8], [12] that consider only a subset of these parameters provide a suboptimal solution to the adaptive link management problem. The adaptive link management for HT wireless networks should consider all of these parameters simultaneously to find out the optimal link configuration. However, the major challenge is to enumerate all the possible combinations at run-time to find out the optimal solution which takes infeasible amount of time. Therefore in this paper, we use an estimation and sampling mechanism to filter out the non-preferable configurations before going to find out the optimal solution. The detailed procedure is discussed in the next section.

III. Adaptive Link Management: Estimate, Sample and Select

Based on the observations from the testbed analysis, we design a three phase mechanism for adaptive link management in HT wireless networks. We exploit the acknowledgement (ACK) feedback mechanism, where the receiver piggybacks extra information for coordination with the transmitter. Although several works in the literature use open-loop link adaptation to reduce the transmission overhead, we believe that hybrid link adaptation can perform much better in this scenario because of the complex interdependencies among different parameters along with throughput-fairness trade-off as discussed in the previous section. However, even with a receiver feedback, the problem of adaptive link management is non-trivial because of the accurate estimation of channel quality and the design of switching strategy from one configuration set to another.

The three phase mechanism for adaptive link management in HT wireless networks works as follows:

1) Estimate the SNR at transmitter from the measured received signal strength (RSS) at the receiver,
2) Sample the feature sets based on the estimated SNR thresholds,
3) Select the final data rate from the filtered samples.

The detailed design of the adaptive link management mechanism is given next.

A. Estimation of SNR based on Kalman Filter

Estimation of SNR in HT wireless networks is nontrivial because of two reasons,

(i) The legacy hardware devices measure the received signal strength for each packet arrived at the receiver interface. The noise level significantly depends on parametric settings (number of spatial streams, channel width, short guard interval) of the neighboring nodes, and therefore fluctuates abruptly with respect to time. Therefore simple subtraction of noise margin from the signal strength (as done in many legacy IEEE 802.11 drivers, like MadWiFi and its extensions [18]) may not give a good estimate.

(ii) The transmitter needs to sample from the link feature set, whereas the performance of the link depends on receiver side channel quality, as the receiver gets affected from channel interference.

For the estimation of SNR, we use Kalman Filter [19] based approach to subtract the noise floor estimate from the RSS. Let $\bar{\mathcal{R}}_t$ and $\mathcal{R}_t$ be the estimated RSS and measured RSS (when a packet is received), respectively, at time t. With every ACK packet, the receiver piggybacks the information $\{\mathcal{R}_t, t\}$ from which the transmitter estimates $\bar{\mathcal{R}}_t$. Let the estimated noise be modeled as a Gaussian random variable $\mathcal{N}_t \sim N(0, \mathcal{Q})$ with variance $\mathcal{Q}$. Similarly, the measured noise is captured by another Gaussian random variable $\nu_t \sim N(0, \mathcal{S})$ where $\mathcal{S}$ is its variance. Then the system can be captured using following set of equations;

$$\bar{\mathcal{R}}_t = \bar{\mathcal{R}}_{t-1} + \mathcal{N}_{t-1} \tag{1}$$
$$\mathcal{R}_t = \mathcal{R}_{t-1} + \nu_{t-1} \tag{2}$$

Therefore, we need a model for RSS behavior from eq. (1) and another model from noise estimate from eq. (2). The SNR (dB) can be computed by subtracting the noise floor (dBm) from RSS estimate (dBm).

Let $\bar{\mathcal{R}}_t^-$ and $\bar{\mathcal{R}}_t^+$ be the a-priori and post-priori estimate of the RSS. $\bar{\mathcal{P}}_t^-$ and $\bar{\mathcal{P}}_t^+$ are the a-priori and post-priori estimates of the error variance. Then the *time update equations* [19] for Kalman filter work as follows,

$$\bar{\mathcal{R}}_t^- = \bar{\mathcal{R}}_{t-1}^+ \tag{3}$$
$$\bar{\mathcal{P}}_t^- = \bar{\mathcal{P}}_{t-1}^+ + \mathcal{Q} \tag{4}$$

Similarly, the *filter measurement equations* [19] for the Kalman filter work as follows,

$$\mathcal{K}_t = \bar{\mathcal{P}}_t^-(\bar{\mathcal{P}}_t^- + \int)^{-1} \tag{5}$$
$$\bar{\mathcal{R}}_t^+ = \bar{\mathcal{R}}_t^- + \mathcal{K}_t(\mathcal{R}_t - \bar{\mathcal{R}}_t^-) \tag{6}$$
$$\bar{\mathcal{P}}_t^+ = (1 - \mathcal{K}_t)\bar{\mathcal{P}}_t^- \tag{7}$$

Here K_t is the *Kalman gain*. The set of equations given above updates the error variance at every instance of time, and finds out the RSS and the noise floor. It is well understood that Kalman filter minimizes the mean square error at every iteration, and predicts the optimal noise variables for Gaussian distribution.

Therefore in the proposed link quality estimation mechanism, the receiver piggybacks the measures RSS and noise level with the ACK packets, and the transmitter estimates the SNR at every time instance based on the Kalman filter approach. This estimated SNR value is used in the next step to sample the features from the available set of feature set.

B. Sampling of the Feature Set

As discussed earlier, the number of features for adaptive link management in HT wireless networks is significantly large. Therefore, the transmitter needs to sample the possible features from the available feature set, that can be considered in the next step to find out the optimal link setup parameters. For this purpose, we apply a similar method as proposed in [6]. However our scheme use a different metric, termed as *diffESNR*, instead of transmitter side average RSS. The *diffESNR* at time t is defined as the squared difference between the estimated SNR in time t and time $t-1$ multiplied by the SNR at time t. Let $\bar{\mathcal{S}}_t$ be the estimated SNR at time t. Then diffESNR_t can be defined as;

$$\text{diffESNR}_t = \bar{\mathcal{S}}_t(|\bar{\mathcal{S}}_t^2 - \bar{\mathcal{S}}_{t-1}^2|) \tag{8}$$

The diffESNR parameter considers the current estimated SNR along with the variance in SNR with respect to the time scale. Therefore it intelligently considers both the performance as well as fairness, as the increased variation in SNR may result in unfairness in link adaptation. Whenever the SNR variance is high, diffESNR can select the features which will in turn reduce the performance variation in the link. For example, as shown in Fig. 2 and Fig. 3, 20 MHz works well when SNR variation is high, as it reduces the PSR variation. It has been observed from the testbed analysis that although the SNR value fluctuates abruptly with respect to time and the best link adaptation parameters are not monotonic with respect to SNR, but they follow monotonic behavior with respect to diffESNR. This has been established by following results as discussed next.

In Fig. 6 and Fig. 7, we have shown the effect of diffESNR in the selection of spatial streams and channel bonding. The network and link setup for this experiment is kept similar as discussed in previous section. Fig. 6 and Fig. 7 indicate that a low diffESNR implies single stream link to be more preferable compared to double stream. Intuitively when diffESNR is less, SNR variation gets decreased and also the end-to-end SNR is on the lower side. The single stream link can sustain at low SNR values with small variation. On the other side, high SNR variation can be controlled by double stream links at the presence of high SNR values. Therefore, double stream provides better result in that scenario. The figure indicates that with diffESNR threshold at 20 dB, the communication can be

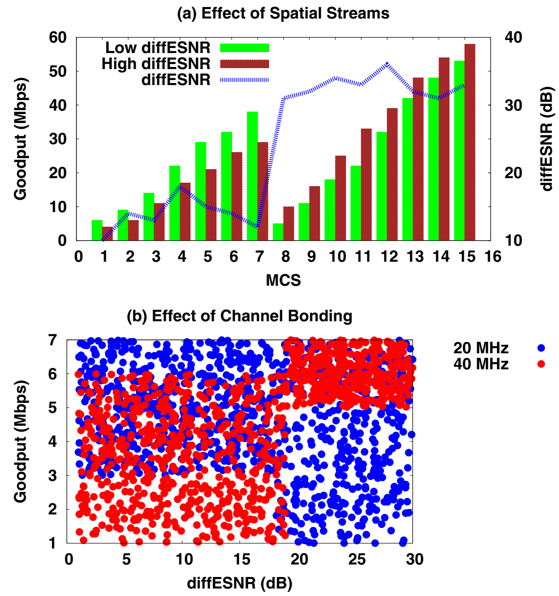

Fig. 6. Effect of diffESNR over Spatial Streams and Channel Bonding (IEEE 802.11n)

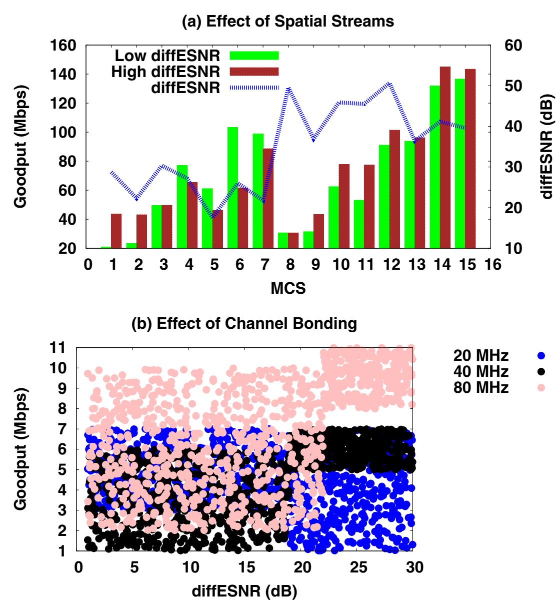

Fig. 7. Effect of diffESNR over Spatial Streams and Channel Bonding (IEEE 802.11ac)

switched from the single stream mode to the double stream mode. Similar observation can be done for the selection of channel bonding. 40 MHz channel requires high SNR value with less SNR fluctuation. To get a fair estimation of the impact of diffESNR over channel bonding, we have used MCS 0 at 40 MHz channel and MCS 1 at 20 MHz channel. MCS 0 at 40 MHz provides 13.5 Mbps whereas, MCS 1 at 20 MHz supports 13 Mbps physical data rate. We compute the average goodput for these two cases, and plot the same in Fig. 6(b) and Fig. 7(b). The figures indicate that at low diffESNR, 20

Algorithm 1 Filtering out Non-Preferable Choices from the Set of Link Features

```
if diffESNR ≤ Threshold(Streams) then
    Stream = Single Stream
else
    Stream = Double Stream
end if
if diffESNR ≤ Threshold(Channel_20) then
    Channel = 20 MHz
else
    if diffESNR ≤ Threshold(Channel_40) then
        Channel = 40 MHz
    else
        Channel = 80 MHz
    end if
end if
if diffESNR ≤ Threshold(GI) then
    GI = 800 ns
else
    GI = 400 ns
end if
if diffESNR ≤ Threshold(Aggregate) then
    Aggregate = OFF
else
    Aggregate = ON
end if
```

MHz gives better result whereas at high diffESNR, 40 MHz outperforms. 80 MHz performs better in further increase of diffESNR value. In this scenario, the switching from 20 MHz to 40 MHz can be triggered at a threshold of 18 dB. Similarly, a transition from 40 MHz to 80 MHz can be triggered with a threshold of 22 dB. Similar observations have been found for guard interval and frame aggregation, however not reported in this paper due to the space constraint.

Based on the above observation, the transmitter defines diffESNR thresholds for the spatial streams, channel bonding, short guard interval and frame aggregation. Based on these thresholds, a simple mechanism is executed for filtering out the non-preferable choices from the set of features, as shown in Algorithm 1. It can be noted that the threshold values are note fixed, but they are adaptive based on the observation of the link performance. Initially the threshold value is fixed to some user defined value based on experimental evidences. We use an interleaved monitoring mechanism, where the transmitter periodically updates one threshold at a time by ± 1 dB and ± 2 dB, transmits a few test HELLO packets and observes the PSR. If there is an improvement in PSR, it updates the corresponding threshold. To bound such signaling overhead, we limit such HELLO packet to 2% of all the data packets transmitted by the corresponding transmitter.

C. Rate Selection

The sampling mechanism reduces the search spaces only to select the appropriate MCS level either from the single stream or from the double stream. The other parameters like channel bonding, guard interval and frame aggregation get fixed during the sampling phase as binary (or distinct n-ary where n is very less: ex. $n = 3$ for channel bonding in IEEE 802.11ac) decision can be made over such parameters. As an example it can be noted that for IEEE 802.11n, the sampling mechanism reduces the search spaces from 256 possibilities to 8 possibilities (every stream supports 8 different MCS levels). At this level, the adaptive link management problem in HT wireless network reduces to conventional rate adaptation problem in wireless networks, where the data rate need to be selected, given that other features are fixed. Any standard rate adaptation mechanism can be used in this context. For this paper, we use `Minstrel` [16], the legacy rate adaptation mechanism for Linux operating system. Once other parameters are fixed, the proposed adaptive link management module executes `Minstrel` rate adaptation to find out the optimal data rate for that selection.

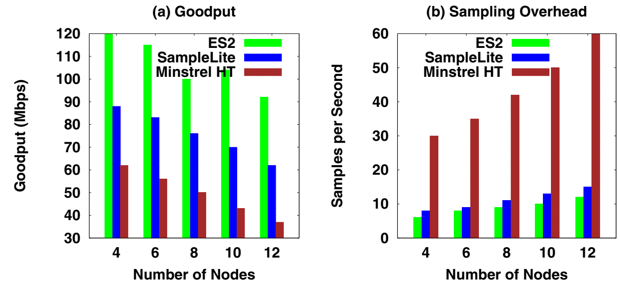

Fig. 8. Static Scenario (IEEE 802.11n)

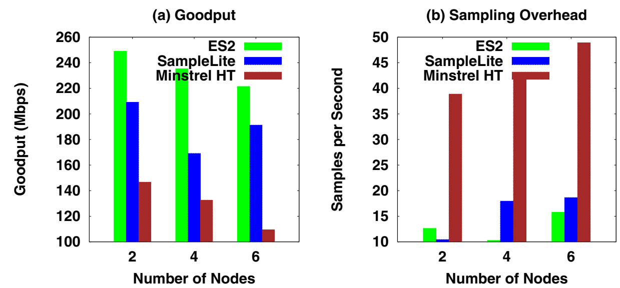

Fig. 9. Static Scenario (IEEE 802.11ac)

IV. Performance Analysis and Comparison

The proposed adaptive link management, *ES2*, is implemented over the testbed as discussed earlier, as a kernel submodule in mac80211 module. We have also implemented `Minstrel HT` [15] and SampleLite [6], two other link adaptation protocols for HT wireless networks. `Minstrel HT` is the default link adaptation protocol for IEEE 802.11 based HT MAC submodule of Linux kernel, however, it only considers MIMO spatial streams, MCS levels and channel bonding for adaptive link management. SampleLite is the recent and to the best our knowledge, the only link adaptation protocol that considers all the feature sets together. However, SampleLite does a transmitter side RSS calculation for sampling out the non-preferable options.

Two different network scenarios are considered in this paper

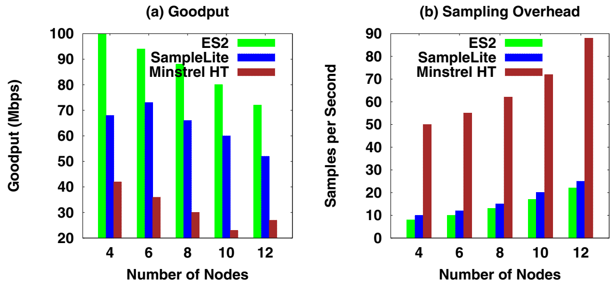

Fig. 10. Mobile Scenario (IEEE 802.11n)

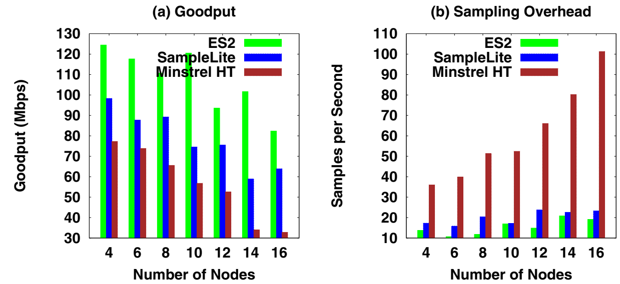

Fig. 14. Static Scenario (Mixed Testbed with Interoperability)

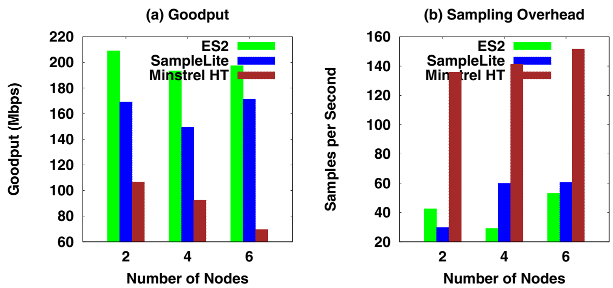

Fig. 11. Mobile Scenario (IEEE 802.11ac)

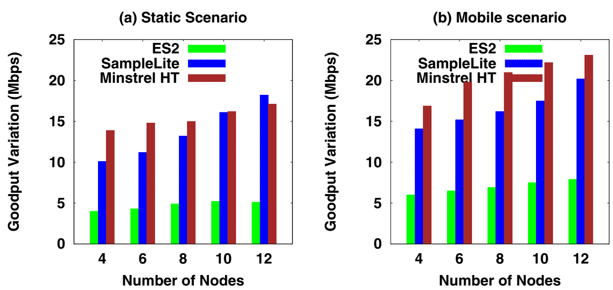

Fig. 12. Fairness: Average Link Goodput Variation (IEEE 802.11n)

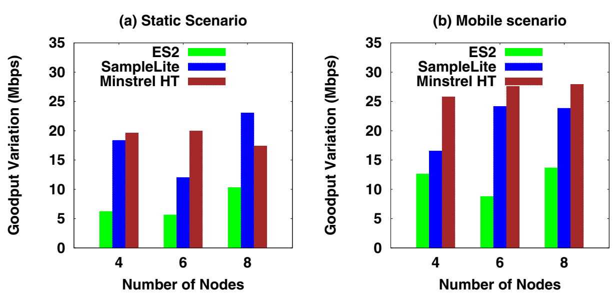

Fig. 13. Fairness: Average Link Goodput Variation (IEEE 802.11ac)

for performance evaluation - one static scenario and one mobile scenario. We consider both IEEE 802.11n and IEEE 802.11ac testbed. In the mobile network scenario, the nodes move at a rate of 1 m/s using random mobility model. We apply both UDP and TCP traffic at the application using `iperf` tool. The UDP traffic generation rate is 4 Mbps. The UDP and TCP traffic distribution in the network follows UDP:TCP = 30 : 70 ratio. Similarly the upload and download traffic follows upload:download = 40 : 60 ratio.

Fig. 8 and Fig. 9 show the goodput and sampling overhead in static scenario whereas Fig. 10 and Fig. 11 plot the same in the mobile network scenario. It can be noted that the total number of nodes include the HT wireless access points plus the client devices. The figures reveal that there is almost 60% performance improvement compared to SampleLite with a marginal reduction in sampling overhead. Although *ES2* and SampleLite both use similar sampling mechanism, however SampleLite relies on transmitter side RSS which can not predict the link quality correctly. In wireless networks, interference effects are more severe at the receiver. *ES2* uses a novel metric, called diffESNR, that predicts the link quality from receiver side feedback. As a consequence, the proposed *ES2* scheme shows around 60% performance improvement compared to SampleLite.

As discussed earlier, link fairness is another important metric that need to be considered during the link adaptation mechanism. We measure link fairness in terms of average goodput variation per link, that is plotted in Fig. 12 and Fig. 13 for the static and mobile network scenarios. The figures reveal that *ES2* significantly reduces per link goodput variation compared to SampleLite and `Minstrel HT`. The sampling metric, diffESNR, considers the effect of SNR variation which reduces the goodput variation, and in turns improve link fairness.

V. Discussion: *ES2* Operation in a Mixed IEEE 802.11n + IEEE 802.11ac Testbed

Next we show and analyze the results from a mixed network testbed with IEEE 802.11n and IEEE 802.11ac. We configure both the IEEE 802.11n and IEEE 802.11ac access points in 5 GHz channel in fixed channel operating mode. The results are shown in Fig. 14 and Fig. 15, for static and dynamic scenarios, respectively. Similar to the earlier scenario, we can observe a significant goodput improvement for the proposed *ES2* mechanism, with marginal sample overhead. It can be noted that in the mixed mode scenario, we have kept similar set-ups of channel bonding (20 MHz and 40 MHz) and MCS levels (MCS 0 to MCS 15) to make them compatible with each

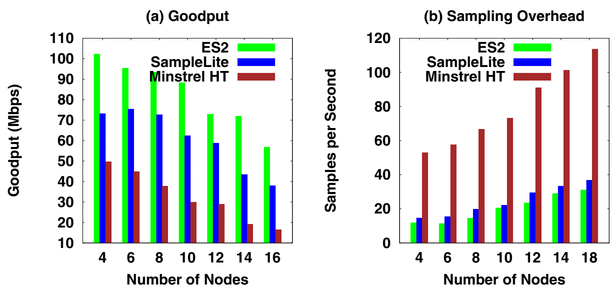

Fig. 15. Mobile Scenario (Mixed Testbed with Interoperability)

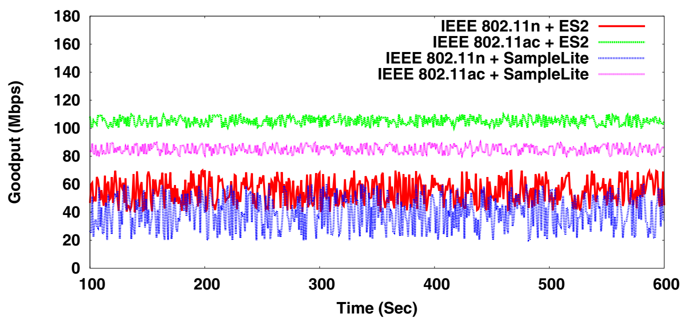

Fig. 16. Performance From IEEE 802.11n + IEEE 802.11ac Mixed Testbed

other. When the access points work in a non-coordinate mode (for example IEEE 802.11ac operates in 80 MHz and IEEE 802.11n works in 40 MHz), IEEE 802.11n nodes fail to detect the set-up for IEEE 802.11ac access points, and therefore the performance for IEEE 802.11n suffers.

We do an experiment over the mixed testbed scenario, as shown in Fig. 1, to analyze and observe the performance of IEEE 802.11n and IEEE 802.11ac access points, when they operate in different configuration set. In this setup, we allow all the available configurations for both the different variants of HT wireless. Fig. 16 shows the average goodput for IEEE 802.11n and IEEE 802.11ac, for *ES2* and SampleLite. The figure indicates that although *ES2* improves the performance for IEEE 802.11ac, many of the times the performance for IEEE 802.11n drops. We analyzed the trace, and observed that when IEEE 802.11ac operates in 80 MHz, which is not supported in IEEE 802.11n, the later fails to coordinate and filter out the non-preferable configurations. As a result, the performance for IEEE 802.11n drops. The heterogeneous mode operation of *ES2* is kept as a future extension of our work as it require a thorough analysis of interoperability.

VI. Conclusion

In this paper, we propose a novel link adaptation mechanism, called *ES2*, that considers different link configuration parameters in HT wireless networks. The proposed mechanism takes into account the effect of MIMO streaming, MCS selection, channel bonding, guard interval and frame aggregation to find out the optimal set of parameter configuration based on link quality assessment. Because of the large volume of feature set, we propose a sampling strategy based on a new metric, called diffESNR. The metric diffESNR is calculated from estimated SNR at the transmitter side, based on receiver feedback, with the help of Kalman filtering approach. The proposed scheme is implemented in a real testbed, and the analysis of the testbed result reveals that *ES2* significantly improves goodput and link fairness while using marginally less control overhead compared to SampleLite, the most recently proposed mechanism for adaptive link management.

References

[1] W. Sun, O. Lee, Y. Shin, S. Kim, C. Yang, H. Kim, and S. Choi, "Wi-Fi could be much more," *IEEE Communications Magazine*, vol. 52, no. 11, pp. 22–29, 2014.

[2] M. X. Gong, B. Hart, and S. Mao, "Advanced wireless LAN technologies: IEEE 802.11ac and beyond," *SIGMOBILE Mob. Comput. Commun. Rev.*, vol. 18, no. 4, pp. 48–52, Jan. 2015.

[3] D.-J. Deng, K.-C. Chen, and R.-S. Cheng, "IEEE 802.11ax: Next generation wireless local area networks," in *proceedings of QShine*, 2014, pp. 77–82.

[4] S. H. Y. Wong, H. Yang, S. Lu, and V. Bharghavan, "Robust rate adaptation for 802.11 wireless networks," in *Proceedings of the 12th Mobicom*, 2006, pp. 146–157.

[5] I. Pefkianakis, S.-B. Lee, and S. Lu, "Towards MIMO-aware 802.11n rate adaptation," *IEEE/ACM Trans. Netw.*, vol. 21, no. 3, pp. 692–705, 2013.

[6] L. Kriara and M. K. Marina, "SampleLite: A hybrid approach to 802.11n link adaptation," *ACM SIGCOMM Computer Communications Review*, vol. Online First, 2015.

[7] D. Nguyen and J. Garcia-Luna-Aceves, "A practical approach to rate adaptation for multi-antenna systems," in *proceedings of 19th ICNP*, 2011, pp. 331–340.

[8] L. Deek, E. Garcia-Villegas, E. Belding, S.-J. Lee, and K. Almeroth, "Joint rate and channel width adaptation for 802.11 MIMO wireless networks," in *proceedings of 10th SECON*, 2013, pp. 167–175.

[9] L. Kriara, M. K. Marina, and A. Farshad, "Characterization of 802.11n wireless LAN performance via testbed measurements and statistical analysis," in *proceedings of 10th SECON*, 2013, pp. 158–166.

[10] R. Combes, A. Proutiere, D. Yun, J. Ok, and Y. Yi, "Optimal rate sampling in 802.11 systems," in *Proceedings of IEEE INFOCOM*, 2014, pp. 2760–2767.

[11] S. Seytnazarov and Y.-T. Kim, "Cognitive rate adaptation for high throughput ieee 802.11n WLANs," in *proceedings of 15th APNOMS*, 2013, pp. 1–6.

[12] Z. Zhao, F. Zhang, S. Guo, X.-Y. Li, and J. Han, "RainbowRate: MIMO rate adaptation in 802.11n wild links," in *proceedings of IPCCC*, 2014, pp. 1–8.

[13] L. Deek, E. Garcia-Villegas, E. Belding, S.-J. Lee, and K. Almeroth, "The impact of channel bonding on 802.11n network management," in *Proceedings of the Seventh CoNEXT*, 2011, p. 11.

[14] S. Chakraborty and S. Nandi, "Controlling unfairness due to physical layer capture and channel bonding in 802.11n+s wireless mesh networks," in *Proceedings of ICDCN*, 2015, pp. 21:1–21:10.

[15] F. Fietkau and D. Smithies, "minstrel_ht: New rate control module for 802.11n," 2010. [Online]. Available: https://lwn.net/Articles/376765/

[16] W. Yin, P. Hu, J. Indulska, and K. Bialkowski, "Performance of mac80211 rate control mechanisms," in *Proceedings of the 14th ACM MSWiM*, 2011, pp. 427–436.

[17] X. Chen, P. Gangwal, and D. Qiao, "RAM: Rate adaptation in mobile environments," *IEEE Transactions on Mobile Computing*, vol. 11, no. 3, pp. 464–477, 2012.

[18] The MadWifi project, "RSSI in MadWifi," 2015. [Online]. Available: http://madwifi-project.org/wiki/UserDocs/RSSI

[19] M. Senel, K. Chintalapudi, D. Lal, A. Keshavarzian, and E. J. Coyle, "A Kalman filter based link quality estimation scheme for wireless sensor networks," in *Proceedings of IEEE GLOBECOM*, 2007, pp. 875–880.