SOLOR: Self-Optimizing WLANs with Legacy-Compatible Opportunistic Relays

Andres Garcia-Saavedra, Balaji Rengarajan, Pablo Serrano *Member, IEEE* Daniel Camps-Mur, Xavier Costa-Pérez

Abstract—Current IEEE 802.11 WLANs suffer from the well-known *rate anomaly* problem, which can drastically reduce network performance. Opportunistic relaying can address this problem, but three major considerations, typically considered separately by prior work, need to be taken into account for an efficient deployment in real-world systems: 1) relaying could imply increased power consumption, and nodes might be heterogeneous, both in power source (e.g., battery-powered vs. socket-powered) and power consumption profile; 2) similarly, nodes in the network are expected to have heterogeneous throughput needs and preferences in terms of the throughput vs. energy consumption trade-off; and 3) any proposed solution should be backwards-compatible, given the large number of legacy 802.11 devices already present in existing networks.

In this paper, we propose a novel framework, Self-Optimizing, Legacy-Compatible Opportunistic Relaying (SOLOR), which jointly takes into account the above considerations and greatly improves network performance even in systems comprised mostly of vanilla nodes and legacy access points. SOLOR jointly optimizes the *topology* of the network, i.e., which are the nodes associated to each relay-capable node; and the *relay schedules*, i.e., how the relays split time between the downstream nodes they relay for and the upstream flow to access points. Our results, obtained for a large variety of scenarios and different node preferences, illustrate the significant gains achieved by our approach. Specifically, SOLOR greatly improves network throughput performance (more than doubling it) and power consumption (up to 75% reduction) even in systems comprised mostly of vanilla nodes and legacy access points. Its feasibility is demonstrated through test-bed experimentation in a realistic deployment.

Index Terms—Wireless LAN, 802.11, rate anomaly, relays

I. INTRODUCTION

IN IEEE 802.11 WLANs, stations associated to an Access Point (AP) can experience different signal-to-noise ratios (SNRs), depending on several factors, e.g., their distance to the AP, the presence of physical obstacles, or the particular characteristics of their RF equipment. The various physical
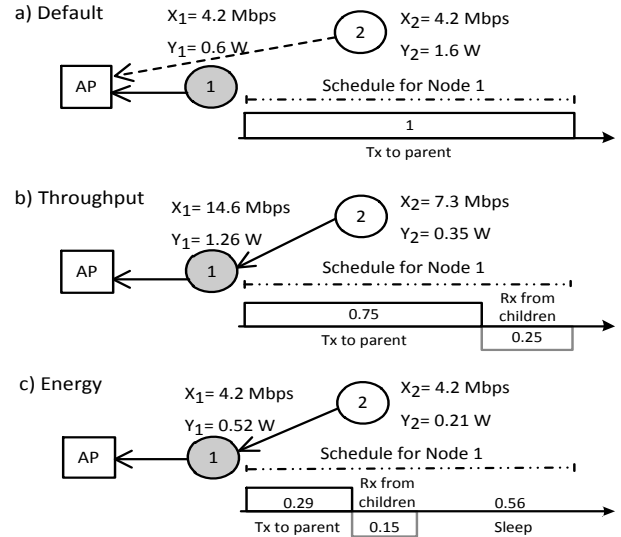
Fig. 1: Different configurations for a deployment consisting of one AP and two stations (one with relay capabilities, marked in grey).

layers available (see [1] for a survey of 802.11 standards) offer stations a variety of modulation and coding schemes (MCS) to choose from, in order to optimally adapt the MCS to the channel conditions. However, it is well-known that this heterogeneity in the use of MCS may induce the *rate anomaly* problem [2], which degrades the performance of the WLAN.

To illustrate the above, let us consider the case of uplink traffic in the simplified scenario of Fig. 1a, which we refer to as the "Default" case and that consists of two stations (nodes 1 and 2) simultaneously transmitting to an AP. Given their different radio conditions, node 1 uses the 48 Mbps rate, while node 2 uses 6 Mbps. In this case, both stations will receive equal throughput of approximately $X_1 = X_2 = 4.2$ Mbps,[1] which for the case of node 1 is well below its maximum achievable rate. This phenomenon is termed the *rate anomaly* problem, and is a direct consequence of the medium access mechanism, which results in the station transmitting at low rate occupying the channel for the majority of time.

A method that has been proposed to address this rate anomaly problem, and in general to lessen the impact of poor radio conditions, is to use the relaying capabilities of

---

[1]The model used to compute the throughput and power consumption figures is detailed in Section III.

some nodes [3]–[8] (related work is discussed in detail in Section II), which can act as APs for those suffering from poor radio conditions. Indeed, this opportunistic use of the "AP-like" functionality has been defined in the Wi-Fi Direct specification [9], which is readily available in several devices (e.g., recent Android phones), some of them building on the p2p open-source implementation.[2]

For example, in our simplified scenario, if node 1 is relay-capable, it could enable the cases of Fig. 1b and Fig. 1c, which we name as "Throughput" and "Energy", respectively, for reasons that will become evident shortly. In these cases, node 1 acts as an AP for node 2, and is responsible for sending both its own data and that of node 2 to the AP. This creates a different **topology**, i.e., the paths between stations and the AP (we will formally introduce our terminology in the next section). Assuming that nodes are equipped with a single radio, node 1 has to time share between serving node 2 and transmitting to the AP. We refer to this choice of the fractions of time a relay spends in these activities as the **relay schedule**. Given the new topology considered in the figure, the schedule will determine the network performance, and therefore it has to be tuned depending on some optimization criterion.

For the case of Fig. 1b, the network is optimized based exclusively on throughput considerations, and according to the *proportional fairness* criterion, which results in node 1 spending 25% of its time serving node 2, and the rest of the time transmitting to the AP. Clearly, even in this fairly simple scenario, the throughput improvements obtained through the intelligent use of relaying can be significant. However, although all nodes get higher throughput, now the power consumption of the relay (Y_1) is higher than in the Default case, due to the increased time spent in energy-intensive operations, i.e., transmitting and receiving packets. With mobile, battery-powered devices being sensitive to energy consumption, this trade-off between performance and energy consumption has to be carefully managed [10]. An alternate relay schedule, which minimizes energy consumption (by making use of sleep modes) while guaranteeing minimum throughput above the Default scenario, is given in Fig. 1c. Here, node 2 is forced to sleep for 85% of the time, while node 1 sleeps for 56%, thus reducing the overall energy consumption from 2.20 W to 0.73 W (i.e., a 67% reduction).

The relative importance of throughput and power consumption depends on the characteristics of each station, e.g., if it is battery-powered or plugged in to a socket, or has specific throughput requirements. The criterion used, and the topology and schedule chosen should reflect the preferences of the nodes in the network. Another important consideration from the point of view of practicality is backwards-compatibility. Given the large number of legacy 802.11 devices already present in existing networks, mechanisms that require changes in all nodes in order to work are impractical. A practical scheme must be able to work under the distributed coordination function (DCF), which is the most prevalent operating mode in existing 802.11 networks. As we show in the sequel, significant performance gains and power savings can be obtained even when the ratio

of relay-capable nodes to legacy nodes is low.

The key contributions of this paper are:

- A novel, legacy-compatible framework for optimization of performance and power consumption of a WLAN with relay-capable nodes, reflecting heterogeneous power vs. performance preferences of individual nodes.
- A low-complexity algorithm for topology control, that enables the joint optimization of network topology and relay schedule in a fast, scalable manner.
- Numerical evaluation for a large variety of scenarios in terms of node density, proportion of relays, network size, and performance criteria that illustrate the flexibility and benefits of the proposed framework.
- Experiments using a real-world testbed comprised of off-the-shelf devices that demonstrate the practicality of the proposed approach and validate the model and the achieved gains: more than double network throughput performance improvement and power consumption reduction up to 75%.
- A performance comparison of SOLOR vs. the most prevalent solutions based on the use of relays. This comparison shows notable gains with SOLOR, which are due to the increase knowledge of the network, the coordination between relays, and the relaxed requirement of *smart* nodes in the network.

The rest of the paper is organized as follows. Related work is discussed in Section II. In Section III we introduce the key parameters of our model, namely, topology and relay schedule, and present the throughput and power consumption models used throughout the paper. In Section IV we present our optimization framework that can be solved for the optimal relay schedule and heuristics to pick the best topology. The results from the optimization are provided in Section VI for a variety of WLAN deployments, while in Section VII we report our experimental results using a mid-sized testbed composed of commercial, off-the-shelf devices. We compare the performance of SOLOR against previous approaches in Section VIII. Finally, Section IX summarizes our contributions and concludes the paper.

II. RELATED WORK

One of the first proposals to improve performance through the use of relays is RAMA [3], which incurs in a high implementation complexity, is not tested experimentally, and does not optimize energy efficiency. Another proposal that lacks experimental support is [4], which is tailored to multicast traffic.

In contrast to the above schemes, both Soft-Repeater [5] and PRO [6] have been implemented and tested in practice. The former is designed to address the rate anomaly problem, while the later opportunistically retransmit those frames that may have been missed by the intended destination. However, these schemes do not take into account energy consumption, and therefore cannot be used in scenarios where e.g. devices run on batteries or have different energy consumption characteristics. Furthermore, they do not support operation with legacy nodes.

Energy-efficient operation is considered by both Coop-MAC [7] and CRS [8], but they do not support operation

---

[2]http://linuxwireless.org/en/developers/p2p/

with legacy nodes (which challenges their practicality) and do not take into account device heterogeneity in terms of the performance vs. consumption trade-off.

In contrast to all these schemes, our SOLOR framework is able to optimize performance taking into account nodes' preferences and is compatible with the operation of legacy nodes. Moreover, the works presented above ([3]–[8]), either assume a static topology or propose a naïve topology control scheme that could not deal with complex networks like SOLOR does. Indeed, we will show in its performance evaluation that SOLOR provides substantial improvements even in scenarios comprised mostly of legacy nodes. To provide these improvements, SOLOR optimizes the way nodes reach the Access Point, i.e., the topology of the network. Several works have studied topology control in the field of multi-hop adhoc networks, particularly in the context of sensor networks (see [11] for a recent survey). However, these works focus on transmission power control to adapt the transmission ranges of the nodes to reduce their consumption. In contrast, SOLOR adapts the topology to enable the required MCS rates to improve performance, considering both throughput and energy consumption, enabling a per-node specific trade-off of these performance figures.[3]

III. SYSTEM MODEL AND NOTATION

Our scenario consists of a network with one AP, denoted node 0, and N other nodes, together denoted by the set $\mathcal{N} = \{0, 1, 2, \ldots, N\}$. Let $\mathcal{S} \subseteq \mathcal{N}$ be the set of relay-capable nodes, which for notational convenience includes the AP. We assume that all nodes are single-radio, i.e., they cannot simultaneously transmit over two different channels. We focus, for simplicity, on the uplink case (we relax this assumption later) and assume that all nodes are saturated, i.e., their buffers are always backlogged. We denote by R_{ij}, the rate corresponding to the MCS used between nodes i, j, and with R_i the data rate of the MCS at which node i transmits to the AP, i.e., $R_i := R_{i0}$.

We assume that the AP and relays use an orthogonal set of channels to communicate with their respective clients. While in the 2.4 GHz band this assumption restricts the use of SOLOR to small networks, we note that even in those cases, the performance improvements are remarkable (as we will see in the performance evaluation). Furthermore, this assumption results less restrictive in the 5 GHz band, given the larger set of orthogonal channels available.

A. System Abstractions

Network Topology: We assume that each node uses only one path, consisting of one or more wireless links, to reach the AP (i.e., no multi-path). We refer to the **topology** of the network as the set of paths that nodes use to reach the AP. More formally, the network topology is specified by defining for each node n, its *parent* $A_n \in \mathcal{S}$, which is the first-hop node on the path to the access point. For the case of e.g. Fig. 1b, the topology is

---

[3]Although the schemes of [11] could be used to develop new topology control mechanisms for SOLOR, we note that, for the considered scenarios, the performance gain of the heuristic presented in this paper is very close to the one resulting from exhaustive searches.
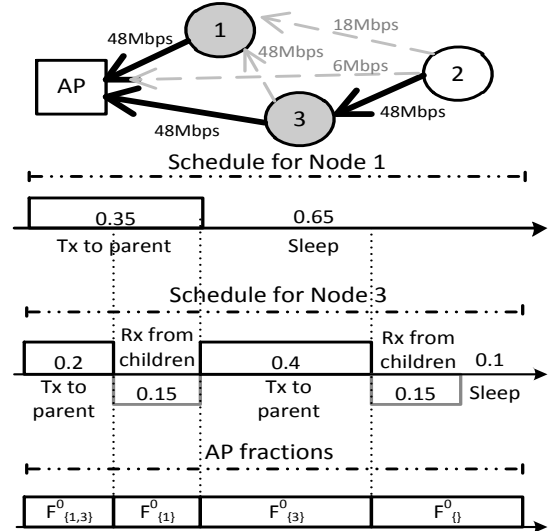

Fig. 2: Scenario with one AP and three stations (two with relay capabilities).

defined as $\{A_1 = 0, A_2 = 1\}$. Given a topology $\{A_n\}$, we can determine for each node $m \in \mathcal{N}$ its set of *children* $\mathcal{C}_m$, i.e., the set of nodes one hop away from m that reaches the AP through it, as $\mathcal{C}_m = \{n : n \in \mathcal{N}, A_n = m\}$. The complete set of nodes that use m to reach the AP is defined as $\mathcal{T}_m = \mathcal{C}_m \bigcup_{n \in \mathcal{C}_m} \mathcal{T}_n$. Note that, for a node $m \notin \mathcal{S}$, $\mathcal{T}_m = \mathcal{C}_m = \emptyset$.

Relay Schedule: A relay-capable node can, in general, be in one of three different states, namely, (i) serving its children, (ii) communicating with its corresponding parent, or (iii) in the sleep state. **Relay schedules** refer to the timing of the state transitions for each relay in the network. Given that there can be other relays among the children of a relay, the set of children contending for access to a relay can vary over time. This is illustrated in Fig. 2 where both node 1 and node 3 are relay-capable. Here, node 3 spends part of the time transmitting to the AP, and part of the time acting as AP for node 2. We denote by $\mathbb{W}^s$ the collection of all possible sets of nodes that could simultaneously transmit to a given relay s. For the case of Fig. 2, we have $\mathbb{W}^3 = \{(2)\}$ and $\mathbb{W}^0 = \{(1, 3), (1), (3)\}$ (note that node 1 is also a relay-enabled node). Fig. 2 also illustrates that the relay schedules determine the fraction of time that a particular set of nodes $\mathcal{V} \in \mathbb{W}^s$ is simultaneously transmitting to relay s. For the case of the AP, we have that e.g. it receives traffic from nodes 1 and 3 for 20% of the time, which we denote as $F^0_{\{1,3\}} = 0.2$, and it does not receive traffic from any node 25% of the time, which is denoted as $F^0_{\{\}} = 0.25$. As we will detail in Section IV, the policy that we follow to compute the relays' schedules ensures a one-to-one mapping between these and the set of fractions $\vec{F} = \{F^s_{\mathcal{V}}; s \in \mathcal{S}, \mathcal{V} \in \mathbb{W}^s\}$.

The **configuration** of the network is jointly determined by the topology $\{A_n\}$ and the relay schedules with the induced set of fractions $\vec{F}$.

B. Throughput model

Let $R^s_{\mathcal{V}}(n)$ be the instantaneous throughput obtained by node $n \in \mathcal{V}$ from relay s when the set of nodes simultaneously

transmitting to s is $\mathcal{V}$. This can be computed by following, for example, our analysis in [12] that extends the seminal work of [13] to address heterogeneous MCS.

We will use the convention that $R_{\mathcal{V}}^s(n) = 0$ if $n \notin \mathcal{V}$. In the sequel, we suppress the relay identity, s. Based on this, the total throughput obtained by a non-relay node n is computed as the average throughput over time as:

$$X_n = \sum_{\mathcal{V} \in \mathbb{W}^{A_n}} R_{\mathcal{V}}(n) F_{\mathcal{V}}^{A_n}. \tag{1}$$

In order to compute the throughput of a relay node $s \in \mathcal{S} \setminus \{0\}$ we need to subtract from the total throughput it obtains, the throughput required to serve the set of nodes that access the AP through it (i.e., $\mathcal{T}_s$),

$$X_s = \sum_{\substack{\mathcal{V} \in \mathbb{W}^{A_s} \\ s \in \mathcal{V}}} F_{\mathcal{V}}^{A_s} R_{\mathcal{V}}(s) - \sum_{t \in \mathcal{T}_s} X_t. \tag{2}$$

C. Power Consumption Model

We follow the conventional model (see e.g. [14] and references therein) that the power consumption of an 802.11 node can be modeled after the fraction of time it spends in transmit, receive, idle, and sleep modes, along with the corresponding per-state power consumption figures, i.e., ρ_{tx}, ρ_{rx}, ρ_{id} and ρ_s, respectively (see [15] for an extensive survey). We denote by $P_{\mathcal{V}}^T(n)$ the power consumed by node n when the set of active nodes transmitting to its relay is $\mathcal{V}$. Here, the dependence on $\mathcal{V}$, the set of contending nodes, reflects the effect of contention, and the frame spacings mandated by the 802.11 standard. The above can be computed by following, for example, our results in [14]. We will assume that whenever a node is not actively transmitting data because its parent is not available, it remains in the sleep state with corresponding power consumption ρ_s.[4] Based on this, the power consumed by a non-relay node $n \notin \mathcal{S}$ is computed as

$$Y_n = \sum_{\mathcal{V} \in \mathbb{W}^{A_n}} F_{\mathcal{V}}^{A_n} P_{\mathcal{V}}^T(n) + (1 - \sum_{\mathcal{V} \in \mathbb{W}^{A_n}} F_{\mathcal{V}}^{A_n}) \rho_s. \tag{3}$$

Similarly, we denote with $P_{\mathcal{V}}^R(s)$ the power consumed by relay s when receiving traffic from the set $\mathcal{V}$ of children, which again can be computed following [14]. Hence, the power consumption of a relay node $s \in \mathcal{S} \setminus \{0\}$ is given by:

$$Y_s = \sum_{\substack{\mathcal{V} \in \mathbb{W}^{A_s} \\ s \in \mathcal{V}}} F_{\mathcal{V}}^{A_s} P_{\mathcal{V}}^T(s) + \sum_{\mathcal{V} \in \mathbb{W}^s} F_{\mathcal{V}}^s P_{\mathcal{V}}^R(s)$$
$$+ (1 - \sum_{\substack{\mathcal{V} \in \mathbb{W}^{A_s} \\ s \in \mathcal{V}}} F_{\mathcal{V}}^{A_s} - \sum_{\mathcal{V} \in \mathbb{W}^s} F_{\mathcal{V}}^s) \rho_s.$$

IV. COMPUTING THE OPTIMAL CONFIGURATION

We propose SOLOR, a utility-based framework for optimizing the configuration of the WLAN. We compute the total

[4] We discuss in Section V how to enforce this with legacy nodes.

utility of a node n that obtains a throughput X_n and consumes Y_n as

$$V_n = U_n(X_n) - L_n(Y_n) \tag{4}$$

In the above, $U_n(\cdot)$ is a concave function that maps user n's throughput to a utility, and $L_n(\cdot)$ is a convex function that maps the energy consumption of user n to an incurred cost. For example, the energy cost could model the effect on the user of the implied reduction in battery lifetime. Both the concave nature of the energy cost and the throughput utility functions derive from the common assumption of diminishing marginal returns [16].

We divide the problem of optimizing the network configuration into two parts. First, we consider that the topology is fixed, and optimize the relay schedule following one of the proposed maximization criterion. In this way, we compute using convex optimization techniques the best performance achievable with a given topology. Second, we address the problem of selecting the topology that provides the overall best performance, leveraging on the previous optimization.

A. Computing the Optimal Relay Schedule

We frame the problem of choosing the relay schedule in terms of choosing a feasible set of fractions $\vec{F}$ that globally maximizes user utilities subject to resource allocation constraints.

1) Feasibility of time fractions and mapping them to relay schedules: In order to guarantee feasibility, we impose the following constraints that guarantee that the fractions chosen are such that, from the point of view of any relay or the AP, the total fraction of time it is required to stay connected to either its children or parent is less than one and thus achievable.

$$0 \le F_{\mathcal{V}}^s \le 1, \forall s \in \mathcal{S}, \forall \mathcal{V} \in \mathbb{W}^s \tag{5a}$$

$$\sum_{\mathcal{V} \in \mathbb{W}^0} F_{\mathcal{V}}^0 \le 1 \tag{5b}$$

$$\sum_{\substack{\mathcal{V} \in \mathbb{W}^{A_s} \\ s \in \mathcal{V}}} F_{\mathcal{V}}^{A_s} + \sum_{\mathcal{V} \in \mathbb{W}^s} F_{\mathcal{V}}^s \le 1, \forall s \in \mathcal{S} \tag{5c}$$

The first term on the left hand side of constraint (5c) is the fraction of time relay s is connected to its parent, and the second term is the fraction of time it serves its children. Note that a relay spends the time that it is neither transmitting or receiving, i.e., the gap in constraint (5c), in sleep mode. Given a feasible set of fractions, $\vec{F}$, many compliant schedules can potentially be constructed. We describe below a deterministic policy to construct a schedule consistent with a given set of fractions that demonstrates clearly that a set of fractions satisfying the above constraints is indeed realizable.

2) Throughput and power consumption limits:

$$X_n \ge 0, 1 \le n \le N \tag{6a}$$

$$\sum_{n=1}^{N} X_n \le C \tag{6b}$$

$$X_n \ge x_n^{\min} \tag{6c}$$

$$Y_n \le y_n^{\max} \tag{6d}$$

Equations (6a) and (6b) constrain that the throughput figures take positive values and that sum user throughput does not exceed the maximum backhaul capacity C. Eq. (6c) specifies a per-node lower bound on the throughput of a node and thus is a lower bound on performance, while (6d) specifies an upper limit on the amount of power each node is willing to expend.

3) Optimization criteria: The SOLOR framework supports a large set of optimization criteria, ranging from overall utility maximizations to allocations based on minimum improvements in performance. Below, we introduce the two optimization formulation used in this paper, which are based on the per-node utility V_n defined in (4). These policies support a wide range of optimizations, which should be tailored to the specific scenario and nodes' willing to collaborate (e.g., home, office, hotspot scenario).

Sum utility maximization: this optimization consists on the maximization of the sum utility of all the nodes in the network, and is formulated as

$$\max_{\vec{F}} \quad \sum_{n=1}^{N} V_n \tag{7a}$$

$$\text{subject to} \quad V_n \geq d_n, \ 1 \leq n \leq N \tag{7b}$$

$$(6a), (6b), (6c), (6d), (5a), (5b), (5c)$$

where d_n is a parameter that specifies how collaborative or selfish the node n is. More specifically, (7b) specifies the trade-off between energy consumption and performance that is acceptable to each node. When a node chooses $d_n \rightarrow -\infty$, the node is collaborative, willing to sacrifice its individual utility in order to maximize the overall utility (for example, in a home network where all the devices share an owner, this might be appropriate). More subtle preferences are also supported, e.g., setting d_n equal to the utility of the node in the default case imposes the constraint that every node must benefit from the relay-based setup.

Maximizing minimum user improvement: the above formulation can enforce that nodes obtain *some* improvement, but these could result very diverse among nodes. In this way, some users could perceive tremendous gain while others see very little and potentially even performance degradation depending on the choice of parameters (we will explore further this issue in Section VI).

Based on the above, in some cases it could be better suited to maximize the minimum user improvement, denoted as g_n. The optimization is formulated as

$$\max_{\vec{F}} \quad g_n \tag{8a}$$

$$\text{subject to} \quad \left(\frac{V_n - V_{n,baseline}}{V_{n,baseline}}\right) \geq g_n, \ 1 \leq n \leq N \tag{8b}$$

$$(6a), (6b), (6c), (6d), (5a), (5b), (5c)$$

where $V_{n,baseline}$ is the node utility in the no relaying case. This ensures a measure of fairness and could be a reasonable criterion in e.g. a public setting where users do not have an intrinsic reason to collaborate.

In this paper, for simplicity we assume long-lived flows, therefore assume the usual convention of log-like utility functions. To account for delay-sensitive flows, the SOLOR framework should be extended by (i) adding a model for the delay under general conditions, following e.g. our previous work [12], and (ii) introducing a different utility function for these flows, e.g., $\log(D_{\max}-D)$, with $D_{\max}$ being a maximum bound on the average delay, and D its value according to the model.

4) Solving the optimization problem: In both the above cases, the optimization problem maximizes a concave objective function under a convex set of constraints and thus admits a unique optimum. It can be used to model a number of scenarios depending on the subset of constraints that are included and the choices of the utility functions and the energy cost function, as we will demonstrate in the sequel. For example, consider omitting constraints (6c) - (6b), and setting $L_n(Y_n) = 0, \forall n$ in (7). Proportional fairness could be modeled by choosing log utility functions, and max-min fairness (when achievable) could be achieved by setting $U_n(X_n) = X_n, \forall n$ and adding the constraints: $X_n = X_1, \forall n$.

Unless otherwise noted, in the rest of the paper we focus on scenarios where utility functions are of the form:

$$U_n(X_n) = \alpha_n log(X_n)$$
$$L_n(Y_n) = (1 - \alpha_n)Y_n,$$

where $\alpha_n \in [0, 1]$ models the per-node priorities of power consumption vs. performance (a high value of α_n prioritizes performance over power consumption and vice-versa).

B. Computing the Relay Topology

Given a topology, the optimization problem above determines the optimal relay schedule. Here, we focus on the problem of computing the relay topology that maximizes overall network utility. In general, this is a combinatorial problem, and efficiently finding the optimal topology does not appear to be possible as the decision of a single node to switch its parent could affect the throughput that can be achieved by all the nodes in the network. We consider three possible approaches to the topology selection problem with varying degrees of complexity:

Brute Force: This algorithm simply tests *all* valid network topologies, solving the optimization problem (7) for each topology, and choosing the topology that maximizes the overall utility. For large networks, especially those with many relays, this approach is not computationally tractable. However, since this brute force search is guaranteed to find the globally optimal solution, we use it to benchmark the other heuristics.

Closest-first: In this simple heuristic, each node associates to the relay to which it has the highest MCS, irrespective of the set of nodes that are connected to that relay, or the quality of the channel between the AP and the relay, as long as the maximum number of hops to the AP is two. Once the topology is chosen, the optimization problem is solved once in order to configure the network. As compare to the previous scheme, this heuristic is extremely simple, but does not take into account the interactions between various key variables of the WLAN.

A "greedy" algorithm: This is a heuristic aiming at balancing the performance of the brute-force approach with the simplicity of the closest-first scheme. The scheme starts with the default topology (i.e., all nodes associated with the AP) and runs in stages. At every stage, the new topologies to consider are only those in which one node changes its parent; the heuristic solves the optimization problem for each of these alternatives and picks the topology that maximizes the utility.[5] Note that since the utility is bounded, and the overall utility increases monotonically as the heuristic progresses, it is guaranteed to finalize.

In the sequel, we show that for those scenarios in which we could perform the exhaustive searches in the configuration space (Figs. 6 and 7), results show that the heuristic provides very similar gains to those resulting from the brute-force search. These results suggest that the use of other topology creation algorithms would not bring substantial improvements in terms of performance, although it may reduce the computational cost –we leave this as part of our future work.

C. Bi-Directional Traffic

Note that while we focus on the case of uplink traffic for simplicity of exposition, the above problem formulation can also be used to model the scenarios with bi-directional traffic. In this case, utility function are defined separately for each of the uplink and downlink flows. In each time fraction $F_{\mathcal{V}}^s$ that we consider, both uplink and downlink traffic are in contention, and a throughput model similar to the one defined in Section III is used to calculate the throughput of both the uplink and downlink flows. Here, we separately define the average uplink and downlink rates received by a node in each time fraction, i.e., we replace the rates $R_{\mathcal{V}}(n)$ with the uplink and downlink versions $R_{\mathcal{V}}^{\text{UL}}(n)$ and $R_{\mathcal{V}}^{\text{DL}}(n)$. The above rates are still be calculated using [12] with the AP/relay being another contending node in the network and the power consumption model is similarly modified. We present results for the case of bidirectional traffic in Section VI.

V. Protocol Details

This section describes the operation of a SOLOR node to derive and apply a common configuration. For simplicity, we decided to implement SOLOR in a distributed manner, although a centralized scheme could also be used the configuration distributively. The operation of a SOLOR node is illustrated in Fig. 3.

A. Protocol overview

When powered on, a SOLOR node multicasts its presence to the rest of SOLOR nodes (if any), following the communication scheme detailed below. Then, to estimate the topology of the network, it continuously snoops the transmissions from all nodes (legacy and SOLOR) and collects the MCS used to transmit to the AP and the SNR that it measures, using an exponentially weighted moving average to filter out small fluctuations. Following [17], the SNR info serves to

---

[5]We provide in the Appendix a formal description of the algorithm.

estimate the MCS a node will use when transmitting to the SOLOR node, which completes the estimation mechanism of the network conditions.

Based on the above mechanism, a SOLOR node compares the network conditions vs. the information utilized in the last re-configuration. In case conditions change (or when the first time the node is powered on), it multicasts a re-configuration message, which is extended by the other SOLOR nodes as they forward it with additional information (as described next). The re-configuration is triggered with a `net_reconf_req` message containing: (i) SOLOR ID, (ii) the estimated network conditions (i.e., MCS between pair of nodes), (iii) the SOLOR operation parameters (α values), both from the node and clients, based on a default set of parameters or an estimation of the type of device (e.g., based on their MAC addresses, or the "Device Type" attribute of the Wi-Fi Protected Setup), (iv) its per-station power consumption figures ($\rho_{tx/rx/id/s}$) and those of the legacy clients it can hear (again, using a pre-defined set of parameters, or after an estimation), (v) the channel list where the relay can operate, and (vi) the timestamp when the re-configuration is issued.
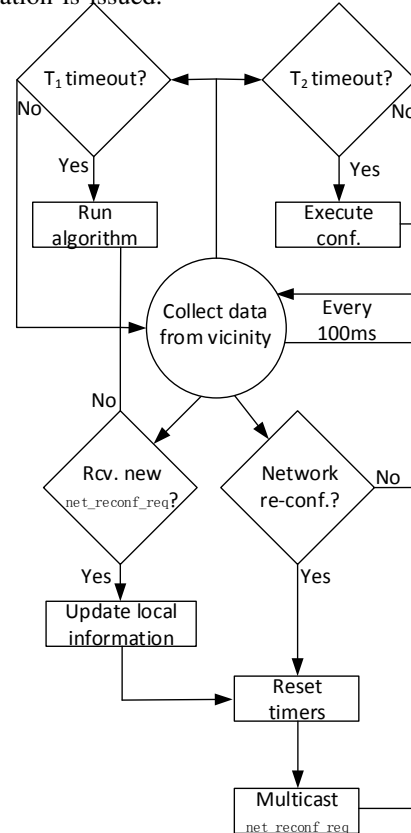

Fig. 3: Operation of a SOLOR node.

A SOLOR node that receives a *new* `net_reconf_req`, updates its local database, updates the `net_reconf_req` by adding its local data, and multicasts this updated message with its own SOLOR ID. This simple controlled flooding protocol allows the SOLOR nodes to have a global view of the scenario, i.e., each relay knows the MCS for all potential links, and the individual preferences and per-state power consumption figures of the nodes ($\alpha_i, \rho_{tx}^i, \rho_{rx}^i, \rho_{id}^i, \rho_s^i$), to run the algorithm with the same shared information. SOLOR relays record the

timestamp of the initial re-configuration message, but do not immediately initiate the computation of the optimal configuration; instead, they wait T_1 seconds with no new messages to trigger the computation. This configuration is committed T_2 seconds after the timestamp, which guarantees synchronization between SOLOR nodes. Note that T_2 has to be longer than the time it takes for the re-configuration message to reach all relays, plus the time to compute the optimal configuration (the complexity of this computation is analyzed in Section VI-D).

B. Communication between SOLOR nodes

The operation of SOLOR relies on a mechanism to reliably deliver messages across all relays. To this aim, in our experiments we leverage the default multicast operation, as we found that it results extremely reliable due to the use of a robust MCS (i.e., 100% delivery rate). Still, for harsher network conditions, we could easily extend SOLOR with one of the mechanisms from the Group Addressed Transmission Service described in the recent 802.11aa standard, which specifies more reliable multicast services, as there is an implementation readily available [18].

The direct communication between SOLOR nodes, when one is acting as a parent for the other, results immediate, as they share the same schedules and therefore the transmitter knows when the intended destination can receive the data. However, when SOLOR nodes communicate through the (legacy) AP, they need to be associated with the AP long enough, so the multicast transmission is successfully forwarded from one SOLOR node to the other. To this aim, we fix a minimum amount of time that all clients have to be simultaneously connected to their parent, i.e., $F_{\mathcal{C}_{A_n}}^{A_n} = 10$ ms $\forall n \in \mathcal{S}$, and schedule multicast messages at the beginning of this time fraction.

C. Computing a feasible schedule

To find a feasible schedule for the optimal configuration, we start with the relays one hop away from the AP, and then move one hop at a time (the schedule of the relays at the same number of hops from the AP can be computed in any order). For each relay $s \in \mathcal{S}$, we impose a deterministic ordering of the sets in $\mathbb{W}^s$, based on the size of the set (note that $\mathbb{W}^s$ does not include the empty set) and using the smallest node identifier (its MAC address) as a tiebreaker. We use this ordering of $\mathbb{W}^0$ to arrange the fractions $F_{\mathcal{V}}^0, \mathcal{V} \in \mathbb{W}^0$, which specifies the time periods when the children of the AP have to contend for access. Next, for each relay s one hop from the AP, we determine the rest of its schedule by splitting the time that s is not sending to the AP into the time fractions $F_{\mathcal{V}}^s$, ordered after the set $\mathbb{W}^s$ as well. The time left at the end of the schedule is the fraction of time the relay spends in sleep mode. Following this methodology, we find a feasible schedule that fulfills the requirements of the solution to the optimization problem. Fig. 4 illustrates the above schedule computation for a scenario with five relays (R1–5) and two legacy clients (C1–2) with a 2-hops topology.
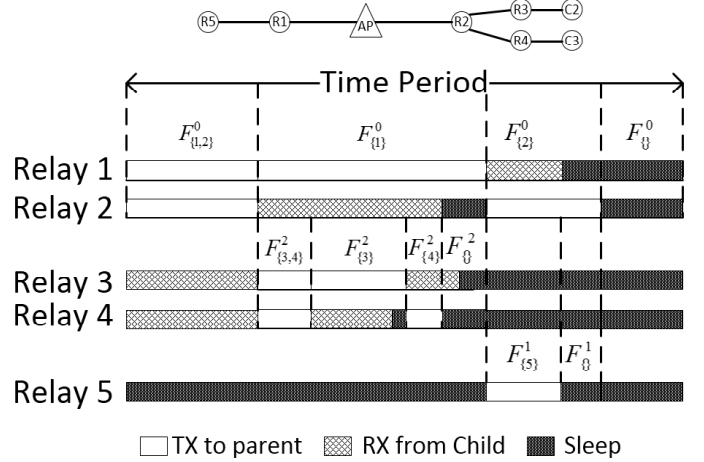

Fig. 4: Relay schedules computation for a 2-hops topology.

D. Applying the new configuration

Once the optimal configuration is found, the links between nodes must be configured. To force legacy nodes to disassociate from the AP and associate to the relay, we use a simple scheme based on the behavior of most wireless network managers, which consists on the relay *forging* a disassociation message as if it were sent from the AP, thus forcing the legacy node to re-scan the network to look for the *best* AP announcing the same SSID to associate with. This AP should be the relay node, as it supports the use of better MCS and therefore has better link quality. For simplicity, in our experiments of Section VII the client obtains a new IP address after the re-association, but this could be prevented if the SOLOR relay sends a "gratuitous ARP" to the AP [5].

Finally, we need to ensure that legacy nodes go to sleep or, at least, do not transmit while the relay is not available (either sleeping or sending data to its parent). For simplicity, we use the *Notice of Absence* (NoA) [9] protocol, specified for WiFi Direct and already present in many current devices (e.g. Android phones), which allows the relay node to send a unicast packet to its attached clients with the relay's sleep schedule. We confirmed that other schemes also work, e.g., sending `null data` frames with the Network Allocation Vector set to the time the AP is not available, which enables the node to sleep for that period of time (we confirmed that old NICs overhearing all traffic do not go to sleep, but do not transmit neither).

VI. PERFORMANCE EVALUATION

In this section we quantify the performance improvements that can be achieved using SOLOR via numerical analyses, while in Section VII-VIII we confirm the good match between these and experimental results. The simple case of a network with two nodes, one with relay capabilities, has already been discussed in Section I (this was the only case considered in [5]). In what follows, we first analyze the case of a two-relay network like the one depicted in Fig. 2, with homo- and heterogeneous per-node settings for the performance vs. energy trade-off, and then address the case of random topologies with larger number of nodes and relays.

A. A two-relay, three node network of homogeneous nodes

Uni-directional traffic: We first consider the scenario illustrated in Fig. 2, in which nodes 1 and 3, both with relays capabilities, can transmit to the AP at $R_1 = R_3 = 48$ Mbps. Node 2 can transmit to the AP at $R_2 = 6$ Mbps, and could send traffic to nodes 1 and 3 at $R_{21} = 18$ Mbps and $R_{23} = 48$ Mbps, respectively. We first obtain, as a benchmark, the (equal) throughput, X(default), achieved by each node in the "default" case, i.e., when all nodes directly transmit to the the AP. We analyze the performance of the SOLOR framework under the following optimization criteria:

- The "energy-optimal" configuration, obtained by setting $\alpha_i = 0\ \forall i$ and $x_n^{\min} = X(default)$.
- The "max-min" optimal configuration, i.e., maximizing the lowest individual throughput.
- The "proportional-fair" (PF) configuration, without energy considerations ($\alpha_i = 1\ \forall i$) and with energy considerations ($\alpha_i = 0.25\ \forall i$).

Note that the topology chosen by our framework is identical to the one depicted in Fig. 2 in all the cases, and also coincides with the optimal topology.

Fig. 5a depicts the throughput and power consumption of each node in the different settings. The results demonstrate the gains that can be achieved by SOLOR along the two dimensions of interest, depending on the preferences of the nodes. For example, in the case of PF with no energy considerations, the overall throughput increases by 170%, with each node benefiting substantially (note that the share is almost purely fair). However, in this case, node 3 acting as the relay for node 2 does consume higher power than in the default scenario. The fact that the performance obtained with the "max-min" and "proportional-fair" criteria is the same is particular for the optimal topology and scheduling policy computed for this specific scenario and does not respond to a general conclusion as we can see in Fig. 5c. When the nodes are highly energy constrained, SOLOR enables power savings of 74% with no throughput reductions.

Bi-directional traffic: Using the same scenario as before, we set up now three new flows from the AP towards each of the nodes competing with the three uplink flows. The results are shown in Fig. 5b. Note that the "Default" configuration shows an asymmetric performance due to the fact that the three downlink flows act as one entity when competing against the three uplink flows. The remaining configurations, however, show a symmetric behavior because (i) the configuration imposes certain fairness criteria to all the flows, and (ii) we remove the asymmetric competition among flows, i.e., Relay 1 and 3 never compete because of the coordination and Node 2 is the only Relay 3's child operating in a non-interfering channel.

Multi-hop relaying: To demonstrate the effectiveness of SOLOR in scenarios that call for multi-hop relay topologies, we consider again the network in Fig. 2, with the link from node 3 to the AP degraded to $R_3 = 6$ Mbps, emulating for example the presence of an obstacle. In this case, the best topology for all the settings considered (and the one chosen by SOLOR) is one in which node 3 accesses the AP through

node 1 at $R_{13} = 48$ Mbps while continuing to relay for node 2. The results for this scenario are depicted in Fig. 5c, and show the same qualitative behavior as in the earlier case. The raw throughput (and power savings) achieved, in this more hostile environment, is not as high as in the earlier scenario, however the gain over the default case is still significant (160% throughput increase under PF, and 60% energy savings in the energy-optimal case).

B. A two-relay, three node network of heterogeneous nodes

One of the key features of SOLOR is its ability to support individual node preferences. We explore the effect of the parameter α and the ability of SOLOR to adapt, focusing from this point forward on the PF criterion. We consider again the WLAN depicted in Fig. 2 without the obstacle between node 3 and the AP, and assume that node 1 is not power constrained (e.g., connected to a wall socket) and thus has $\alpha_1 = 1$. We examine a range of scenarios where the sensitivity of nodes 2 and 3 to power consumption progressively increases as they become increasingly power constrained (mobile devices).

Fig. 6 depicts the gain achieved by SOLOR over the default scenario as the value of $\alpha_2 = \alpha_3$ increases. The results demonstrate that SOLOR is able to adapt to different per-node preferences on the trade-off between power and throughput. Indeed, Fig. 6 illustrates that when throughput performance is critical, and nodes 2 and 3 prioritize throughput over power savings, the topology chosen is the one illustrated in Fig. 2 that favors higher throughput ($R_{21} < R_{23}$). However, as node 3 becomes increasingly power constrained, the topology chosen switches to one in which node 2 reaches the AP through node 1, as shown in Fig. 6, enabling node 3 to save power. Note that in the power hungry scenarios, the gain achieved by SOLOR explodes as nodes are able to obtain their desired throughput in a highly energy-efficient manner.

Guaranteeing minimum gains: we focus on the same scenario as before, where Relay 1 is forced to relay for Node 2, and $\alpha_1 = 0.25$, $\alpha_2 = 1$ and $\alpha_3 = 0.75$. Instead of the overall utility (problem described in (7), now we compute the *relative* utility gains over the baseline case when the sum utility is maximized, with the results depicted in Fig. 7, bottom ("Sum Utility"). We can see that Relay 3 does not have a strong incentive to collaborate, as its relative performance has worsened to maximize the sum utility gain. To address this, we next use the optimization problem described by (8), which introduces max-min fairness in relative utility gains. The effectiveness of this approach is confirmed by Fig 7, top ("Min Gains"), where no node is experiencing a decrease in performance but instead all nodes improve their utility by at least 30%.

C. Random network topologies with multiple relays

Finally, we analyze the performance improvements of SOLOR in random topologies consisting of different number of nodes and relays. The generation of a random deployment consist of the following steps: (i) we assume a square area of size 20 m×20 m, in which the AP is located in one of the corners; (ii) we randomly deploy N nodes in the area,
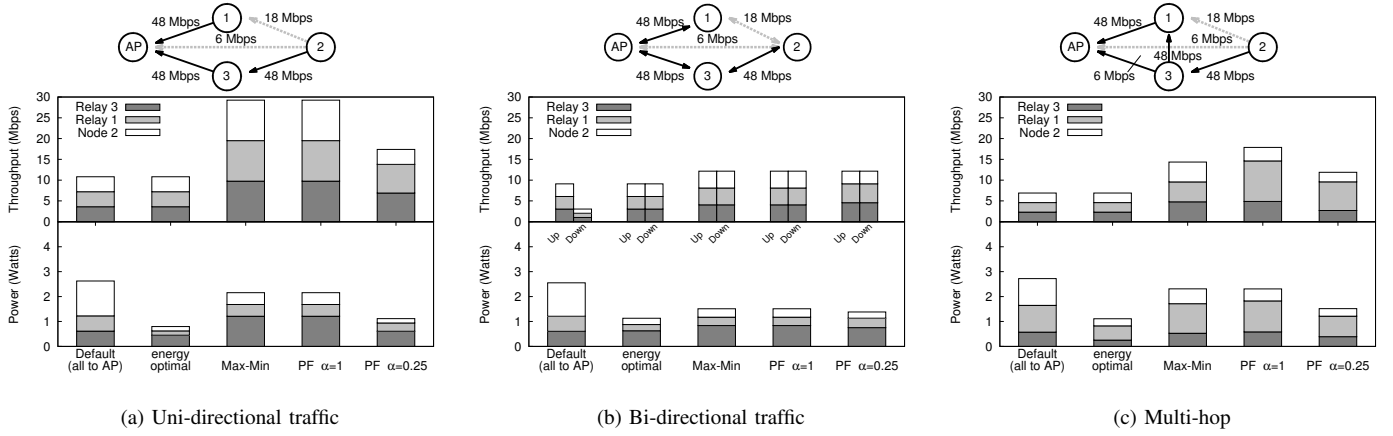
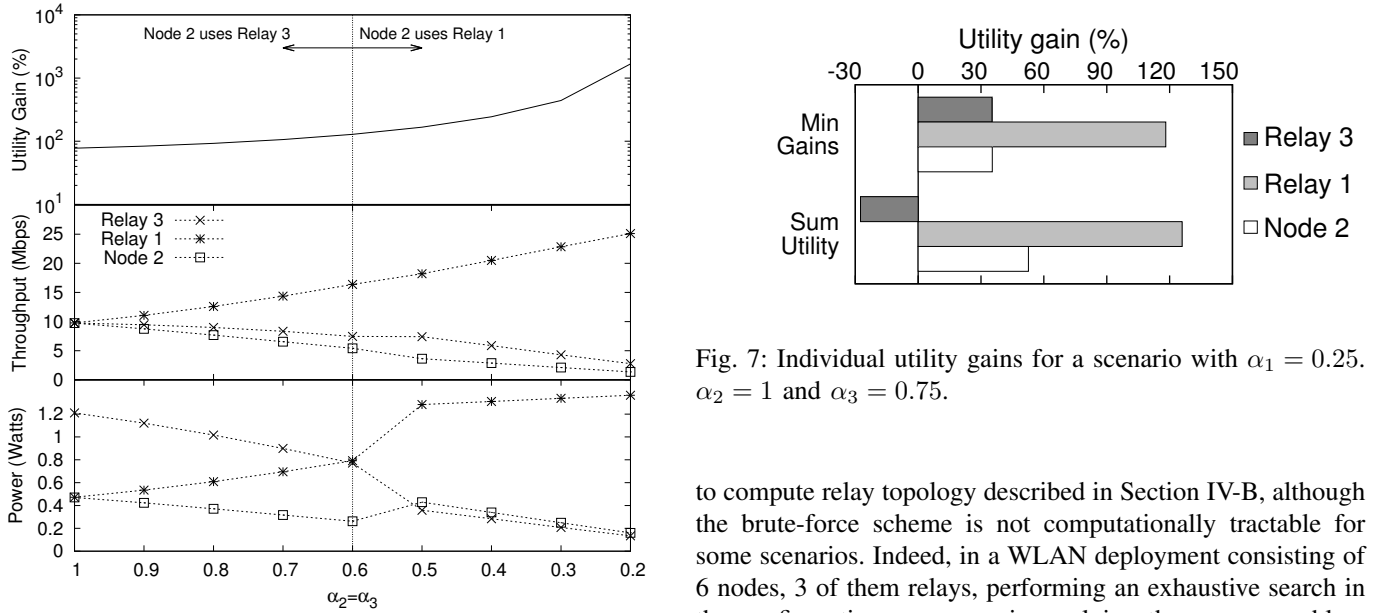Fig. 5: A two-relay, three node network of homogeneous nodes

(a) Uni-directional traffic (b) Bi-directional traffic (c) Multi-hop

Fig. 6: WLAN performance for the deployment of Fig. 2 and different configurations of $\alpha_{2,3}$.

Fig. 7: Individual utility gains for a scenario with $\alpha_1 = 0.25$. $\alpha_2 = 1$ and $\alpha_3 = 0.75$.

following a 2-D Gaussian distribution centered on the AP and with $\sigma = 10$ m (if a node falls outside the considered area, it is re-deployed).[6] (*iii*) we randomly pick R out of the N nodes, as being relay-capable; (*iv*) finally, based on the distances between nodes (we apply the log-distance path loss model with shadowing parametrized for an office environment with hard partitions [21]), we use the MCS vs. SNR curves provided in [17] to obtain the transmission rates between each pair of nodes.

For each scenario, we first compute the WLAN performance for the "default" case, and then the performance when using SOLOR. We compare the performance of the three approaches

to compute relay topology described in Section IV-B, although the brute-force scheme is not computationally tractable for some scenarios. Indeed, in a WLAN deployment consisting of 6 nodes, 3 of them relays, performing an exhaustive search in the configuration space requires solving the convex problem almost 400 times, while our greedy scheme reduces this number to 60. To obtain statistically significant results, we generate as many random topologies as required to obtain 95% confidence intervals whose size is less than 10% of the mean.

Impact of network size: We first analyze performance with varying number of nodes in the WLAN, when half of them are relay-capable. Like in the previous section, we stick to the PF optimization, for two different choices of α_n, namely, $\alpha_n = 1$ (indifferent to power saving) and $\alpha_n = 1/7$ (sensitive to power consumption). For each scenario we compute the gain in the overall utility as well as the gains in throughput and power consumption relative to the default case.

The results are depicted in Fig. 8, which demonstrates that SOLOR is able to improve performance in all the considered scenarios, with gains that increase as the size of the network increases –the larger the network, the more opportunities to find better configurations. According to the results, the utility improvements of the greedy scheme are very similar to those of the brute-force approach, despite the reduced computational complexity. In contrast, associating to the closest relay seems

---

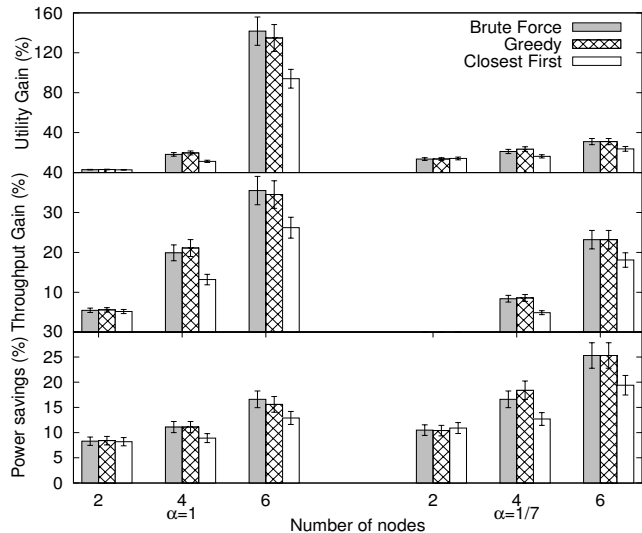[6]Although there are well-known random generators available, such as the Hyacinth-Laca tool used in e.g. [19], [20], these are typically used for the case of large muti-hop wireless (mesh) networks, while our focus is on smaller-sized deployments.

Fig. 8: Performance improvements for different network sizes, when half of the nodes has relay capabilities.

Fig. 9: Impact of the proportion of relays.[7]

to be effective in small scenarios, but fails to extract the maximum gains in larger settings. Finally, the figure also illustrates how setting α_n appropriately can calibrate the trade-off between throughput performance and power consumption.

Impact of relay density: Next, we analyze the performance of SOLOR as the proportion of relay-capable nodes changes, for topologies consisting of five nodes. The results are depicted in Fig. 9, and show that when the relative number of relays is low (1 out of 5), the performance improvements are low, a result that is not surprising as the relay is chosen by randomly picking one of the five nodes deployed, rendering it ineffective in most cases. Despite this, the results show that even when only two of the nodes are relay-capable, the performance improvement is significant (e.g., throughput gains around 20% for $\alpha_n = 1$), and these can grow up to 100% improvement in the case of all-relay networks. When $\alpha = 1/7$, power savings on the order of 80% are achieved on average in all-relay networks while overall throughput performance is also improved by 20%. Finally, the results from the greedy algorithm are very similar to those from the brute-force approach, whose computational complexity is prohibitive for topologies with more than three relays (note that given our requirements on the size of the confidence interval, for these configurations we have to run more than 1000 random topologies).

The results in this section demonstrate the effectiveness of SOLOR in maximizing performance in very diverse heterogeneous settings. In the next section, we describe a preliminary deployment of SOLOR in a real-life testbed consisting of seven machines that validates our findings.

D. Computational Cost

We next assess the computational complexity of computing the optimal configuration. To this aim, we set up three different

scenarios and run each of the three algorithms proposed before to compute the network configuration, measuring the average number of calls to the optimizer function (i.e., the number of topologies evaluated). We run the experiment using a different random topology for each scenario as many times as needed to obtain 95% confidence intervals within 10% of the shown average. Results are summarized in Table I, showing that, as expected, "Closest-first" only requires one optimization, while the exhaustive search needs up to 400 calls for the most complex scenario. The heuristic algorithm, in contrast, finds near-optimal solutions up to 7 times faster.[8]

TABLE I: Computational cost of the algorithms.

Algorithm	3 legacy 1 relay	3 legacy 3 relays	6 legacy 3 relays
Brute-force	9	34	397
Greedy	6	16	59
Closest-first	1	1	1

VII. EXPERIMENTAL EVALUATION

Here we describe the results from a first implementation of the SOLOR framework. Our 802.11g testbed, represented in Fig. 10b, is comprised of seven nodes, all using Ubuntu 11.10 with kernel 3.00. There are four legacy nodes, one of which is the AP, and three relay-enabled nodes. The legacy nodes are standard laptops equipped with WLAN cards based on the Atheros AR5413 chipset, using the `ath5k/mac80211` wireless subsystem, while the relay-capable nodes are desktop machines, each equipped with *two* WLAN cards based on the Atheros AR922X chipset and using the `ath9k/mac80211` subsystem. We decided, for simplicity, to use two NICs (Network Interface Cards) to emulate a single NIC with the ability to serve as AP on one channel and to connect to an

---

[7]Note that some of the Brute Force results are not shown due to its heavy computational load in the cases when there exist many potential links.
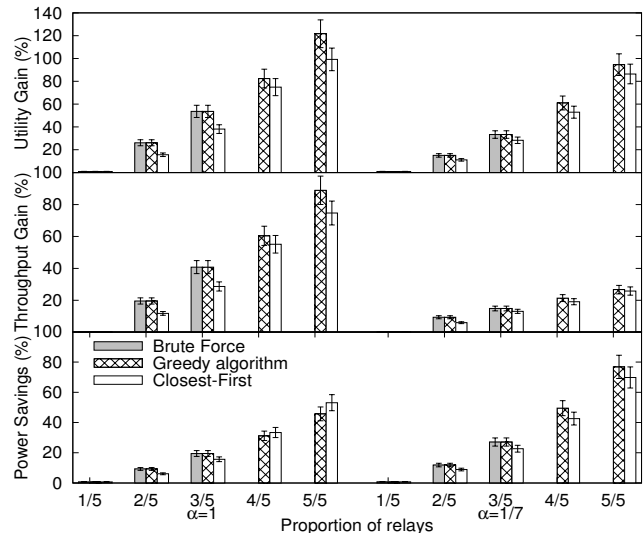
[8]We present our results in terms of calls to the optimization function to be SW/HW agnostic. For a dual core laptop with 2GB RAM, 2Ghz processors, and Ubuntu 12.10, solving the convex problem with a non-optimized library requires between 50 ms (for the case of one relay) and 500 ms (for the case of three relays and 6 legacy nodes).

AP on a different channel, as existing open-source drivers do not support this feature yet.[9] On the other hand, our implementation will not require any modification once this feature becomes available. Note that, throughout our experiments, we take great care in confirming that only one of the two NICs is active at any point in time.

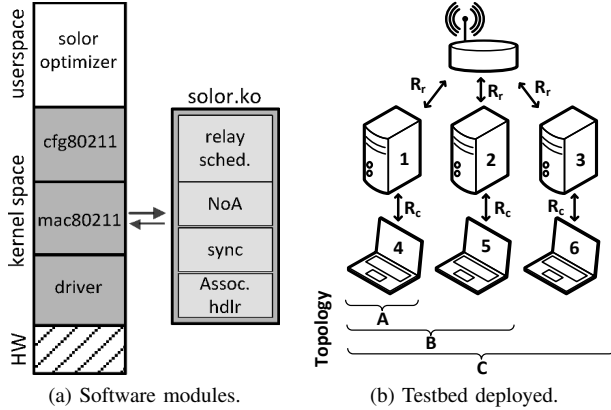(a) Software modules.   (b) Testbed deployed.

Fig. 10: Implementation architecture.

A. Implementing SOLOR

In order to implement SOLOR, three main functionalities are required: a) to analyze the WLAN deployment and compute the optimal configuration; b) to implement the resulting relay schedules; c) to force legacy nodes to connect to the proper relay and to sleep when needed. This is achieved by the software architecture depicted in Fig. 10a, consisting of a user-space application that computes the optimal configuration, and a kernel module (`solor.ko`) to interact with the Linux wireless subsystem.

The optimal configuration of the network is independently computed by the *SOLOR optimizer* of each SOLOR node; given the policies described in Section V, using MAC addresses as node IDs, this will result in all relays computing the same joint schedule with fractions $\vec{F}$. Unless otherwise stated, the individual preference parameters α_i's are set to 1, and the timers are set to $T_1 =500$ ms and $T_2=1$ s.

To implement the schedule, the `solor.ko` module builds on the *synchronization* provided by beacon frames sent by each parent, and triggers the corresponding notifications to the *relay scheduler*. This one reacts upon a notification and apply the required context change in the driver through mac80211 (i.e., transmit buffered data, received and buffer data, or sleep). The setup of the links computed by the new topology is handled by the *Association handler* which, as explained in Section V, forges a disassociation message and announces to the network as an AP (which will have better SNR with the target clients). Finally, `solor.ko` implements the *Notice of Absence* protocol to advertise the sleeping policies to the relay's clients.

[9]Previous works, e.g. [22], describe the required ad-hoc modifications to support this for the case of the MadWiFi driver, which is based on a proprietary API to interact with the hardware.

B. Performance Evaluation

Static conditions: We start our experimental evaluation by measuring the throughput performance of different static settings with a fixed topology, in order to validate the results from the previous sections. To this end, we consider the three topologies depicted in Fig. 10b and different settings of the transmission rate between the laptops and the relays (denoted as R_c), and the rates between the relays and the AP (denoted by R_r), and compare the per-node throughput figures X_n obtained in the testbed with the analytical ones both for uni-directional and bi-directional flows. The results are depicted in Table II, showing that in all cases the experimental figures match remarkably well the results from the analytical model, which are provided in parenthesis (the same conclusions are obtained for different values of α_n, omitted for space reasons).

TABLE II: Per-node throughput (in Mbps) for the topologies in Fig. 10b.

Topo.		R_c, R_r (Mbps)	X_1, X_2, X_3 (X_{model})	X_4, X_5, X_6 (X_{model})
Uni-directional	A	48, 48	14.60, -, - (14.62)	7.31, -, - (7.31)
		48, 24	14.22, -, - (14.62)	5.51, -, - (5.57)
		24, 24	8.7, -, - (9.00)	4.65, -, - (4.5)
	B	48, 48	7.46, 7.42, - (7.31)	6.98, 7.12, - (7.31)
		48, 24	7.64, 7.63, - (7.31)	7.16, 7.23, - (7.31)
		24, 24	4.11, 4.92, - (4.50)	4.32, 4.10, - (4.50)
	C	48, 48	5.30, 4.21, 4.42 (4.87)	3.80, 4.12, 3.84 (4.87)
		48, 24	4.53, 4.98, 4.41 (4.87)	4.30, 4.56, 4.52 (4.87)
		24, 24	2.92, 3.22, 3.15 (3.00)	2.63, 2.52, 2.77 (3.00)
Bi-directional	A	48, 48	Up: 6.59, -, - (6.08)	Up: 2.49, -, - (3.04)
			Dwn: 6.01, -, - (6.08))	Dwn: 2.96, -, - (3.04)
	B	48, 48	Up: 3.21, 2.94, - (3.04)	Up: 2.81, 3.05, - (3.04)
			Dwn: 3.09, 3.34, - (3.04)	Dwn: 2.71, 2.98, - (3.04)
	C	48,48	Up: 1.75, 2.11, 1.87 (2.03)	Up: 2.22, 1.89, 2.08 (2.03)
			Dwn: 2.12, 1.99, 2.23 (2.03)	Dwn: 1.87, 2.04, 1.91 (2.03)

Dynamic conditions: We next assess the performance of SOLOR in a dynamic scenario, in which nodes activate the relaying functionality in real-time and thus the topology changes over time. Nodes 1–3, which do not have the relay functionality activated at the beginning of the experiment, can transmit to the AP at 48 Mbps, while nodes 4–6 transmit to the AP at 6 Mbps, and could transmit to nodes 1–3 at 48 Mbps. Our experiment is divided in stages of approximately 20 seconds each. During the first stage, all nodes are transmitting to the AP, this being the "default" scenario; during the second stage, node 1 enables the SOLOR functionality and as a consequence starts relaying traffic for nodes 4–6; in the third stage, node 2 also enables the SOLOR functionality and relays the traffic from node 6, while node 1 keeps relaying for nodes 4 and 5; finally, in the last stage, node 3 is also enabled as a SOLOR node and, as a consequence, each relay-enabled node serves one client, i.e., the topology C depicted in Fig. 10b.

We display the evolution of the per-node throughput figures over time in Fig. 12 (top), in which the transient caused by the re-association periods can be easily identified. The corresponding overall utility of the WLAN is depicted in the bottom subplot, along with the theoretical values. We conclude from this experiment that enabling the relay functionality supports increasing the utility of the network, with a good match between experimental and analytical results, and that the
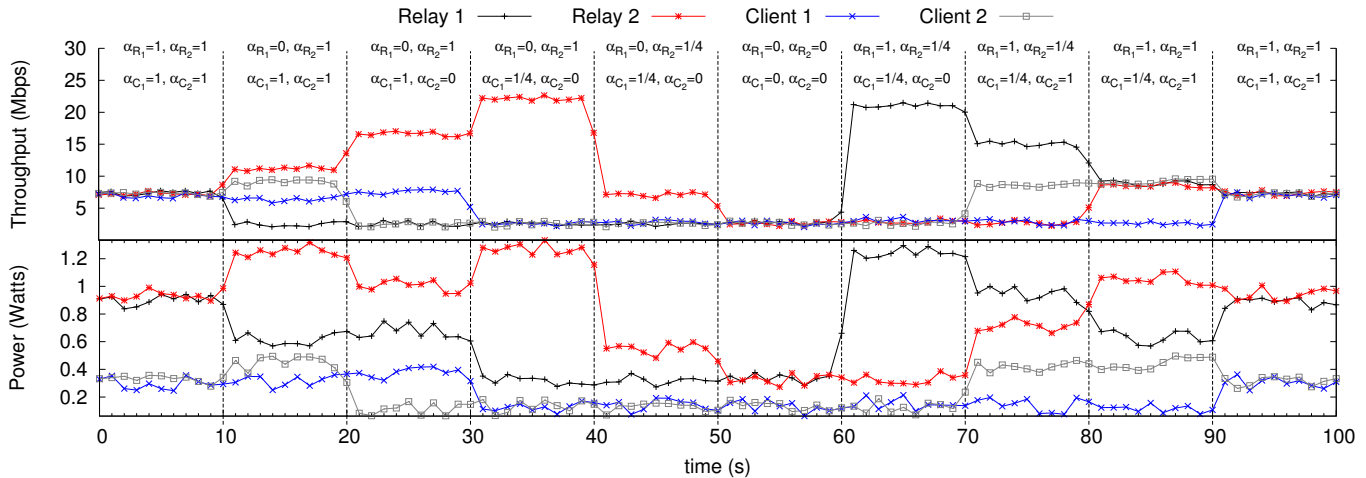
Fig. 11: Per-node preferences

SOLOR framework is easily implementable using commercial, off-the-shelf hardware.
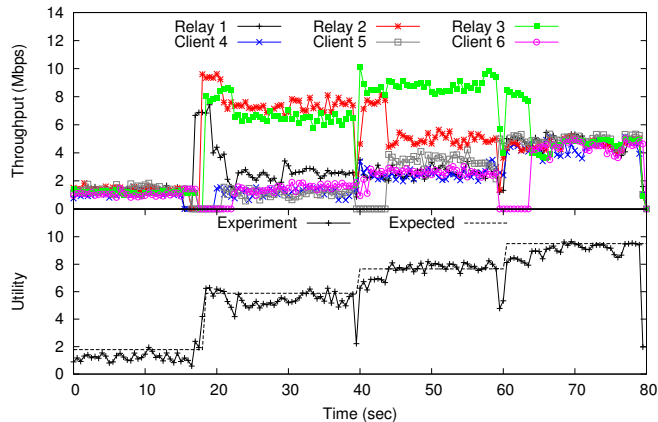

Fig. 12: Dynamic experiment

Energy performance and per-node preferences: We now evaluate our prototype with dynamic individual preferences and show the results in Fig. 11. For the sake of readability, we only use Relay 1-2 and Client 4-5 and initialize a static topology with 1 and 2 serving 4 and 5, respectively. We start off by selecting $\alpha_i = 1$, $\forall i$, just like we did in our previous evaluations and we vary each node preferences sequentially every 10 seconds, illustrating that the larger the α, the more emphasis is given to throughput perforamnce. We conclude from this experiment that SOLOR succeeds at tuning the per-node preferences in the throughput vs. power consumption trade-off.

VIII. COMPARISON WITH OTHER APPROACHES

In order to compare the performance of SOLOR with other approaches proposed in the literature, we set up an illustrative topology as depicted in Fig. 13. In this scenario, 6 nodes transmit data to the AP, nodes 4-6 use a MCS=48 Mbps and nodes 1-3 a low MCS=6 Mbps, though the latter could transmit at 48 Mbps had they used one of the nodes closest to the

AP as a relay. Fig 14 shows the performance of the network in terms of total throughput, power and total net utility as described in S. IV, equation 7a with a homogeneous parameter of $\alpha = 1$, i.e., the utility is $\sum_{n=1}^{6} \log(X_n)$. The comparison is done by means of numerical analysis (lines) and experimental evaluation (points).

In order to compare SOLOR with other mechanisms, we vary the number of *smart* nodes, i.e., stations that have the ability to enable SOLOR, CRS [8], and/or soft-repeater [5]. Initially, all of the eight stations are smart and, thus, represent the best-case scenario for this 6-node topology. Then, sequentially, we deactivate each node's intelligence (becoming a regular *legacy* node) starting from node 1 and ending with node 6. Note that for this last case, all of the nodes are legacy IEEE 802.11 stations and therefore all the nodes transmit directly to the AP (half of them at a low MCS). We have chosen a max-min fair scheduling for this experiment as it is the one proposed in all three papers, though any other would show relatively similar gaps in performance. The conclusions that we can get out of our results are threefold:

Relay coordination: In CRS, the modified AP is able to provide a fair allocation by granting each station a certain number of tokens. In Soft-repeater, however, the AP does not take scheduling decisions for the relayed-relay pair (in fact, it can be a legacy AP in its simplest version), and therefore the scheduling is done by the relay that is only aware of the presence of its clients. For this reason, CRS performs better than soft-repeater with the presence of multiple relays. SOLOR, in turn, provides the best performance because (i) is able to provide a good scheduling since all relays have knowledge of each other's presence and (ii) the coordination among relays reduce the number of collisions, particularly when there are many smart nodes in the network.

Presence of legacy nodes: The performance gap is larger if we reduce the ratio of smart nodes in the network. This is so because with both, CRS and soft-repeater, a legacy station can't be relayed without implement modifications on them, a limitation that SOLOR does not suffer of.

Energy performance: Even though the selected utility does

not target energy optimization (i.e., $\alpha_i = 1 \ \forall i$), SOLOR substantially improves the energy performance of the network with respect to Soft-repeater thanks to the efficient utilization of the sleeping schedules. Moreover, although CRS also supports sleeping policies, SOLOR also betters the energy performance with respect to CRS in most of the scenarios. Only when all the nodes are "smart", CRS shows a light energy improvement due to the important throughput reduction.

Finally note that, in order to be able to compare, the AP of this experiment has the intelligence required (e.g. to run CRS), a modification that is not required by SOLOR.
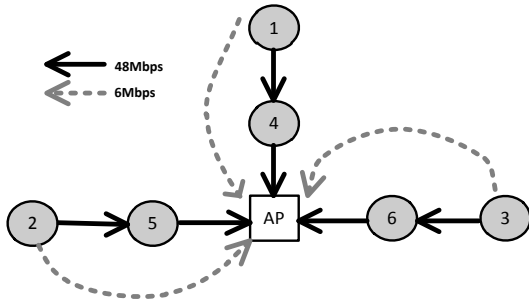

Fig. 13: Topology for comparison of SOLOR, CRS [8] and soft-repeater [5].

Fig. 14: Comparison of SOLOR, CRS [8] and soft-repeater [5].

Now, to explore the trade-off between power consumption and throughput further, we analyze numerically in Fig. 15 the performance of each of the nodes in the "all smart nodes" scenario of the previous experiment (the best-case scenario for CRS and Soft-repeater), computing the throughput and expected lifetime of the devices (assuming a 1440 mAh battery) for the same strategies of Fig. 14. As the top figure shows, Soft-repeater and CRS results in very similar throughput values, while SOLOR improves performance by approximately 25% when the parameter α is set to 1. On the other hand, the bottom part of the figure shows that SOLOR and CRS perform very similarly in terms of lifetimes, with the latter providing slightly longer times (about 8%), while the lifetimes provided

by Soft-repeater are well below one third of the others. In this way, SOLOR is able to exchange 8% in energy consumption (when compared to CRS) for an 25% increase in throughput. Moreover, if the underlying data flows only require a certain bit rate (e.g., video delivery), SOLOR is capable of trading the unused capacity with further energy savings (for instance, by increasing the α).
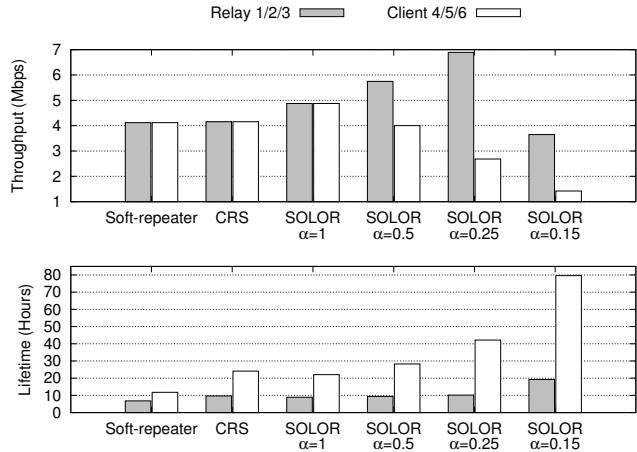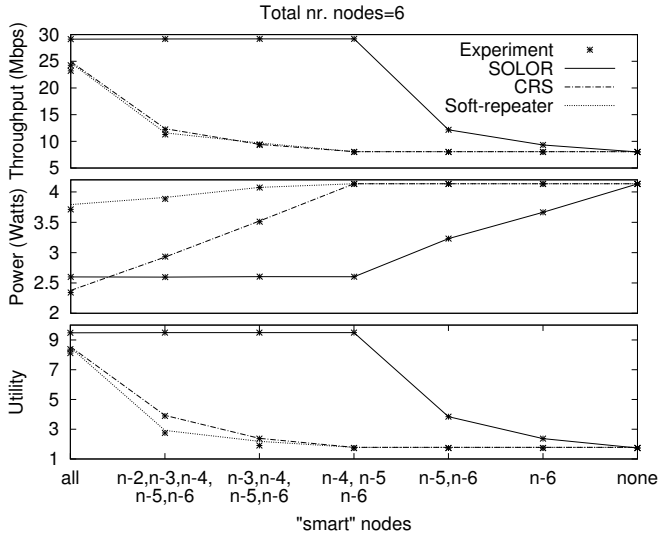

Fig. 15: Comparison of SOLOR, CRS [8] and soft-repeater [5] for the "all-smart" scenario.

IX. CONCLUSIONS

In this paper we presented SOLOR a novel Self-Optimizing, Legacy-Compatible Opportunistic Relaying framework which addresses the *rate anomaly* problem by taking into account three major considerations to achieve an efficient deployment in real-world systems: 1) relaying could imply increased power consumption, and nodes might be heterogeneous, both in power source (e.g., battery-powered vs. socket-powered) and power consumption profile; 2) similarly, nodes in the network are expected to have heterogeneous throughput needs and preferences in terms of the throughput vs. energy consumption trade-off; and 3) any proposed solution should be backwards-compatible, given the large number of legacy 802.11 devices already present in existing networks.

SOLOR jointly optimizes the *topology* of the network, i.e., which nodes associate to each relay-capable node; and the *relay schedules*, i.e., how the relays split time between the downstream nodes they relay for and the upstream flow to an AP. The proposed framework has been evaluated considering a large variety of scenarios and different node performance/power consumption trade-off preferences and its feasibility demonstrated through test-bed experimentation using off-the-shelf equipment. Our results show that SOLOR greatly improves network throughput performance (more than doubling it) and power consumption (up to 75% reduction) even in systems comprised mostly of vanilla nodes and legacy access points.

REFERENCES

[1] G. Hiertz, D. Denteneer, L. Stibor, Y. Zang, X. Pérez-Costa, and B. Walke, "The ieee 802.11 universe," *Communications Magazine, IEEE*, vol. 48, no. 1, pp. 62 –70, january 2010.

[2] M. Heusse, F. Rousseau, G. Berger-Sabbatel, and A. Duda, "Performance anomaly of 802.11b," in *INFOCOM 2003. Twenty-Second Annual Joint Conference of the IEEE Computer and Communications. IEEE Societies*, vol. 2, march-3 april 2003, pp. 836 – 843 vol.2.

[3] S. Zou, B. Li, H. Wu, Q. Zhang, W. Zhu, and S. Cheng, "A relay-aided media access (rama) protocol in multirate wireless networks," *Vehicular Technology, IEEE Transactions on*, vol. 55, no. 5, pp. 1657 –1667, sept. 2006.

[4] K.-W. Chin and S. Li, "Novel association control strategies for multicasting in relay-enabled wlans," *Comput. Netw.*, vol. 56, no. 8, pp. 2168–2178, May 2012.

[5] P. Bahl, R. Chandra, P. Lee, V. Misra, J. Padhye, D. Rubenstein, and Y. Yu, "Opportunistic use of client repeaters to improve performance of wlans," *Networking, IEEE/ACM Transactions on*, vol. 17, no. 4, pp. 1160 –1171, aug. 2009.

[6] M.-H. Lu, P. Steenkiste, and T. Chen, "Design, implementation and evaluation of an efficient opportunistic retransmission protocol," in *Proceedings of the 15th annual international conference on Mobile computing and networking*, ser. MobiCom '09. New York, NY, USA: ACM, 2009, pp. 73–84.

[7] P. Liu, Z. Tao, S. Narayanan, T. Korakis, and S. S. Panwar, "Coopmac: A cooperative mac for wireless lans," *Selected Areas in Communications, IEEE Journal on*, vol. 25, no. 2, pp. 340 –354, february 2007.

[8] L. Guo, X. Ding, H. Wang, Q. Li, S. Chen, and X. Zhang, "Cooperative relay service in a wireless lan," *Selected Areas in Communications, IEEE Journal on*, vol. 25, no. 2, pp. 355 –368, february 2007.

[9] Wi-Fi Alliance, "Wi-fi peer-to-peer (p2p) technical specification v1.0," *Wi-Fi Alliance*, Sep. 2009.

[10] Y. Chen, S. Zhang, S. Xu, and G. Li, "Fundamental trade-offs on green wireless networks," *Communications Magazine, IEEE*, vol. 49, no. 6, pp. 30–37, 2011.

[11] M. Li, Z. Li, and A. Vasilakos, "A survey on topology control in wireless sensor networks: Taxonomy, comparative study, and open issues," *Proceedings of the IEEE*, vol. 101, no. 12, pp. 2538–2557, Dec 2013.

[12] P. Serrano, A. Banchs, P. Patras, and A. Azcorra, "Optimal Configuration of 802.11e EDCA for Real-Time and Data Traffic," *IEEE Transactions on Vehicular Technology*, vol. 59, no. 5, pp. 2511–2528, Jun. 2010.

[13] G. Bianchi, "Performance analysis of the ieee 802.11 distributed coordination function," *IEEE JSAC*, vol. 18, no. 3, pp. 535–547, Mar 2000.

[14] A. Garcia-Saavedra, P. Serrano, A. Banchs, and M. Hollick, "Balancing energy efficiency and throughput fairness in ieee 802.11 wlans," *Pervasive and Mobile Computing*, no. 0, pp. –, 2012.

[15] S. Chiaravalloti, F. Idzikowski, and L. Budzisz, "Power consumption of WLAN network elements," Tech. Univ. Berlin, TKN Tech. Rep. TKN-11-002, aug 2011.

[16] S. Shakkottai and R. Srikant, "Network optimization and control," *Found. Trends Netw.*, vol. 2, no. 3, pp. 271–379, 2007.

[17] J. Kim, S. Kim, S. Choi, and D. Qiao, "CARA: Collision-aware rate adaptation for IEEE 802.11 WLANs," in *INFOCOM 2006. 25th IEEE International Conference on Computer Communications. Proceedings*, 2006, pp. 1–11.

[18] P. Salvador, L. Cominardi, F. Gringoli, and P. Serrano, "A first implementation and evaluation of the ieee 802.11aa group addressed transmission service," *SIGCOMM Comput. Commun. Rev.*, vol. 44, no. 1, pp. 35–41, Dec. 2013.

[19] A. Raniwala, K. Gopalan, and T.-c. Chiueh, "Centralized channel assignment and routing algorithms for multi-channel wireless mesh networks," *SIGMOBILE Mob. Comput. Commun. Rev.*, vol. 8, no. 2, pp. 50–65, 2004.

[20] A. Raniwala and T. Chiueh, "Architecture and algorithms for an IEEE 802.11-based multi-channel wireless mesh network," in *Proceedings IEEE INFOCOM 2005. 24th Annual Joint Conference of the IEEE Computer and Communications Societies*, vol. 3, 2005.

[21] J. Andersen, T. Rappaport, and S. Yoshida, "Propagation measurements and models for wireless communications channels," *Communications Magazine, IEEE*, vol. 33, no. 1, pp. 42 –49, jan 1995.

[22] D. Giustiniano, E. Goma, A. Lopez Toledo, I. Dangerfield, J. Morillo, and P. Rodriguez, "Fair wlan backhaul aggregation," in *Proceedings of the sixteenth annual international conference on Mobile computing and networking*, ser. MobiCom '10. New York, NY, USA: ACM, 2010, pp. 269–280.

Appendix A
Description of the Greedy Algorithm

The following describes the pseudo-code for the two algorithms presented in Section IV-B. Let us first define a set of variables:

- $N :=$ Array containing the set of all nodes.
- $S :=$ Array containing the set of SOLOR nodes and the AP.
- $A :=$ Array containing, for each node's index, its parent.
- $U := |N|x|N|$ matrix with the utility if any node $n \in N$ uses any other as a parent (pairs of legacy nodes has a null value).
- $MCS := |N|x|N|$ matrix with the modulation and coding schema that can be used in all links in the network. Note that each SOLOR node collects this information online according to the measured SNR of each node towards it and shares it with other SOLOR nodes.

Algorithm 1 Greedy Algorithm

1: **function** GREEDY_ALG
2: $\quad A = SortParents(S, N)$ $\quad\quad\quad\quad$ ▷ Initialization
3: $\quad [CurrentUtility, F_{\mathcal{V}}^S] = optimize(S, N, A)$
4: $\quad$ **while** (1) **do**
5: $\quad\quad$ **for** $s \in S$ **do**
6: $\quad\quad\quad$ **for** $n \in N$ **do**
7: $\quad\quad\quad\quad A\_tmp = A$
8: $\quad\quad\quad\quad A\_tmp(n) \leftarrow s$
9: $\quad\quad\quad\quad [U(n, s), F_{\mathcal{V}}^S] = optimize(S, N, A\_tmp)$
10: $\quad\quad\quad$ **end for**
11: $\quad\quad$ **end for**
12: $\quad\quad [BestUtility, n, s] = max(U)$
13: $\quad\quad$ **if** $BestUtility > CurrentUtility$ **then**
14: $\quad\quad\quad A(n) \leftarrow s$
15: $\quad\quad\quad CurrentUtility \leftarrow BestUtility$
16: $\quad\quad$ **else**
17: $\quad\quad\quad Break$
18: $\quad\quad$ **end if**
19: $\quad$ **end while**
20: **end function**
21:
22: **function** SORTPARENTS(S,N)
23: $\quad A(n) \leftarrow zeros(1, |N|)$
24: $\quad$ **for** $n \in N$ **do**
25: $\quad\quad BestMCS \leftarrow MCS(n, A(n))$
26: $\quad\quad$ **for** $s \in S$ **do**
27: $\quad\quad\quad$ **if** $n \neq s$ & $A(s)! = n$ **then**
28: $\quad\quad\quad\quad$ **if** n is not in A **then** ▷ Forbids multi-relay
29: $\quad\quad\quad\quad\quad$ **if** $getMCS(n, s) > BestMCS$ **then**
30: $\quad\quad\quad\quad\quad\quad BestMCS = MCS(n, s)$
31: $\quad\quad\quad\quad\quad\quad A(n) = s$
32: $\quad\quad\quad\quad\quad$ **end if**
33: $\quad\quad\quad\quad$ **end if**
34: $\quad\quad\quad$ **end if**
35: $\quad\quad$ **end for**
36: $\quad$ **end for**
37: **end function**

Andres Garcia-Saavedra (andres.garcia received his B.Sc. degree in Telecommunications Engineering from University of Cantabria in 2009, and his M.Sc. degree in Telematics Engineering from University Carlos III of Madrid (UC3M) in 2010. He currently holds the position of Teaching Assistant and pursues his Ph.D. in the Telematics Department of UC3M. His work focuses on performance evaluation and energy efficiency of wireless networks.

Daniel Camps-Mur is currently leading the Ubiquitous Internet group at I2CAT in Barcelona. Previously, he was a senior researcher at NEC Network Laboratories in Heidelberg, Germany. He holds since 2004 a Masters degree and since 2012 a Ph.D. from the Polytechnic University of Catalonia (UPC). His research interests include mobile networks and IoT.

Balaji Rengarajan is currently an algorithms architect with Accelera Mobile Broadband, CA, USA. Previously, he was a staff researcher at IMDEA Networks, Madrid, Spain. He received his Ph.D. and M.S. in electrical engineering from the University of Texas at Austin, USA in 2009 and 2004 respectively, and his B.E. in Electronics and Communication from the University of Madras in 2002. He was the recipient of a 2003 Texas Telecommunications Engineering Consortium (TxTEC) graduate fellowship and a 2010 Marie-Curie Amarout Europe Programme fellowship. He is also the recipient of the best paper award at the 23rd International Teletraffic Congress (ITC), 2011. His main research interests lie in the analysis and design of wireless and wireline telecommunication networks.

Xavier Costa-Pérez is Head of Wireless & Backhaul Networks R&D at NEC Laboratories Europe, where he has managed several projects related to mobile networks. In the wireless LAN area, he led a team contributing to NEC's mobile phones evolution, which received an R&D Award for the work on N900iL, NEC's first 3G/WiFi phone. In the 4G area he managed a team researching on base station enhancements which received NEC's R&D Award for successful technology transfers. In 3GPP he has contributed to the SA1 RAN Sharing Enhancements efforts where new requirements for future systems are being defined. In IEEE he contributed to the development of the 802.11e and 802.11v standards being included in the major contributor list. He has served on the Program Committees of several conferences (e.g. IEEE Greencom, WCNC, and INFOCOM), published over 50 papers and holds over 20 patents. He received both his M.Sc. and Ph.D. degrees in Telecommunications from the Polytechnic University of Catalonia (UPC) and was the recipient of the national award for the best Ph.D. thesis on "Multimedia Convergence in Telecommunications".

Pablo Serrano (M'09) has been with the Telematics Department of UC3M since 2002, where he currently holds the position of Associate Professor. He was a visiting researcher at the Computer Network Research Group at Univ. of Massachusetts Amherst in 2007, and at Telefonica Research Center in Barcelona in 2013. He has over 50 scientific papers in peer-reviewed international journal and conferences. He serves on the Editorial Board of IEEE Communications Letters, has been a guest editor for Computer Networks, and has served on the TPC of a number of conferences and workshops including IEEE INFOCOM, IEEE WoWMoM and IEEE Globecom.