RESEARCH-ARTICLE

Distributed Spectrum Sharing for Enterprise Powerline Communication Networks

KAMRAN ALI, General Motors, Detroit, MI, United States

ALEX X LIU, Qilu University of Technology, Jinan, Shandong, China

IOANNIS PEFKIANAKIS, Apple Computer, Cupertino, CA, United States

KYU-HAN KIM, HP Labs, Palo Alto, CA, United States

IEEE/ACM Transactions on Networking, Volume 29, Issue 3 (June 2021)
https://doi.org/10.1109/TNET.2021.3056512
EISSN: 1558-2566

Distributed Spectrum Sharing for Enterprise Powerline Communication Networks

Kamran Ali, Alex X. Liu, *Fellow, IEEE*, Ioannis Pefkianakis, and Kyu-Han Kim, *Senior Member, IEEE*

***Abstract*— As powerline communication (PLC) technology does not require dedicated cabling and network setup, it can be used to easily connect multitude of IoT devices deployed in enterprise environments for sensing and control related applications. IEEE has standardized the PLC protocol in IEEE 1901, also known as HomePlug AV (HPAV) which has been widely adopted in mainstream PLC devices. A key weakness of HPAV protocol is that it does not support spectrum sharing. Currently, each link in an HPAV PLC network operates over the whole available spectrum, and only one link can operate at any time within a single collision domain. In this work, through an extensive measurement study of HPAV PLCs in a real enterprise environment using commodity off-the-shelf (COTS) HPAV PLC devices, we discover that spectrum sharing can significantly benefit enterprise level PLC networks. To this end, we propose a distributed spectrum sharing technique for enterprise HPAV PLC networks, and show that fine-grained distributed spectrum sharing on top of current HPAV MAC protocols can significantly boost the aggregated and per-link throughput, by allowing multiple PLC links to communicate concurrently, while requiring only a few modifications to the existing HPAV devices and protocols.**

***Index Terms*— Computer networks, Internet of Things, channel state information, distributed algorithms.**

I. Introduction

As POWERLINE communication (PLC) technology does not require dedicated cabling and network setup, it can be used to easily connect multitude of IoT devices deployed in enterprise environments for sensing and control related applications. Thanks to the *plug-n-play* nature of PLC technology, a PLC enabled device just needs to be connected to a wall socket, and it will automatically form a mesh network with nearby PLC devices. IEEE has standardized the PLC protocol in IEEE 1901, also known as HomePlug AV (HPAV) [1], [2], which has been widely adopted in mainstream PLC devices.

A key weakness of HPAV protocol is that it does not support spectrum sharing. Currently, each link in an HPAV PLC network operates over the whole available spectrum, and only one link can operate at any time within a single collision domain. Figure 1 shows an example enterprise level IoT application scenario, where multiple PLC nodes (including multiple gateway nodes) are connected in the same MAC collision domain to a power distribution network. Currently, two *disjoint* PLC links (*e.g.* 5-8 and 12-11 in Fig. 1) cannot operate concurrently with existing HPAV MAC protocols. However, in real enterprise PLC deployments, we often encounter scenarios where a subset of subcarriers on some PLC links are highly underutilized as compared to other links, which implies that the low-modulated subcarriers of one PLC link can be utilized by one of the other links to improve the aggregated throughput. Moreover, if multiple PLC links, which may be competing for the same channel simultaneously, can operate in parallel via sharing spectrum, many costly collisions can be avoided.

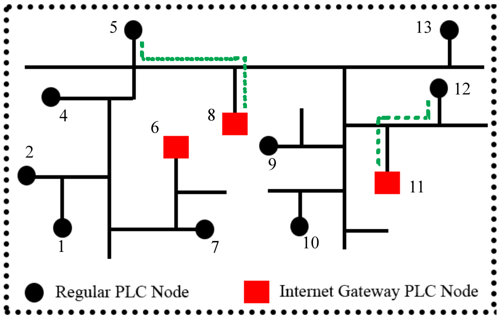


Fig. 1. Example scenario: Links 5-8 and 12-11 in the same collision domain can share spectrum for concurrent operation.

In this work, through an extensive measurement study of HPAV PLCs in a real enterprise environment using commodity off-the-shelf (COTS) HPAV PLC devices, we discover that spectrum sharing can significantly benefit enterprise level PLC networks. Our first finding is that PLC nodes connected under the same circuit breaker in a building's power distribution network can communicate at 6.5 times higher throughput than the PLC nodes connected under two different breakers, and 18-30 times higher throughput than the PLC nodes connected to two completely different power distribution/trunk lines. This implies that enterprise PLC networks must have at least one gateway node connected under every breaker, to provide best possible connectivity to the IoT devices connected under that breaker. As each power distribution line can contain tens to hundreds of breakers with multitude of IoT devices connected to a gateway under each breaker, the number of *disjoint links*, which consist of different source-destination pairs and may compete for the same channel simultaneously, becomes significant. Second, based on our subcarrier level spectral analysis, we observe that PLC channels of more than 50% of

Manuscript received April 3, 2019; revised June 27, 2020 and November 4, 2020; accepted January 18, 2021; approved by IEEE/ACM Transactions on Networking Editor G. Zussman. Date of publication February 11, 2021; date of current version June 16, 2021. This work was supported by the HP Labs research grant. The preliminary version of this article titled "Distributed Spectrum Sharing for Enterprise Powerline Communication Networks" was published in proceedings of the IEEE International Conference on Network Protocols (ICNP), pages 367–377, Cambridge, U.K., September, 2018. *(Corresponding author: Alex X. Liu.)*

Kamran Ali is with General Motors Research and Development, Warren, MI 48557 USA (e-mail: kamran.luminite@gmail.com).

Alex X. Liu is with the Qilu University of Technology, Jinan 250353, China (e-mail: alexliu360@gmail.com).

Ioannis Pefkianakis is with Apple, Cupertino, CA 95014 USA (e-mail: yiannis.ucla@gmail.com).

Kyu-Han Kim is with the Networking and Mobility Group, Hewlett Packard Labs, Palo Alto, CA 94304 USA (e-mail: kyu-han.kim@hpe.com).

Digital Object Identifier 10.1109/TNET.2021.3056512

the PLC links are significantly different from each other due to highly location dependent multipath characteristics. As the performance of different frequency subcarriers varies among different PLC links, low-modulated subcarriers of one link can be utilized by other links, and vice versa. Third, most links in an enterprise PLC network are *pseudo-stationary*, *i.e.* the channel characteristics between any two PLC nodes have low temporal variability (standard deviation of throughput observed over 15 minute time windows is below 2.2Mbps for more than 80% of the links), and therefore, a spectrum sharing scheme can be achieved at low channel estimation related control overhead.

Multiple Frequency Division Multiplexing (FDM) based spectrum sharing techniques have been proposed for PLCs [3], [4]. However, such spectrum sharing techniques have three major limitations. First, they are incompatible with the HPAV MAC, which makes them difficult to be adopted. Second, they are designed for WiFi like point-to-multipoint communications. This may be suitable for home PLC networks where a few IoT devices are connected to a single PLC gateway node. However, it is unsuitable for enterprise PLC environments, as enterprise PLC networks are mesh networks, with multitude of disjoint links between IoT devices and their respective gateway nodes. Third, they have prohibitively high computational and control overheads involved in their underlying subcarrier assignment and bit loading algorithms. This makes them impractical for real world deployment.

In this work, we aim to design a spectrum sharing scheme which is compatible with HPAV MAC, is suitable for enterprise level PLC mesh networks, and incurs minimal computational and control overheads. To this end, we propose an HPAV compatible, distributed, low overhead spectrum sharing approach for enterprise PLC networks. Currently, HPAV MAC protocol uses Carrier-Sense Multiple Access with Collision Avoidance (CSMA/CA) and Time-Division Multiple Access (TDMA) techniques for sharing medium access among PLC nodes. To make our scheme compatible with existing HPAV MAC, we design it such that any link which occupies the PLC channel following the regular HPAV CSMA/CA or TDMA protocol shares a part of its spectrum with another link to improve the aggregated throughput of both links. Moreover, we design our scheme such that it can be enabled in the current HPAV PLC devices while incurring minimum firmware level changes. We call the links which occupy the PLC channel following regular HPAV CSMA/CA or TDMA protocol as *primary links*, and the links with which the primary links share their spectrum as *secondary links*. To make our scheme suitable for enterprise level PLC mesh networks, we develop a distributed spectrum sharing strategy. To achieve this, we develop a spectrum sharing algorithm which each node uses to locally compute a complete set of network-wide spectrum sharing rules for all possible *primary* links and their corresponding *secondary* links in the network. Our algorithm leverages subcarrier level channel information corresponding to all possible links in the network to compute those rules. Based on these rules, any *primary* link can decide which of the possible *secondary* links should it share its spectrum with, and what part of the spectrum should it share, to achieve best possible spectrum sharing gains. When the source node of a *primary* PLC link gets channel access, it broadcasts the link's source-destination IDs to all the remaining nodes in its network. Next, it picks one of the possible *secondary* links to share its spectrum with, based on its locally computed network-wide spectrum sharing rules, and then continues its remaining transmission in the unshared region of spectrum. Meanwhile, the source and destination nodes of the chosen *secondary* link establish connection, and start operating in parallel with the *primary* link over the shared region of spectrum. This happens automatically, as both source and destination nodes of the chosen *secondary* link already know the source-destination IDs of the *primary* link (because of the primary link's source-destination IDs broadcast by its source node to all other nodes) and have the same set of network-wide spectrum sharing rules. Transmission of *secondary* link finishes as soon as the *primary* link finishes its transmission. To minimize the computational and control overhead of our scheme, we take the following design decisions: First, instead of aiming for a globally optimal solution, we develop a sub-optimal spectrum sharing solution such that the basic optimization problem which it solves comes down to optimally sharing spectrum between just two links (*i.e.*, a *primary* and a *secondary*), which is a computationally simpler problem to solve than sharing spectrum with several links simultaneously. This increases the scalability and practicality of our solution while still achieving significant benefits of spectrum sharing. Second, we design our scheme to operate in a distributed manner, in the sense that each node locally computes network-wide spectrum sharing rules. This makes real-time spectrum sharing seamless, as it completely avoids any extra control related communications for coordinating spectrum sharing in the network. Third, our design takes advantage of the *pseudo-stationary* nature of enterprise PLC channels to reduce channel estimation related overhead. The computation of network-wide spectrum sharing rules at each node requires latest subcarrier level channel information of all possible links in the network. To achieve this, each node first gets channel information corresponding to all possible links it can form, and then shares that information with other nodes in the network, which can involve considerable communication overhead. However, as most PLC channels in an enterprise setting are *pseudo-stationary*, PLC nodes do not need to update their copy of network-wide channel information too frequently. Therefore, the frequency of channel probing is significantly reduced, which maintains the spectrum sharing gains.

We implement and evaluate our proposed spectrum sharing techniques on HPAV CSMA protocol only, as the integration of our spectrum sharing technique with HPAV TDMA protocol is relatively straightforward to achieve (we present a detailed discussion on this in § VI). We perform trace driven simulations using channel response (*tonemap*) traces collected from seven different 4-node PLC deployments. We show that fine-grained distributed spectrum sharing can boost the aggregated and per-link throughput by more than 60% and 250% respectively.

II. Related Work

Hayasaki *et al.* [4] and Achaichia *et al.* [3] have proposed FDM based multiple access techniques in the context of point-to-multipoint communication in PLC networks. Hayasaki *et al.* [4] proposed a theoretical bit-loading based OFDMA scheme for in-home PLCs, as an alternative to TDMA/CSMA based medium access. Their scheme consists of two iterative algorithms: a subcarrier assignment algorithm and a bit-loading algorithm. The subcarrier assignment algorithm assigns subcarriers to maximize the whole throughput, while satisfying the minimum throughput guarantees of each destination PLC node first. Afterwards, the bit-loading algorithm is utilized for loading bits into the assigned subcarriers, while optimizing both the bit quantity on each subcarrier as well as the whole code rate, subject to BER constraints on each subcarrier. Achaichia *et al.* [3] proposed a similar technique named *Tone Maps Splitting Algorithm* (TMSA) to orthogonalize spectrum assigned to multiple active links in a point-to-multipoint communication. However, the aforementioned techniques are designed for WiFi like point-to-multipoint communication in PLCs, and proposed as an alternative MAC protocol to existing HPAV TDMA/CSMA based MAC. Moreover, the aforementioned techniques involve high computational and control overheads corresponding to their underlying resource allocation schemes (i.e. subcarrier assignment and bit loading algorithms), which makes them impractical for real world deployment scenarios.

In [5], [6] authors compare HPAV with WiFi performance. They study temporal and spatial variations of the throughput of PLC links and make a case for hybrid PLC-WiFi networks [5]. The measurement study in [7] shows the multi-flow performance of PLC networks. It then presents BOLT, which seeks to manage traffic flows in PLC networks. The above studies ignore the spectral inefficiencies at MAC layer of HPAV networks. In contrast, we extensively study the behavior of PLCs in spatial, temporal and spectral dimensions, and propose novel spectrum sharing strategies to improve per-link and aggregated throughput of enterprise level PLC networks.

III. HomePlug AV Powerline Communications

A. PLC Channel Characteristics

Multipath is a key characteristic of PLC channels, which is attributed to unmatched electric loads or branch circuits connected to different sockets on the powerline. In a typical power distribution network of a large building, there are multiple branch circuits with different impedances, and therefore, PLC signals are reflected from multiple reflection points leading to multipath effects. On top of multipath attenuations, several different types of noise in PLC channels have been identified [8], [9]. Harmonics of AC mains and other low power noise sources in the power lines lead to colored background noise, which decreases with frequency. Periodic impulsive noise is created due to rectifiers, switching power supplies and AC/DC converters, which can be either synchronous or asynchronous with AC line cycle. Aperiodic impulsive noise also exists in PLC channels due to switching transients in power supplies, AC/DC converters, etc.

B. HomePlug AV Standard

The most widely adopted family of PLC standards are HomePlug AV, AV2 and Green PHY standards [10]. HomePlug AV2, which is the latest of these standards, can support up to 1 Gbps PHY rates. Our study focuses on the HomePlug AV standard, which has been widely used in home networks to improve coverage, and can support maximum PHY rates of up to 200 Mbps [1], [2]. However, our findings and solutions can also be generalized for PLC technologies other than HPAV, such as HPAV2.

HPAV PHY-Layer: HPAV uses 1.8-30 MHz frequency band and employs OFDM with 917 subcarriers (for the USA devices), where each subcarrier can use any modulation scheme from BPSK to 1024-QAM depending on the channel conditions [10]. In order to update the modulation schemes for each subcarrier, two communicating HPAV PLC devices continuously exchange and maintain *tonemaps* between them. Tonemaps refer to the information about the modulation scheme used per subcarrier, i.e. the number of bits modulated per subcarrier. The tonemaps exchanged are estimated for multiple different sub-intervals of the AC mains cycle. Tonemaps are exchanged between PLC devices through a *sounding* process, where the transmitter sends sounding frames to the receiver using QPSK for all subcarriers, the destination estimates the channel quality and sends back the tonemaps corresponding to different sub-intervals of AC mains cycle back to the transmitter. The destination can communicate up to 7 tonemaps, i.e. 6 tonemaps for the different sub-intervals of the AC line cycle called slots and one default tonemap [10], depending on the condition of noise and attenuation observed in different parts of AC line cycle. Tonemaps are continuously updated by default after 30 seconds or when the error rate exceeds a threshold [10]. *Tonemaps provide us with the information about Channel Frequency Response (CFR) of the channel between two communicating PLC devices.*

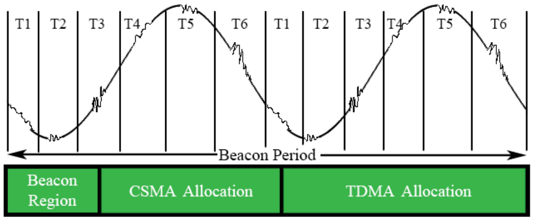


Fig. 2. Basic Beacon Period structure in HPAV MAC.

HPAV MAC-Layer: MAC-layer of HPAV based PLCs works very differently from that of WiFi MAC. First, unlike WiFi, *channelization* is not allowed and not used in PLCs, which limits the possibility of deploying non-interfering networks, such as WiFi networks on different channels. Second, there is no concept of a central *Access Point* (AP) in PLCs. There exists a dynamically chosen central authority to manage network, called the *Central Coordinator* or CCo, and large PLC networks can contain multiple CCo's managing their own collision domains. However, CCo's role is passive, mainly authentication and association of new nodes, monitoring the network, synchronizing it with the AC line cycle, and taking time-division access decisions in terms of allocating TDMA and CSMA/CA slots. In contrast, the WiFi AP-mode forces downlink/uplink traffic types and a star-like logical network. PLCs only form mesh networks, and every node can communicate with its peers, without relaying through the CCo. Both TDMA and CSMA/CA are supported by HPAV [10]. Tonemaps are optimized for the QoS required for the traffic in the TDMA allocations. HPAV uses a Beacon Period, managed by a CCo, for allocating CSMA and TDMA sessions (Fig. 2). The Beacon Period is synchronized with AC line cycle and is two AC line cycles in length. The schedules advertised in the Beacon are persistent and are not changed for a number of Beacon Periods. The CSMA protocol of HPAV devices (IEEE 1901) is different from CSMA/CA used by WiFi devices (IEEE 802.11). Both use a time-slotted random backoff with a *backoff counter* (BC) and a *contention window* (CW). 1901 includes two more counters in its backoff procedure, i.e., *backoff procedure counter* (BPC) and the *deferral counter* (DC). Using DC based backoff procedure, HPAV PLC nodes increase their contention windows not only after a collision, but also after sensing the medium to be busy, which reduces the probability of collision. Details of the 1901 backoff procedure can be found in [11]. HPAV frames are 512 byte aggregated physical blocks (PBs) of data. In order to reduce protocol overheads, HPAV employs two-level frame aggregation, where the First, the data is organized in 512 byte physical blocks (PB). PBs are then aggregated into HPAV frames. Reception of each PB of a frame is separately acknowledged, so that the transmitter retransmits only the corrupted PBs.

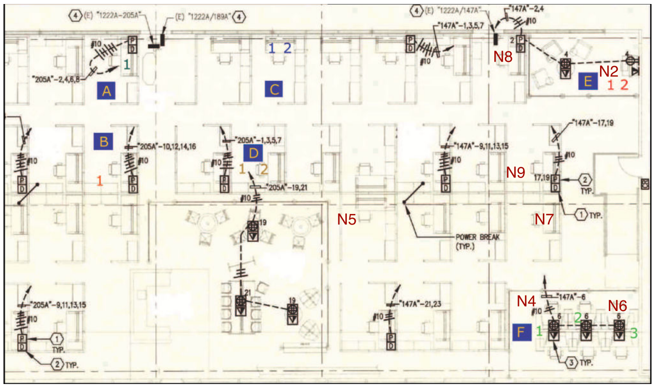


Fig. 3. Building power distribution plan.

IV. A Measurement Study of Enterprise PLCs

Experimental Setup: Our study is based on measurements with commodity HomePlug AV hardware. We use Meconet HomePlug AV mini-PCI adapters with Intellon INT6300 chipsets, which can support 200 Mbps PHY rates. We connect the PLC adapters to ALIX 2D2 boards, which run OpenWrt operating system. We use open source PLC software tool named *open-plc-utils*, which is developed by Qualcomm, to extract PHY and MAC-layer feedback (such as tonemaps), directly from the Meconet HPAV adapters. Note that the measurements, analysis and solutions proposed in this paper can also be applied to newer HomePlug AV2 devices with a few modifications.

Experimental Methodology: For our experiments we place our PLC nodes in various locations of an enterprise building. We generate saturated *iperf* UDP traffic among the PLC nodes. Results we report are averaged over multiple runs.

Metrics: We analyze the performance of PLC networks by first collecting *iperf* throughput statistics. We further elaborate on the per-subcarrier PLC network performance by analyzing the tonemaps extracted by the *open-plc-utils* tool running on PLC nodes. For a given PLC communication link and for the k^{th} sub-interval of AC line cycle, the effective PHY rate can be estimated from tonemaps as $R_{phy}^{\{k\}} = \frac{[\sum_{j=1}^{N} T[j]^{\{k\}}]\cdot C^{\{k\}}\cdot(1-B_{err}^{\{k\}})}{T_s}$ [12], where j is subcarrier number and N is total number of subcarriers. $T[j]$ is the modulation rate (i.e., bits per subcarrier) of the j^{th} subcarrier [12]. C is Forward Error Correction (FEC) code rate. HomePlug AV supports FEC code rates of 1/2 and 16/21. Finally, B_{err} is the bit error rate and T_s is the symbol interval of OFDM communication. T_s is approximately $\sim$46μs for HomePlug AV including all overheads [10]. The expected throughput, averaged over all the sub-intervals of the AC line cycle, can be written as $\mathcal{T} \approx (1 - F_o) \cdot \sum_{k=1}^{N_{AC}} R_{phy}^{\{k\}}/N_{AC}$. Here F_o accounts for HPAV protocol overheads and N_{AC} is the number of sub-intervals of AC line cycle. N_{AC} is 5 or 6 for USA frequencies and F_o is typically $\sim$ 0.4 based on *iperf* throughput measurements. In all our experiments, we observed that the FEC code rate was always 16/21 for the communication among our HPAV devices. Therefore, we assume FEC code rate of 16/21 in rest of the paper, unless explicitly mentioned otherwise.

A. Many Disjoint PLC Links Compete for Channel Access

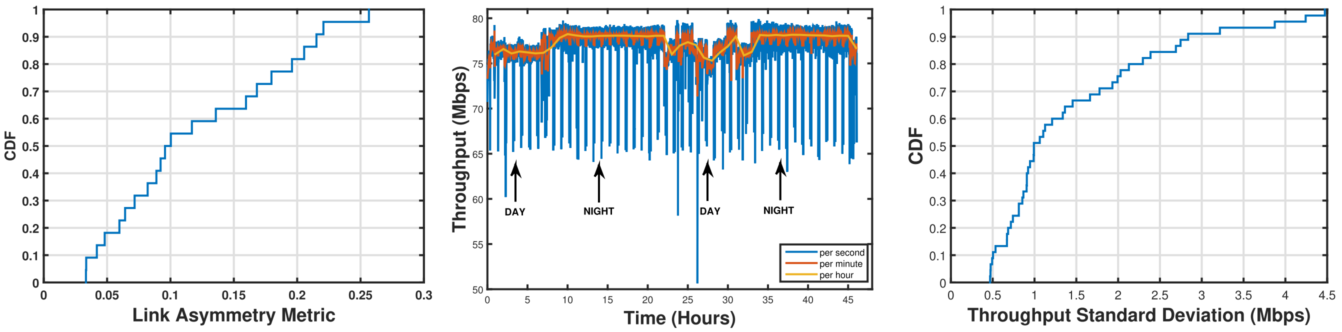
To understand how significant the number of *disjoint links* can become in a an enterprise level PLC network for IoT applications, we study the impact of different components of a power distribution network (*e.g.* phases, breakers and distribution/trunk lines) by measuring the throughput performance of more than 40 links (PLC transmitter-receiver pairs). Power distribution network floorplan of the enterprise building, where we conducted our experiments, is shown in Figure 3. The main switchboard of the enterprise, steps down the voltage from thousands to hundreds of Volts and the down converted electric power is then distributed towards different floors of different buildings in the enterprise, through multiple different distribution lines or *trunk lines* [13] (represented with hexagonal boxes with #4 written on them). The power from the trunk lines coming into the floor of a building is then further distributed into different parts of the floor, through a distribution board containing a set of circuit breakers which divide the electrical power feed into subsidiary circuits. Each trunk line consists of 3 cables corresponding to 3 different phases and each distribution board contains multiple breakers per phase. The letters (A-E) and numbers in the floorpan of Figure 3 represents some of the different locations where we placed our PLC nodes. Next, we elaborate on our experiments.

Case 1: We observed that the performance of a PLC link operating on same breaker and same distribution line is mainly affected by the location of PLC nodes with respect to the interfering electrical appliances. Highly attenuating device impedances or severe device interferences can lead to significant performance degradation (we observe $\sim$6.5 fold decrease in throughput). Moreover, as shown by the CDF in Fig 5, throughputs of more than 70 Mbps were observed across the tested links approximately 75% of the time. Jitter was low, with the median being 0.2 ms and a maximum of 2.5 ms.

Case 2: PLC nodes connected to same (or different) phase but different breakers operate over lower throughputs as compared to same phase, same breaker case ($\sim$20-30% decrease in observed throughput).[1] This is because signals experience higher attenuations while passing through the breaker circuitry located between the PLC nodes. We observed maximum throughput of 63 Mbps, which is 25.6 Mbps lower (29% decrease) than the previous case where nodes were connected under the same breaker. The median throughput observed was 51 Mbps, with minimum being 26 Mbps, which is higher than the minimum of same breaker case as we did not encounter any high interference from electric appliances this case.

Case 3: PLC performance significantly drops ($\sim$18-30 folds throughput decrease) when nodes are located at different distribution lines. Distribution lines can make PLC connectivity

[1] We have excluded the cases of high interference from electric devices.



(a) Link asymmetry. (b) PLC throughput timeseries (48 hours). (c) Link stability.

Fig. 4. (a) Link asymmetry, (b) Temporal variation in throughput over 2 days, (c) Link throughput stability CDF (45 links).

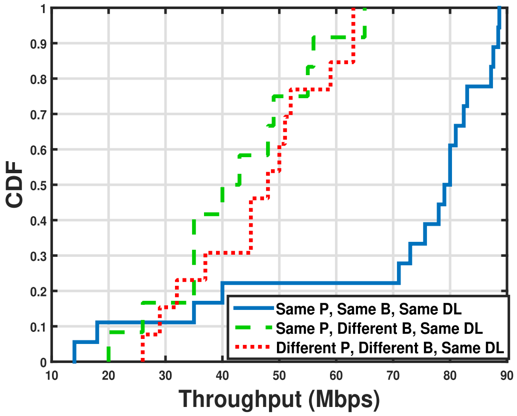


Fig. 5. CDF of throughputs observed in different cases.

often impossible, due to transformers in between. The maximum throughput that we observed between any two pair of nodes was 3 Mbps and 5 Mbps for both directions, and the jitter varied between 2.03 ms and 5.7 ms.

Conclusions: PLC nodes connected under the same breaker in a building's power distribution network were observed to communicate at 6.5 times higher throughput than the PLC nodes connected under two different breakers, and 18-30 times higher throughput than the PLC nodes connected to two completely different power distribution lines. This implies that enterprise PLC networks must have at least one gateway node connected under every breaker, to provide best possible connectivity to the IoT devices connected under that breaker. As each power distribution line can contain tens to hundreds of breakers with multitude of IoT devices connected to a gateway under each breaker, the number of *disjoint links*, which consist of different source-destination pairs and may compete for the same channel simultaneously, becomes significant.

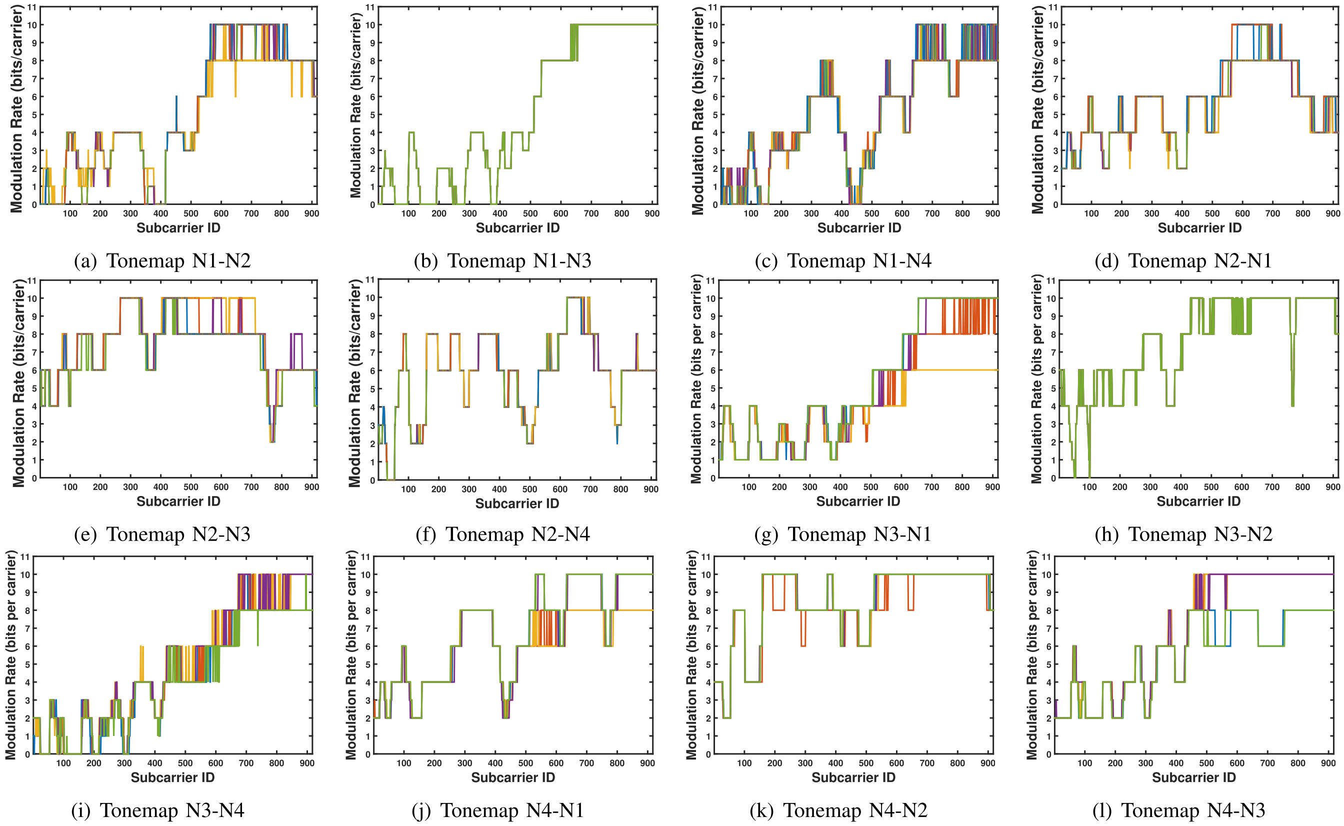
B. Enterprise PLC Channels Are Pseudo-Stationary

Performance of PLCs in enterprise settings can be dynamic either due to interference from already connected appliances, or due to a multitude of electrical devices being turned on/off on a regular basis. In order to study temporal dynamics, we measure performance of a PLC link for a long time periods. Figure 4(b) shows a representative scenario of a PLC link throughput variation, for 2 days (48 hours) period. The throughput variations are averaged over one second, one minute and one hour time windows respectively. We observe that the throughput performance can vary from 52 Mbps to 80 Mbps. The link appears to be highly bursty, which shows that some intense performance dynamics happening at small time scales, which are attributed to interference created by nearby electrical devices. The throughput variations observed at coarser time scales (minutes or hours) are attributed to human activity (e.g. connection/disconnection of new devices, etc.). The analysis of tonemaps (not shown here) also verifies the link variations with time, as we observed that the tonemaps exchanged among PLC nodes during day were different from those during night. However, we observed that throughput between most PLC links remained quite stable. Figure 4(c) shows the CDF plot of standard deviation (averaged over 10 second intervals) of the real time throughput of 45 different links we tested in our building. Throughput for each link was collected over 15 minute time windows. It can be observed that more than 60% of the time, the standard deviation of throughput is below 1.5 Mbps, which shows that throughput performance of most PLC links remains consistent over time.

Conclusion: Most links in an enterprise PLC network are *pseudo-stationary*, i.e. the channel characteristics between any two PLC nodes have low temporal variability. Therefore, a spectrum sharing can be realized at low control overheads.

C. Enterprise PLC Channels Are Highly Location Dependent

We measure the intensity of location dependence of PLC channels through a PLC *link asymmetry* metric $\mathcal{A}_{a,b}$. Asymmetry of a PLC link depends on channel frequency response or transfer function between PLC nodes communicating over that link, and it can be directly attributed to the different multipath characteristics of the powerline, which can vary depending on the location of PLC nodes compared to branch circuits or other connected electrical devices [14]–[16] (*i.e.* location dependent multipath characteristics). We quantify asymmetry of a PLC link $a - b$ as $[\mathcal{A}_{a,b} = \sum_{k=1}^{N_{AC}} [\sum_{j=1}^{N} |T_{a\rightarrow b}[j]^{\{k\}} - T_{b\rightarrow a}[j]^{\{k\}}|]]/N_{AC}$, where N is the number of subcarriers, T_j is the modulation rate of the j^{th} subcarrier and N_{AC} is the number of sub-intervals of AC line cycle. The above equation estimates asymmetry between two links as the distance between tonemaps of these links, averaged over all AC line cycle sub-intervals. The max and min values for $A_{a,b}$ are 9170 (917 subcarriers × 10 bits/carrier) and 0, respectively. In Figure 4(a) we present the distribution of our link asymmetry metric $A_{a,b}$ normalized by the maximum $A_{a,b}$ (which is 9170), from the tonemaps of 25 pair of nodes in



(a) Tonemap N1-N2 (b) Tonemap N1-N3 (c) Tonemap N1-N4 (d) Tonemap N2-N1

(e) Tonemap N2-N3 (f) Tonemap N2-N4 (g) Tonemap N3-N1 (h) Tonemap N3-N2

(i) Tonemap N3-N4 (j) Tonemap N4-N1 (k) Tonemap N4-N2 (l) Tonemap N4-N3

Fig. 6. Tonemaps of 12 links among 4 PLC nodes in one of our PLC deployments, showing possibility of gains from SS. The colored lines in each graph represent information corresponding to different AC line cycle slots.

the same neighborhood a, b. We observe that for more than 50% of the links, the normalized $A_{a,b}$ is greater than 0.1 (917 bits). The maximum throughput difference observed in asymmetric links is 15 Mbps. Note that when normalizing, our goal is to simply divide the asymmetry measure by a constant factor. However, we can also use "intersection over union" like metrics to interpret asymmetry more intuitively, e.g. by defining the asymmetry between two links as $2|T_{a\to b} - T_{b\to a}|/(T_{a\to b} + T_{b\to a})$.

Figure 6 shows snapshots of the tonemaps of 12 different links from a real world scenario, where we deployed a network of 4 PLC nodes in our test environment. We observe that the same subcarriers perform differently for different links. If we consider the last 200 subcarriers (717-917) for all the links of node N1, we observe the modulation is at least 6 bits per carrier (cf. Figures 6(a), 6(b), 6(c)). On the other hand, the last 200 subcarriers for all the links of node N2, show lower modulation, which can be as low as 2 bits per carrier (cf. Figures 6(d), 6(e), 6(f)). The modulations of N2's links, for the first 100 subcarriers (e.g. 1-100), are overall better compared to the corresponding modulations of N1's links. A spectrum sharing strategy could allow both N1 and N2 to transmit at the same time to their neighbors (e.g. N1-N3 and N2-N4) using only their high-performance subcarriers. Similar observations hold for other links (tonemaps not shown here), where certain subcarriers cannot carry data (0 modulation) and others can allow high modulations.

Conclusion: Per-subcarrier performance can vary significantly among different links in enterprise PLC networks. Therefore, the low-modulated subcarriers of one PLC link can be utilized by other PLC links, and vice versa.

V. Distributed Spectrum Sharing for HPAV PLCs

A. Preliminary Definitions

Primary & Secondary Links: We call the links which occupy the PLC channel through regular HPAV CSMA/CA or TDMA protocol as *primary* (P-Link or $p_{i\to j}$), and the links with which a primary link shares spectrum with, as *secondary* (S-Link or $s_{m\to n}$). Whenever a P-Link is established, only one S-Link can operate during that communication slot. For example, if we assume that all links are saturated (i.e., each node always has traffic to send), the S-Link which gives maximum possible gain by sharing spectrum with an established P-Link will operate in parallel with that P-Link. Later on, we will present a ranking based strategy which each node in the network can follow locally to resolve contention for S-Link.

Tonemaps: Let $[T_{i\to j}]^p_{1\times N}$ and $[T_{m\to n}]^s_{1\times N}$ be the vector of tonemaps of a pair of P-Link ($i \to j$) and S-Link ($m \to n$), respectively. The difference between tonemap vectors of P-Link and S-Link can then be denoted by $[D_{i\to j,m\to n}]_{1\times N} = [T_{i\to j}]^p_{1\times N} - [T_{m\to n}]^s_{1\times N}$, and vice versa.

Minimum Throughput Requirement: Let us denote the number of PLC nodes in a network to be E. Moreover, let us denote the minimum throughput requirement of e-th node as $\mathcal{T}_e$. Then we can represent the minimum number of bits to be modulated across a given set of OFDM subcarriers (tonemap), required to meet $\mathcal{T}_e$ as $\tau_e = \frac{\mathcal{T}_e}{T_s}$. Note that for any P-Link ($i \to j$) and S-Link ($m \to n$) pair, our SS strategy needs to meet throughput requirement of the P-Link only.

Allowed Tonemaps & Link Ranks: Each node e in the network will locally calculate an SS matrix using the proposed

SS algorithm. The entries of the SS matrix consist of two entities, namely *allowed tonemaps* - $[ST, PT]$ (where $[ST]$ is a set of subcarriers allowed to be modulated on an S-Link while a P-Link operates, and it is vice versa for $[PT]$), and *link ranks* - r (i.e. a rank proportional to the SS gain of an S-Link when sharing spectrum with a P-Link). For each node e, there are $2 \times (E-1)(E-2)$ possible P-Links which can operate in its vicinity. Moreover, for each of those possible P-Links, there are $2 \times (E-2)(E-3)$ possible S-Links which can operate in parallel. Therefore, a locally computed SS matrix at any node will have $4 \times (E-1)(E-2) \times (E-2)(E-3)$ elements, where each row will correspond to the allowed tonemaps and ranks of all possible S-Links corresponding to one of the $2 \times (E-1)(E-2)$ possible P-Links.

Spectrum Sharing Gain: We represent the gain $G_{m\rightarrow n}$ obtained by allowing an S-Link to operate with a P-Link as:

$$G_{m\rightarrow n} = [\sum_{[ST]}[T_{m\rightarrow n}] + \sum_{[PT]}[T_{i\rightarrow j}]] - \sum_{[1,N]}[T_{i\rightarrow j}] \qquad (1)$$

B. Spectrum Sharing (SS) Algorithm

We design our SS algorithm to meet two key requirements: **(a)** *It must take into account minimum throughput requirements of the destination node of each possible P-Link in the network*, and **(b)** *It should involve minimal channel probing and control overhead.* As our SS approach is designed to work on top of existing user scheduling provided by current HPAV/AV2 CSMA/CA or TDMA procedures, therefore, SS is performed only when a P-Link is established and is already operating. Our SS algorithm runs locally at each node of the network, and therefore assumes that each node in the network has complete tonemap information about all other possible links in the network. Later in section VI, we explain how this can be achieved using current HPAV protocols, while incurring minimal channel probing and control overhead. Moreover, for simplicity of our discussion, we assume that all nodes in the network are in the same collision domain, and that there are no hidden nodes in the network. RTS/CTS delimiters and proxy nodes based *immediate repeating* in regular HPAV MAC can be used to handle hidden nodes during TDMA and CSMA/CA [10]. Although we believe that our scheme can be adapted accordingly, yet hidden nodes aware spectrum sharing is not the scope of this paper.

1) Spectrum Sharing (SS) Algorithm: Our SS algorithm is described in Algorithm 1. The algorithm runs on each node of the network separately, and computes $4 \times (E-1)$ $(E-2) \times (E-2)(E-3)$ *allowed tonemaps* - $[ST]$'s for all S-Links, corresponding to all possible P-Links. To describe in words, for each P-Link and S-Link pair, the above algorithm first sorts the difference vector $D_{i\rightarrow j,m\rightarrow n}$, and starts assigning the subcarriers to P-Link corresponding to descending order of the entries $\widetilde{D}_{i\rightarrow j,m\rightarrow n}$, until its minimum throughput requirement is met. The remaining subcarriers are then assigned to the S-Link. Although, we did not observe such cases in our deployments, but in practice, a PLC network can may contain some extremely bad P-Links (modulation of all subcarriers is very low). Following the aforementioned algorithm, *bad P-Links would only share their spectrum once their own throughput requirements are met, and therefore, will not starve.* Next, we discuss how this SS approach can be optimized for overall network throughput and fairness in spectrum, respectively.

Algorithm 1 Spectrum Sharing (SS) Algorithm

```
1: /*Takes in a set of tonemaps for all possible links*/
2: procedure GetAllowedTonemaps_SS($[T]^E$)
3:     $P \leftarrow all\ possible\ P-links$
4:     $S \leftarrow all\ possible\ S-links$
5:     for each $(i \rightarrow j) \in P$ do
6:         for each $(m \rightarrow n) \in S$ do
7:             $[ST] \leftarrow [1, N]$          ▷ Allowed indices for S-Link
8:             $[PT] \leftarrow \emptyset$          ▷ Allowed indices for P-Link
9:             $t_e \leftarrow 0$          ▷ Temporary variable
10:            $D_{i\rightarrow j,m\rightarrow n} \leftarrow [T_{i\rightarrow j}]^p_{1\times N} - [T_{m\rightarrow n}]^s_{1\times N}$
11:            $\hat{I}, \hat{D}_{i\rightarrow j,m\rightarrow n} \leftarrow sort(D_{i\rightarrow j,m\rightarrow n},\ descend)$
12:                ▷ $\hat{I}$ = indices corresponding to sorted entries
13:            while $t_e < \tau_e$ do
14:                $t_e \leftarrow t_e + T_{i\rightarrow j}(\hat{I}(1))$
15:                $[ST] \leftarrow [ST] - \{\hat{I}(1)\}$          ▷ remove from set
16:                $[PT] \leftarrow [PT] + \{\hat{I}(1)\}$          ▷ add to set
17:                $\hat{I} \leftarrow \hat{I} - \{\hat{I}(1)\}$          ▷ remove from index set
18:            end while
19:        end for
20:    end for
21: end procedure
```

Optimizing overall network throughput: Once minimum throughput requirement of a P-Link is met, the remaining subcarriers are assigned to both P-Link and S-Link such that the total number of modulated bits is maximized i.e. $\max(G_{m\rightarrow n})$. This requires a slight modification to Algorithm 1 (between steps 13-18), such that it will keep assigning subcarriers to P-Link in descending order of the entries in $D_{i\rightarrow j,m\rightarrow n}$, as long as the total number of bits modulated on both links increases.

Optimizing for overall spectrum fairness: Once minimum throughput requirement of a P-Link is met, the remaining subcarriers are assigned to both P-Link and S-Link such that the ratio of number bits modulated along both links approaches 1.0, *i.e.* $\frac{\sum_{[ST]}[T_{m\rightarrow n}]}{\sum_{[PT]}[T_{i\rightarrow j}]} \approx 1.0$. This also requires slight modifications to Algorithm 1 (between steps 13-18).

Complexity: The aforementioned SS approaches will require approximately $4 \times (E-1)(E-2)(E-2)(E-3)(N * log(N) + N)$ computations at each PLC node. The overall complexity of the algorithm can be written as $\mathcal{O}(E^4 * N * log(N))$. Note that the goal of this work is not to support spectrum sharing in networks containing hundreds or thousands of nodes, but to enable such sharing in PLC networks that usually consist of less than a hundred nodes.

2) Ranking of S-Links: While computing allowed tonemaps for each S-Link $m \rightarrow n$, Algorithm 1 also assigns a rank r_{mn} to that S-Link proportional to its SS gain ($r_{mn} = k \cdot G_{m\rightarrow n}$, where $k = 1$ in our paper). We will show how we use these ranks while in the design of our proposed SS protocol for HPAV devices, later in Section VI.

VI. Enabling Spectrum Sharing for HPAV PLCs

In this section, we show how our proposed SS strategy can be enable enabled in the MAC layer of current HPAV PLC devices while incurring minimum firmware level changes.

Channel Probing and Control Overheads: As mentioned before, our SS algorithm works in a distributed manner in the sense that it runs locally at each node, such that the network level SS decisions are eventually known to each node in the PLC network. Therefore, PLC nodes will not have to distribute their SS decisions to other nodes in the network. However, our SS algorithm requires tonemap information about all possible links in the network, which will incur communication overhead. The communication overhead will be on the order of $\mathcal{O}(E^2)$, as each node in a PLC network will broadcast tonemap information for its $(E-1)$ possible links with other nodes in the network. However, due to the pseudo-stationary nature of PLC links (IV-B), this probing overhead will be minimal and will not interfere with regular data transmissions.

A channel probing interval t_{probe} for SS can be set by the CCo of a PLC network. The CCo can then periodically command all nodes the network to log tonemaps of all possible links and formulate their SS decisions. CCo can use control-related messaging schemes already built into HPAV MAC (e.g. Management Messages (MMEs)) for this purpose [1], [2], and t_{probe} can be chosen such that the exchange of control messages incurs minimal overhead and interference to data transmissions. The probing frequency (i.e. $1/\tau_{probe}$) must be kept within a certain threshold in case CCo observes some very dynamic PLC links in its network, because otherwise spectrum sharing may lead to loss of overall network throughput due to high channel probing overhead. CCo can also completely stop spectrum sharing throughout its network and fall back to default HPAV MAC if the channel conditions of PLC links in its network are not conducive to SS. *Note that SS will not be performed during the exchange of control messages.*

Periodic Re-evaluation of Full Spectrum: All nodes in the network will periodically disable SS and transmit across full spectrum following default HPAV MAC. No S-Link will operate in this case. The frequency of this periodic behavior can be chosen by CCo of the network, based on temporal dynamics IV-B of PLC links its network. Such periodic use of the whole spectrum will allow each node to automatically update its full spectrum tonemaps towards other nodes in the network, during regular data transmissions. The network CCo will then re-evaluate the SS decisions in its network by accessing these tonemaps as described before.

Medium Access During SS: Current HPAV MAC is centrally controlled through Beacon signals from CCo. The Beacon signals broadcast by CCo to establish Beacon Periods (BPs) with TDMA and CSMA slots are robust and reliable (Beacons and several other control-related messages operate over ROBust mOdulation (ROBO) modes [10]). Next, we explain how the medium access will work during TDMA and CSMA slots in an SS enabled HPAV MAC.

TDMA: Whenever a P-Link is scheduled to send traffic in a TDMA slot, the highest ranked S-Link (according to the SS algorithm) corresponding to that P-Link will be scheduled to operate in the same slot. In case some of the S-Links corresponding to that P-Link do not have any traffic to send, the highest ranked S-Link will only be chosen from among the S-Links which are waiting in line to send traffic. Therefore, the allowed tonemaps for both P-Link (i.e. $[PT]$) and the selected S-Link (i.e. $[ST]$) will be chosen accordingly.

CSMA/CA: In case of TDMA, the selection of allowed tonemaps $[PT]$ and $[ST]$ is straight forward, since the P-Link and S-Link connections can be specifically scheduled by the CCo to operate in the same slot. However, two major issues arise in case of CSMA: (a) *How will a P-Link know which of the possible S-Links have traffic to send, so it can select its $[PT]$ accordingly?*, and (b) *Assuming issue (a) is resolved, how will the S-Link know that a P-Link is established so that it can select its $[ST]$ according to $[PT]$?* In following steps, we discuss how medium access and the consequent link interactions will differ from the regular HPAV CSMA/CA.

(i) Before broadcasting Beacon signals, the CCo identifies all links with pending traffic, and then shares that information with each node in its network through HPAV control-related messaging. *This resolves issue (a).*

(ii) Once a P-Link gets medium access, the remaining nodes go into their backoff stages, following the regular CSMA/CA procedure. Afterwards, the source node of the P-Link enables the *Multicast Flag* (MCF) in the *Start-of-Frame Control* (SOF) field of its *MAC Protocol Data Unit* (MPDU) [10] while establishing its connection with the destination node, so that all remaining nodes in the network can extract the source and destination IDs of the P-Link from this SOF delimiter field. *This resolves issue (b)* later in step(iv).

(iii) Both source and destination nodes of the P-Link select $[PT]$ corresponding to the highest ranked S-Link among the S-Links with pending traffic (given the information received in step (i)), and use the unshared subcarriers for transmission/reception, while disabling the shared ones.

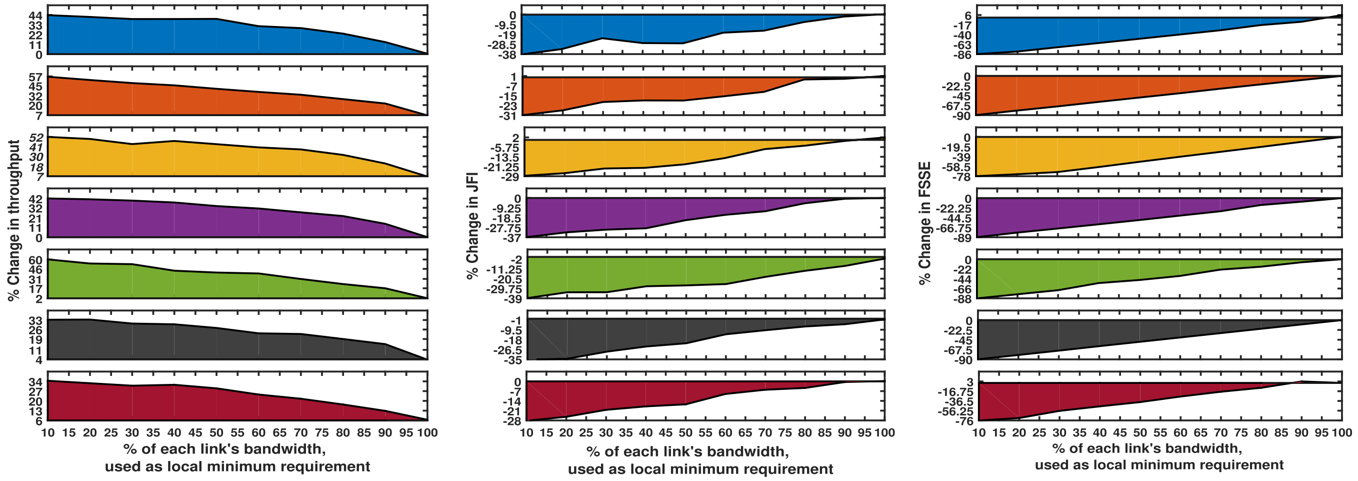
(iv) After knowing the P-Link information from SOF delimiter in P-Link's broadcast MPDU frame, the source and destination nodes of the highest ranked S-Link with pending traffic enable $[ST]$ and disable $[PT]$. The S-Link then operates in parallel with the P-link.

(v) Nodes belonging to any active P-Link come out of their SS state (i.e. re-enable all subcarriers and enter again into contention for whole spectrum) when the transmission between them is finished. As soon as P-Link's transmission ends, the S-Link finishes its transmission as well, and comes out of spectrum sharing state (as the transmission frame ends).

Disabling Modulation of Subcarriers: In current HPAV PHY, each subcarrier is independently modulated based on channel characteristics between transmitter and receiver (i.e. *bit-loading*). HPAV PHY allows dynamic *notching* of specific subcarriers by turning them off, which can be achieved by making soft changes to device's tone mask (enabled subcarriers) [1], [10]. However, currently, this functionality needs proprietary access to firmware supplied by vendors. Therefore, as our evaluations are based on trace-driven simulations, the results in this paper can be taken as an upper bound to the real-world performance as our simulations cannot fully capture the dynamics of a real-time hardware based implementation.

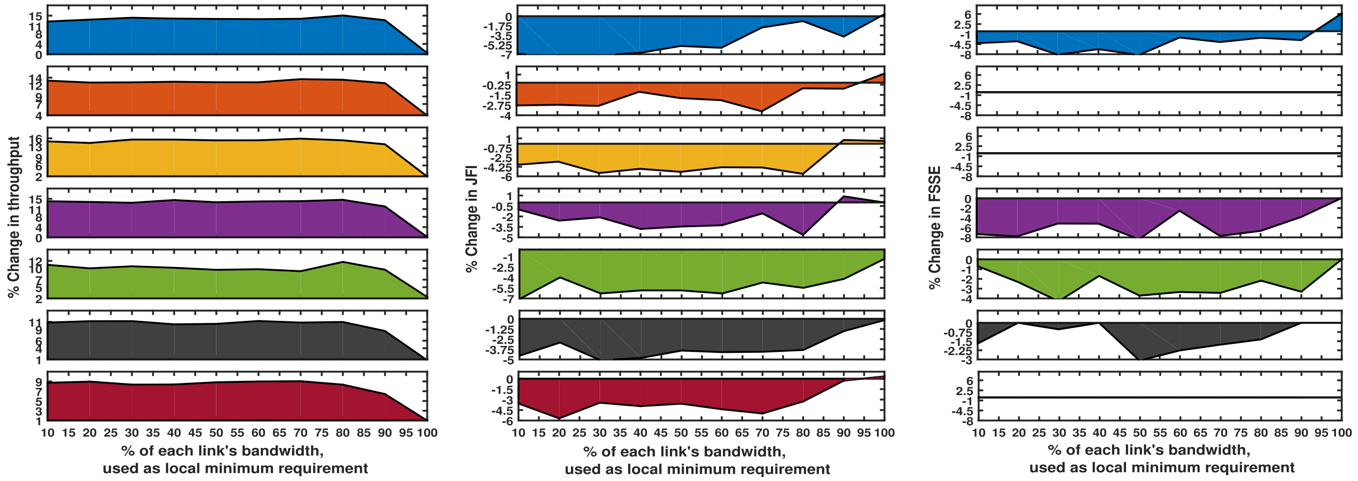
VII. Implementation and Evaluation

We evaluate our proposed SS strategy through trace-driven simulations using traces we obtained from multiple PLC network deployments in our enterprise. We implement and evaluate our proposed SS strategies on top of HPAV CSMA protocol only. We perform trace driven simulations using tonemap traces collected from seven different 4-node PLC deployments (Figure 6 represents Deployment#1). Our simulations do not take into account frame aggregation procedures, bit loading of ethernet frames inside PLC frames, management messages and channel errors, since these parameters are proprietary vendor-specific implementation information. In our simulations, we choose collision duration $\tau_c = 2920.64\mu s$, duration of successful transmission $\tau_s = 2542.64\mu s$ and frame



(a) Network throughput variation. (b) Jain's Fairness Index variation. (c) FSSE variation.

Fig. 7. Testing scenario with per-link (local) minimum $\mathcal{T}_e$, optimizing for net throughput (#1-#7, top-bottom).



(a) Network throughput variation. (b) Jain's Fairness Index variation. (c) FSSE variation.

Fig. 8. Testing scenario with per-link (local) minimum $\mathcal{T}_e$, optimizing overall fairness (#1-#7, top-bottom).

length $F_l = 2050$ [11], [17]. The contention window (CW) and deferral counter (DC) values used for each HPAV CSMA/CA back off stage are $[8,\ 16,\ 32,\ 64]$ and $[0,\ 1,\ 3,\ 15]$, respectively. We assume that there are no hidden terminals, and transmission failures are only due to collisions.

A. Evaluation Metrics

We use following metrics to evaluate the performance of our proposed SS approaches:

Throughput: We calculate the normalized throughput Thr for each link $m \rightarrow n$ in our simulation as follows:

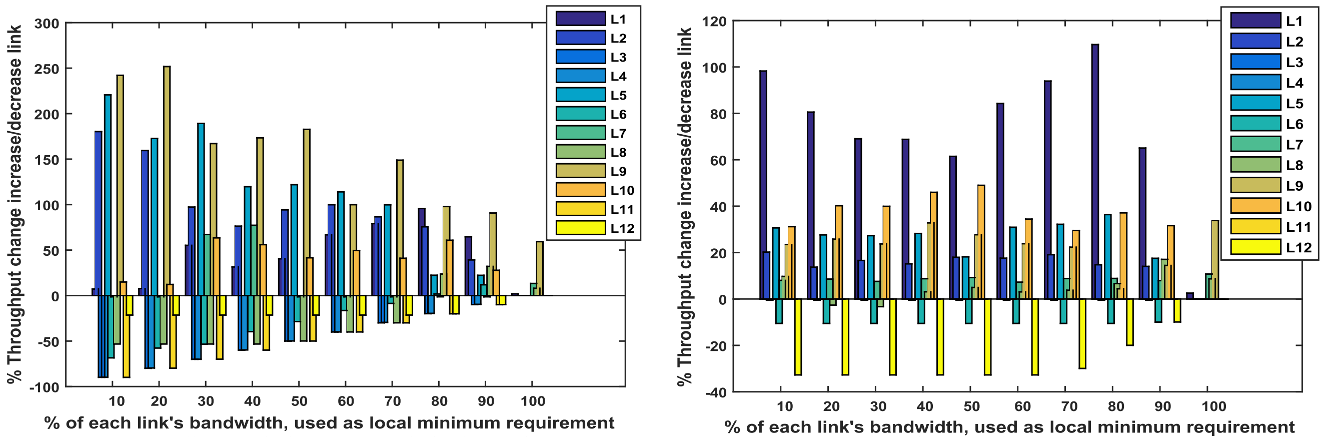
$$Thr = 100 \cdot \frac{[\sum_{i=1}^{[\#SuccessTransmissions]} S_{F_i}] \cdot [Frame\ length]}{Total\ simulation\ time}$$

S_{F_i} represents the fraction of spectrum utilized at i-th transmission. $S_{F_i} = \sum_{j=1}^{N}[T_{m\rightarrow n}]/9170$, such that $\max(S_{F_i}) = 1$ and $\min(S_{F_i}) = 0$.

Fairness: We evaluate the fairness of different SS strategies by calculating *Jain's fairness index* (JFI) [18] and *Fairly Shared Spectrum Efficiency* (FSSE) [19]. An allocation strategy is maximally fair if all nodes in a PLC-Net allocate the same throughput, in which case JFI $= 1$. On the other hand, FSSE of a PLC-Net gives the *spectrum efficiency* (SE) of the PLC node with minimum throughput in the network. In case of maximum spectrum fairness, FSSE is equal to the SE of the whole network. For a PLC network, we define its SE to be its average throughput, and its FSSE to be its minimum throughput.

Next, we show how our SS strategies can achieve higher overall network throughput (when optimized for maximum throughput), and maintain higher fairness (when optimized for fairness), while meeting per-link minimum bandwidth requirements. We test following two scenarios:

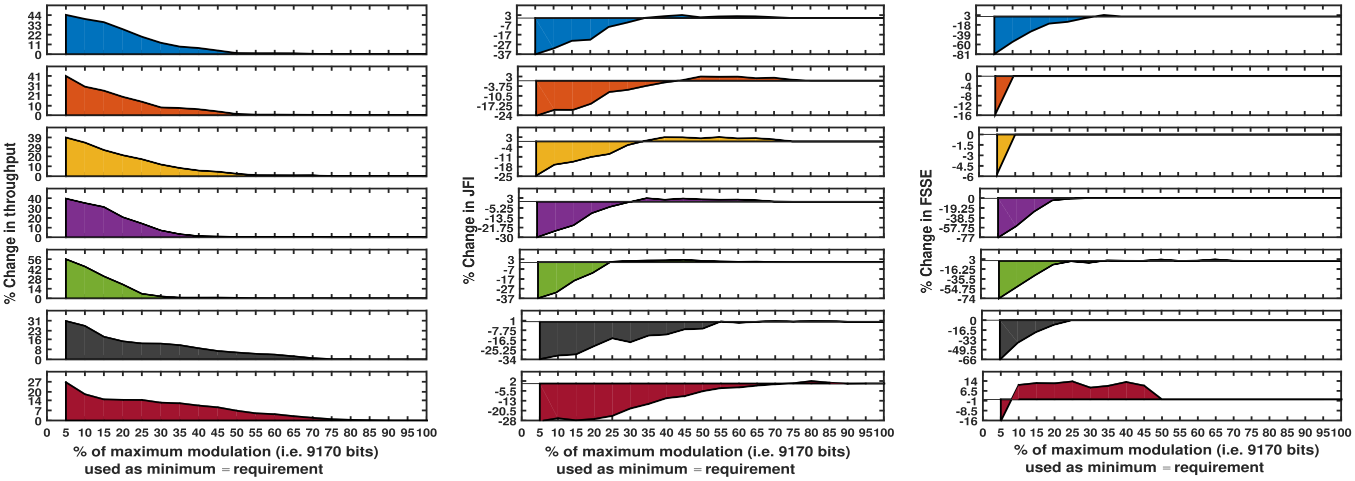
Per-Link (Local) Minimum $\mathcal{T}_e$: In this scenario, each P-Link uses a percentage of it's available bandwidth as it's minimum $\mathcal{T}_e$. Figures 7(a)-7(c) show how net throughput, JFI and FSSE of the seven deployments change as $\mathcal{T}_e$ increases, when SS is optimized for maximum net throughput. The X-axis starts from 10%, i.e. when $\mathcal{T}_e$ is 10% of the available bandwidth. We can observe net throughput gains of up to 60%, and per-link throughput gains as high as 250% (Fig. 9(a)). However, it comes at the expense of large decrements in overall fairness. Figures 8(a)-8(c) show results for scenario



(a) Overall network throughput optimization policy.

(b) Fairness optimization policy.

Fig. 9. Per-link throughput changes for Deployment#1 (testing scenario with per-link (local) minimum $\mathcal{T}_e$ requirement). Note that even if local requirement is 100%, spectrum corresponding to subcarriers that are not modulated at all can still be shared.



(a) Network throughput variation.

(b) Jain's Fairness Index variation.

(c) FSSE variation.

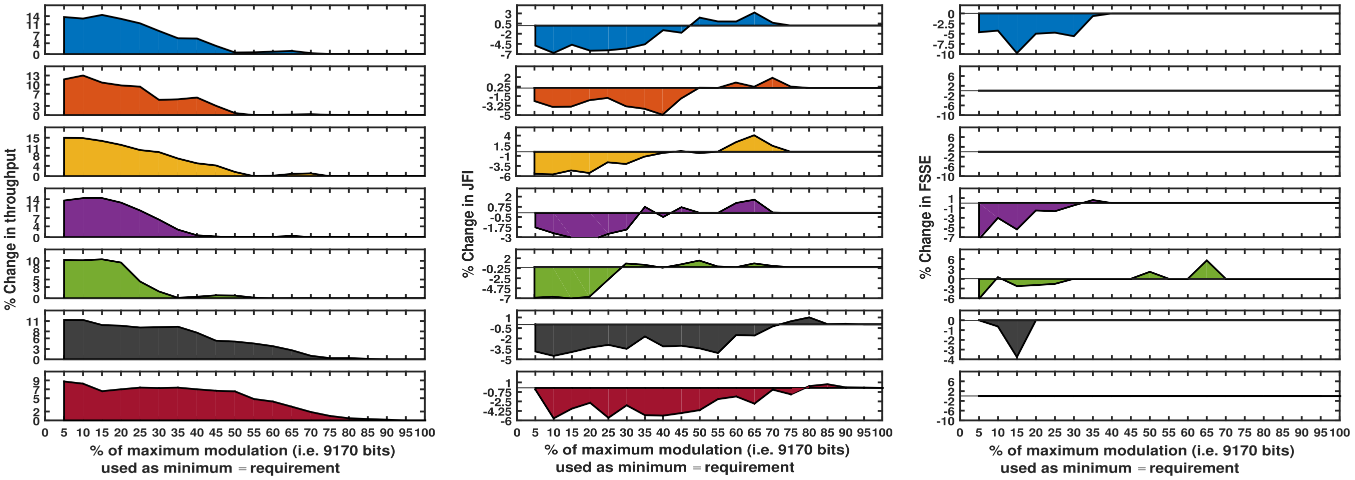
Fig. 10. Testing scenario with network-wide minimum $\mathcal{T}_e$ requirement, optimizing for net throughput (#1-#7, top-bottom).

when SS is optimized to maintain fairness in the network. In this case, we observe net throughput gains of up to 14% and per-link throughput gains as high as 110% (Fig. 9(b)), while incurring much lower decrease in overall fairness, leading to 30% and 87% better JFI and FSSE values (JFI and FSSE of some deployments improve up to 1% and 6%, respectively, for some values of $\mathcal{T}_e$'s)).

Network-Wide Minimum $\mathcal{T}_e$: In this scenario, a percentage of maximum number of bits which can be modulated over any link (i.e. $10 \cdot N = 9170\ bits$) is used as minimum $\mathcal{T}_e$ requirement for each P-Link. Such a scenario can arise when a PLC network is required to meet bandwidth requirements of a certain type of application. Figures 10(a)-10(c) show how net throughput, JFI and FSSE of the seven deployments change as $\mathcal{T}_e$ increases, when SS is optimized for maximum net throughput. We can observe net throughput gains of up to 56%, and per-link throughput gains as high as 180% (Fig. 12(a)), which in most cases comes at the expense of large decrement in fairness performance (except for Deployment#7 (Fig. 10(c) whose FSSE improves up to 14% for some $\mathcal{T}_e$'s)). Figures 11(a)-11(c) show results for scenario when SS is optimized to maintain fairness in the network. In this case, we observe net throughput gains of up to 15% and per-link throughput gains as high as 100% (Fig. 12(b)), while incurring much lower decrease in fairness performance, leading to 25% and 60% better JFI and FSSE values (JFI and FSSE of some deployments improve up to 4% for some values of $\mathcal{T}_e$'s)).

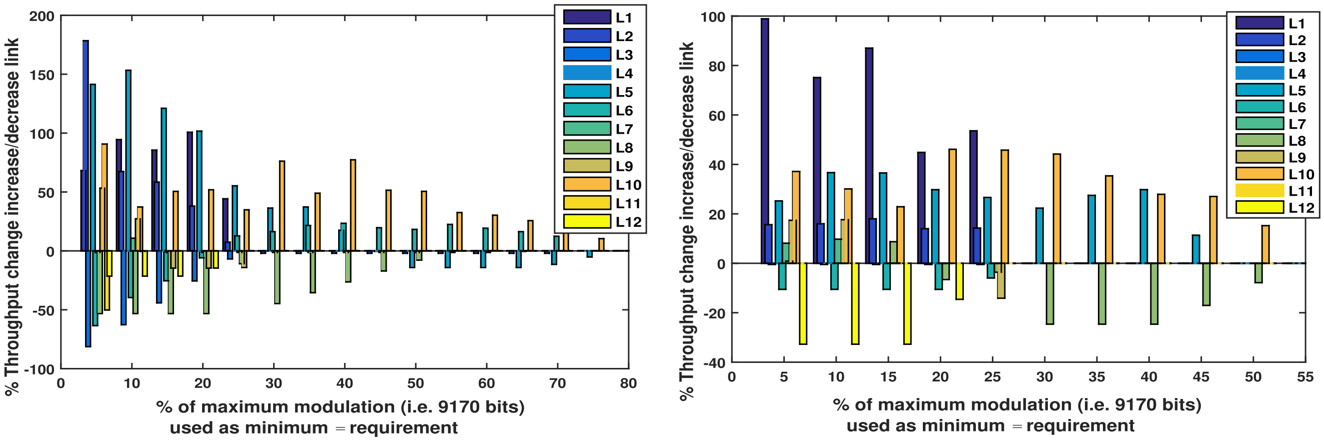
Effect of the Changes in PLC Channels: In real world scenarios, PLC channels can change due to interference caused by electrical appliances or power distribution components, as discussed in § IV. Without regular tonemap updates, PLC links risk high packet loss, specially when the channel between two nodes changes significantly. To counter this, all HPAV devices follow a tonemaps updating protocol (that runs over one of the ROBust mOdulation (ROBO) modes [10]) to update their tonemaps when communicating with other nodes in a network.

Similarly, when SS is enabled, nodes of any S- or P-Link update their tonemaps by simply following the regular HPAV protocol. However, to fully utilize SS, all PLC nodes in the network must share their latest tonemaps with other nodes in the network and then recompute the local SS matrices. However, this incurs both computational and communication related overheads, as discussed in § V and § VI. Figs. 13(a) - 13(d)



(a) Network throughput variation. (b) Jain's Fairness Index variation. (c) FSSE variation.

Fig. 11. Testing scenario with network-wide minimum $\mathcal{T}_e$ requirement, optimizing for overall fairness (#1-#7, top-bottom).

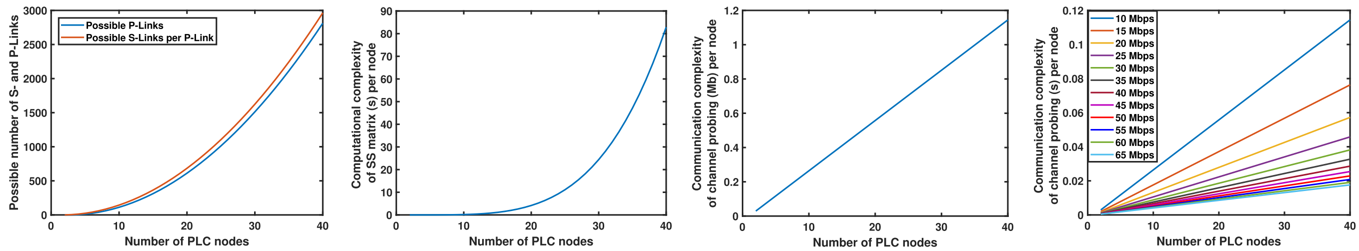


(a) Overall network throughput optimization policy. (b) Fairness optimization policy.

Fig. 12. Per-link throughput changes for Deployment#1 (testing scenario with network-wide minimum $\mathcal{T}_e$ requirement).

show how both overheads change as the number of PLC nodes increase, where the computational complexity of SS algorithm specifically increases significantly. Therefore, spectrum sharing will be beneficial only when most PLC channels in a network are pseudo-stationary, as this will significantly reduce the frequency of channel probing and recalculation of the local SS matrices.

To understand how frequent changes in PLC channels in a network can impact the spectrum sharing gains, we run simulations where we manually add noise to real tonemap data obtained from 3 different deployments. We introduce and evaluate the impact of two parameters in our simulations: (1) *probability* (range 0.05 through 1) that the channel will change during a specific communication instance between two nodes, and (2) *standard deviation* (STD) (range 1 through 3) of the variations introduced in subcarriers (this values stays same for all subcarriers during a specific simulation) whenever the channel changes. In these simulations, we implemented the "net throughput optimization" SS strategy mentioned earlier in this section, where we used a network-wide minimum throughput requirement of $T_e = 25\%$ of the maximum possible modulation (*i.e.* 9170/4 = 2292.5 bits). In our simulations, we measure the normalized throughput and the percentage change in throughput corresponding to 4 different cases: (1) channels do not change at all, (2) channels change, tonemaps updated between communicating nodes, tonemaps shared between all nodes and subcarriers optimally reassigned based on SS algorithm, (3) channels change, tonemaps updated between communicating nodes, but the subcarriers are not reassigned (*i.e.* assignment stays intact), and (4) channels change but tonemaps are not updated between the communicating nodes. Figs. 14 - 20 show the results corresponding to all of the 3 real-world deployments. We observe that throughput deteriorates significantly in case 4, as not updating tonemaps leads to significant number of unsuccessful transmissions. We discuss case 4 for the sake of completeness, as it never occurs either in regular or SS enabled HPAV protocol. Interestingly, we observe that case 3 exhibits significant SS gains even though the latest tonemaps were not shared between all nodes and the subcarriers were not optimally reassigned. However, the gains in case 3 are less than the gains in case 2, and they decrease as the probability of channel change and/or STD of variations increases, as shown by results in Figs 20(a) - 20(c).



(a) Possible links as nodes increase. (b) Complexity of SS algorithm. (c) Communication complexity (Mbs). (d) Communication complexity (secs).

Fig. 13. Computational and communication complexity of our spectrum sharing approach as number of PLC nodes increase.

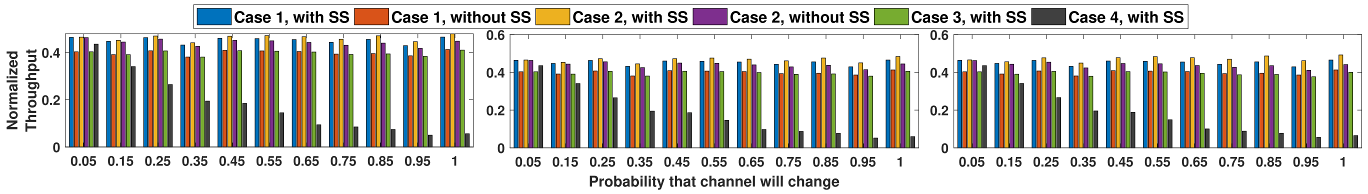


Fig. 14. Normalized throughputs observed for different cases in deployment #1 (STD of noise 1, 2 and 3 from left-right).

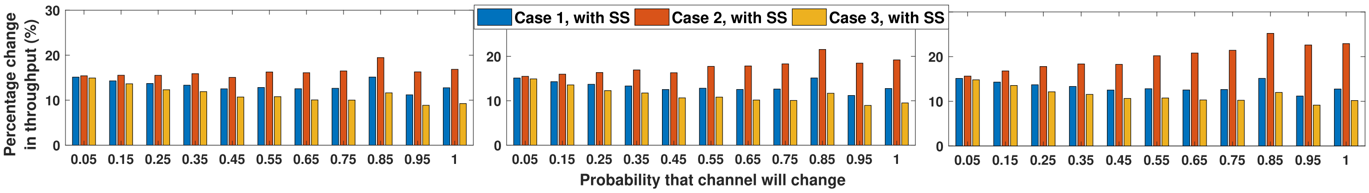


Fig. 15. Percentage change in throughput for different cases in deployment #1 (STD of noise 1, 2 and 3 from left-right).

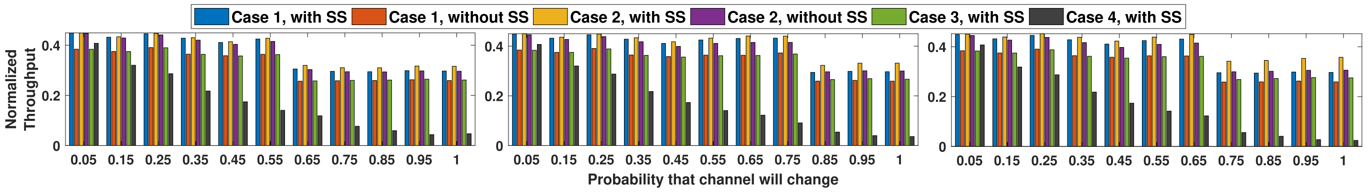


Fig. 16. Normalized throughputs observed for different cases in deployment #2 (STD of noise 1, 2 and 3 from left-right).

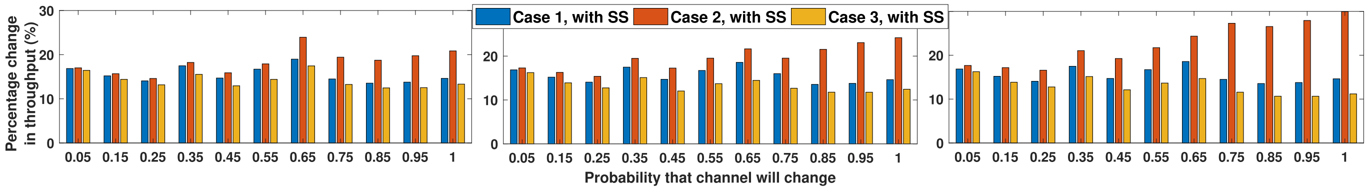


Fig. 17. Percentage change in throughput for different cases in deployment #2 (STD of noise 1, 2 and 3 from left-right).

VIII. Discussions and Future Work

Multi-Hop Routing: HomePlug PLC devices currently do not support multi-hop communication [1], [2]. However, direct link communication in PLC networks can often be impossible or show very low throughput, either due to highly location dependent multipath attenuations and/or interference from appliances. Results from our initial study have shown that mesh routing can significantly boost PLC-Net performance in scenarios where direct PLC links perform very poorly and multi-hop communication is required. For our evaluation, we use the optimized link state routing protocol (OLSR) [20], which is a table-driven proactive link-state routing protocol and has been widely used in 802.11 wireless mesh networks. For our testbed experiments, we first port the open-source OLSR and ETT implementations [21], [22] in our OpenWrt boards. Then, we deploy 9 PLC nodes in various topologies in our floorplan (Fig. 3) and then evaluate routing performance of the PLC-Net. Our results show that routing can significantly improve PLC-Net performance in scenarios where certain PLC links perform very poorly. We identified such a scenario during the communication between PLC nodes N9 and N6, which were located at different breakers but in the same distribution line (cf. Fig. 3). Figure 21, shows the UDP throughput performance between PLC node N9 and

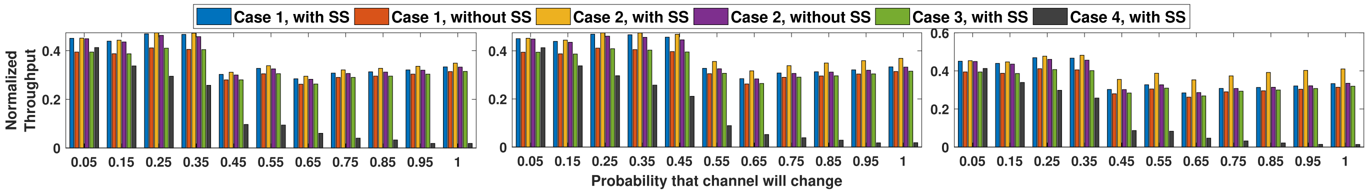


Fig. 18. Normalized throughputs observed for different cases in deployment #3 (STD of noise 1, 2 and 3 from left-right).

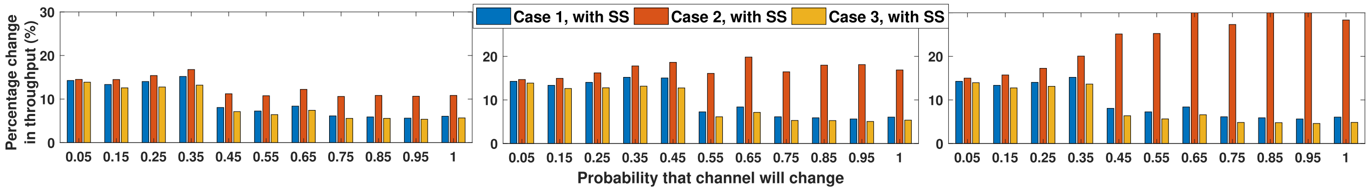
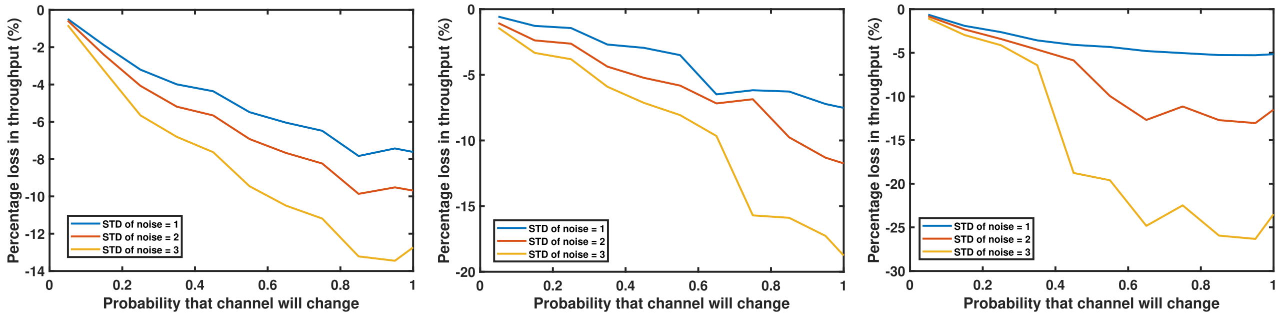


Fig. 19. Percentage change in throughput for different cases in deployment #3 (STD of noise 1, 2 and 3 from left-right).



(a) Deployment #1. (b) Deployment #2. (c) Deployment #3.

Fig. 20. Percentage loss in throughput of case 3 compared to case 2 as the probability of change in PLC channels increases.

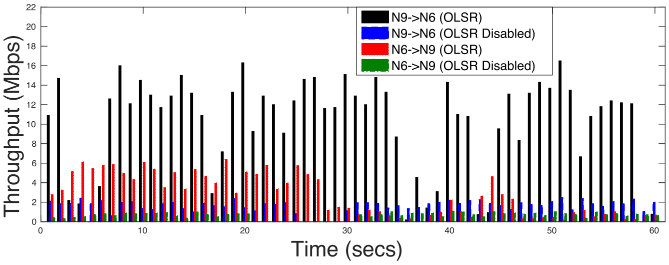


Fig. 21. Throughput with OLSR on/off for 60 secs.

N6 for one minute window, while OLSR is enabled and disabled. When OLSR is turned on, UDP throughput between N9-N6 and N6-N9 is 5.6 and 4.5 times higher, respectively, as compared to the case when OLSR is off. We observe that such communication is affected by electrical devices (lamps, phone chargers, monitors) between N9 and N6, which interfere with the PLC network. When OLSR is enabled, N9 and N6 communicate through node N7 or N8, avoiding such interferences. When OLSR is on, TCP throughput is up to 3.6 times higher compared to the case when OLSR is off. We believe that combining multi-hop routing with fine grained spectrum sharing can potentially improve PLC network performance even further, especially in scenarios where direct PLC links perform poorly.

TABLE I

UDP and Single-Flow TCP Throughput and Jitter With OLSR On/Off (Jitter Is Reported by iperf Only for UDP)

Traffic	Flow	Thr(Mbps)	Jitter(ms)
UDP	N9→N6 (olsr on)	9.5	5.9
	N9→N6 (olsr off)	1.7	17.8
	N6→N9 (olsr on)	2.7	11.6
	N6→N9 (olsr off)	0.6	18.7
TCP	N9→N6 (olsr on)	4.2	-
	N9→N6 (olsr off)	1.4	-
	N6→N9 (olsr on)	1.8	-
	N6→N9 (olsr off)	0.5	-

IX. Conclusion

In this work, we make following contributions. First, we conduct an extensive measurement study of PLCs in a real enterprise environment using COTS HPAV PLC devices, based on which we conclude that spectrum sharing (not supported by existing PLC standards) can significantly benefit enterprise level PLC mesh networks. Second, we propose, implement and evaluate a spectrum sharing scheme, and show that fine-grained distributed spectrum sharing can significantly boost the aggregated and per-link throughput performance by up to 60% and 250% respectively, by allowing multiple PLC links to communicate concurrently, while requiring only a few modifications to the existing HPAV devices and protocols.

Acknowledgment

K. Ali did a part of this work during his internship at Hewlett Packard Labs. I. Pefkianakis was also employed by Hewlett Packard Labs while he contributed to this work.

REFERENCES

[1] (2007). *Homeplug AV WhitePaper*. [Online]. Available: http://www.homeplug.org/tech-resources/resources/

[2] (2011). *Homeplug AV2 WhitePaper*. [Online]. Available: http://www.homeplug.org/tech-resources/resources/

[3] P. Achaichia, M. Le Bot, and P. Siohan, "Point-to-multipoint communication in power line networks: A novel FDM access method," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Jun. 2012, pp. 3424–3428.

[4] T. Hayasaki, D. Umehara, S. Denno, and M. Morikura, "A bit-loaded OFDMA for in-home power line communications," in *Proc. IEEE Int. Symp. Power Line Commun. Appl.*, Mar. 2009, pp. 171–176.

[5] C. Vlachou, S. Henri, and P. Thiran, "Electri-Fi your data: Measuring and combining power-line communications with WiFi," in *Proc. ACM Internet Meas. Conf.*, 2015, pp. 325–338.

[6] R. Murty, J. Padhye, R. Chandra, A. R. Chowdhury, and M. Welsh, "Characterizing the end-to-end performance of indoor powerline networks," Microsoft Res., Harvard Univ., Cambridge, MA, USA, Tech. Rep., 2008. [Online]. Available: http://www.eecs.harvard.edu/~rohan/Research.html

[7] A. O. F. Atya, K. Sundaresan, S. V. Krishnamurthy, M. A. Khojastepour, and S. Rangarajan, "BOLT: Realizing high throughput power line communication networks," in *Proc. ACM CoNEXT*, 2015, pp. 1–13.

[8] L. Di Bert, P. Caldera, D. Schwingshackl, and A. M. Tonello, "On noise modeling for power line communications," in *Proc. IEEE Int. Symp. Power Line Commun. Appl.*, Apr. 2011, pp. 283–288.

[9] H. B. Çelebi, "Noise and multipath characteristics of power line communication channels," Ph.D. dissertation, Dept. Elect. Eng., Univ. South Florida, Tampa, FL, USA, 2010.

[10] H. A. Latchman, S. Katar, L. Yonge, and S. Gavette, *Homeplug AV IEEE 1901: A Handbook for PLC Designers Users*. Hoboken, NJ, USA: Wiley, 2013.

[11] C. Vlachou, A. Banchs, J. Herzen, and P. Thiran, "On the MAC for power-line communications: Modeling assumptions and performance tradeoffs," in *Proc. IEEE Int. Conf. Netw. Protocols (ICNP)*, Oct. 2014, pp. 456–467.

[12] *IEEE Standard for Broadband over Power Line Networks: Medium Access Control and Physical Layer Specifications*, IEEE Standard 1901-2010, 2010.

[13] V. Yenamandra and S. Kannan, "Vidyut: Exploiting power line infrastructure for enterprise wireless networks," in *Proc. ACM SIGCOMM*, 2014, pp. 595–606.

[14] M. Zimmermann and D. Klaus, "An analysis of the broadband noise scenario in powerline networks," in *Proc. Int. Symp. Powerline Commun. Appl.*, 2000, pp. 131–138.

[15] M. Zimmermann and K. Dostert, "Analysis and modeling of impulsive noise in broad-band powerline communications," *IEEE Trans. Electromagn. Compat.*, vol. 44, no. 1, pp. 249–258, Aug. 2002.

[16] M. Zimmermann and K. Dostert, "A multipath model for the powerline channel," *IEEE Trans. Commun.*, vol. 50, no. 4, pp. 553–559, Apr. 2002.

[17] C. Vlachou, A. Banchs, J. Herzen, and P. Thiran, "Analyzing and boosting the performance of power-line communication networks," in *Proc. 10th ACM Int. Conf. Emerg. Netw. Exp. Technol.*, Dec. 2014, pp. 1–12.

[18] R. K. Jain, D.-M. W. Chiu, and W. R. Hawe, "A quantitative measure of fairness and discrimination for resource allocation in shared computer systems," Eastern Res. Lab., Lexington, MA, USA, Tech. Rep. DEC-TR-301, 1984.

[19] M. Eriksson, "Dynamic single frequency networks," *IEEE J. Sel. Areas Commun.*, vol. 19, no. 10, pp. 1905–1914, Oct. 2001.

[20] T. Clausen, *Optimized Link State Routing Protocol (OLSR)*, document RFC 3626, 2003.

[21] *OLSR Routing Protocol*. Accessed: Oct. 15, 2015. [Online]. Available: http://www.olsr.org

[22] *OLSR With Link Cost Extensions*. Accessed: Oct. 15, 2015. [Online]. Available: http://sourceforge.net/projects/olsr-lc/

Kamran Ali received the B.S. degree in electrical engineering and computer sciences from the School of Science and Engineering (SSE), Lahore University of Management Sciences (LUMS), Pakistan, in 2013, and the Ph.D. degree in computer science and engineering from Michigan State University, USA, in 2019. He is currently with General Motors R&D. He has worked with Hewlett Packard Labs, NEC Laboratories, Nokia Bell Labs and Microsofts Applied Sciences Group. His research interests lie in the areas of communication systems, sensing systems, RF sensing, machine learning, signal processing, pervasive and mobile computing, and the Internet of Things.

Alex X. Liu (Fellow, IEEE) received the Ph.D. degree in computer science from The University of Texas at Austin, in 2006. He is an Honorary Professor with the Qilu University of Technology. Before that, he was a Professor with the Department of Computer Science and Engineering, Michigan State University (MSU). He received the IEEE and IFIP William C. Carter Award in 2004, a National Science Foundation CAREER award in 2009, the MSU Withrow Distinguished Scholar (Junior) Award in 2011, and the MSU Withrow Distinguished Scholar (Senior) Award in 2019. He has served as an Editor of the IEEE/ACM TRANSACTIONS ON NETWORKING and an Area Editor of *Computer Communications*. He is currently an Associate Editor of the IEEE TRANSACTIONS ON DEPENDABLE AND SECURE COMPUTING and IEEE TRANSACTIONS ON MOBILE COMPUTING. He has served as the TPC Co-Chair for ICNP 2014 and IFIP Networking 2019. He received Best Paper Awards from SECON-2018, ICNP-2012, SRDS-2012, and LISA-2010. His research interests focus on networking, security, and privacy. He is an ACM Distinguished Scientist.

Ioannis Pefkianakis received the Ph.D. degree in computer science from the University of California Los Angeles (UCLA), in 2012. He is currently a Wireless Architect at Apple. Before that, he was a Senior Research Scientist at Hewlett Packard Labs, California, USA. Before joining HP Labs, he worked as a Research Scientist at Technicolor Labs, France. His research interests broadly span the areas of wireless networking and mobile computing.

Kyu-Han Kim (Senior Member, IEEE) received the B.S. degree from Korea University, Seoul, the M.S. degree from the Georgia Institute of Technology, Atlanta, and the Ph.D. degree from the University of Michigan, Ann Arbor, all in computer science. He is currently a Principal Researcher and the Director at Hewlett Packard Laboratories, California, USA, and is leading the mobility and networking research group. His research interests include performance, quality-of-experience, manageability, real-time analytics and data privacy in mobile, distributed, embedded, and cloud systems as well as wireless networks. He received the ACM MobiCom Best Student Paper Award (coauthor, 2003) and a government scholarship (2001–2005) from the Ministry of Information and Communication, South Korea. He is a Senior Member of the IEEE Computer Society. He served as a TPC, an associate editor and external reviewers in many top-conferences and journals for more than ten years (IEEE INFOCOM, IEEE TMC, IEEE ToN, IEEE TWC, IEEE SECON, ACM MobiSys, ACM MobiCom, and USENIX NSDI).