Robustness and Optimality in CSMA Wireless Networks

Bruno Nardelli and Edward W. Knightly
ECE Department, Rice University, Houston, TX
Technical Report TREE1301

***Abstract*—It has recently been shown that distributed queue-based adaptation of CSMA's contention aggressiveness can provably optimize network utility. However, such an approach is fragile, and was shown to suffer high performance degradation under conditions of frequent occurrence, namely; asymmetric channels, heterogeneous traffic, and packet collisions. In this work, we address the main sources of performance degradation in optimal CSMA to design a distributed system for proportional-fair throughput that delivers high performance in a wide-range of network conditions. First, we propose a generalized version of prior optimal CSMA models to incorporate individual per-link modulation and coding rates. With such a model, we derive adaptive principles that maximize utility under arbitrary channel capacities. Second, we propose a novel structure that can be used in the place of queues to provide optimal CSMA adaptation. As such a structure does not use traffic backlog to operate, the resulting adaptation is optimal for the set of backlogged flows under general traffic arrival patterns. Third, we propose a robustness function that reduces access in high contention scenarios to avoid high performance degradation due to collisions. By evaluating these three solutions against prior designs, we observe vast performance gains, of up to 68% higher logarithmic utility by our approach, on average over randomly-generated scenarios that incorporate the three main sources of performance degradation.**

I. Introduction

Recently, an analytical framework has been proposed to derive distributed CSMA algorithms for network utility maximization [5]. The main idea in such designs is to adapt the back-off distribution at each transmitter based on the length of queues at the MAC layer in a way that was rigorously shown to approach the optimal network-wide throughput distribution. Based on this method, an umbrella of distributed protocols have been derived, showing high performance gains in scenarios considered by the model [5]–[10].

However, later experimental work has shown that the same approach is fragile, and can suffer from high performance degradation as the model assumptions break [13]. In particular, are three the main sources of performance degradation in optimal CSMA networks: channels asymmetries, packet collisions at flow receivers, and dynamic traffic patterns such as congestion-controlled flows. While other real-world conditions have been identified that differ from those assumed by the models (see [12] for a longer list), their impact has been found to be minor in comparison with these three performance degradation sources.

In this work, we derive a novel CSMA system for proportional fairness using a mixed approach that jointly considers *optimization* and *robustness*. On the one hand, we derive techniques to relax the assumptions on channel symmetry and traffic arrival patterns from design models, so that the optimization becomes inherently robust to such conditions. On the other hand, we introduce a robustness function that limits performance degradation due to collisions by reducing network access when contention levels are high. By accounting for the three underlying sources of performance degradation in optimal CSMA, the derived system outperforms current approaches in a wide range of network operating conditions. Our contributions are as follows.

First, we propose a generalized version of the throughput model in [5] to the case of networks with arbitrary link capacities; i.e. to incorporate adaptive modulation and coding rates. While our model is based on a simple extension, it is powerful enough to extend CSMA optimality analysis from the specific case with fixed unitary capacity to the case with general capacity assignments, and to derive adaptation principles robust to such conditions. Furthermore, with this model we show how to derive a distributed CSMA protocol that maximizes proportional-fair throughput in networks with channel asymmetries *without* explicit knowledge of channel error probabilities.

Second, we observe that prior queue-based CSMA guarantees optimal adaptation only when the arrival of packets at each queue follows a specific process derived from the target utility function. For other arrival patterns, the performance of the system remains unspecified, which leads to severe performance degradation under common traffic such as bursty flows and TCP traffic. To solve this, we propose a novel structure, termed the *service meter*, which emulates the operation of a queue, thus inheriting the basic properties that allow optimization, but uses a fictitious flow of abstract transmission units, so that its evolution over time is not affected by (real) traffic arrivals. With such a structure, adaptation can be shown optimal for the set of backlogged flows, with no assumptions on their traffic arrival patterns.

Third, the prior adaptation principle of optimal CSMA models assumes no packet collisions, and can yield severe performance degradation when the network contention levels are high. We show that the goal of optimizing performance alone conflicts with the goal of robustness, such that optimal access can only be attained by arbitrarily increasing the contention rate at all flows, while collisions can only be reduced by decreasing it. Based on this observation, we propose a combined system that balances optimality and robustness by targeting near-optimal access in scenarios with low contention, but reducing channel access to avoid interference as the number of contending flows increases.

Finally, we evaluate the performance of our design against other solutions under the three sources of performance degradation above. Our results show that in scenarios with channel asymmetries our generalized throughput model increases optimization accuracy up to 4 times. In scenarios with heterogeneous traffic, the use of service meters delivers vast fairness gains, restoring the throughput of (otherwise) starving flows. In scenarios with high contention, our approach increases the network throughput with respect to optimal CSMA of about 78% by limiting collisions at flow receivers. Furthermore, the joint operation of these three solutions delivers high performance across a wide-range of network operating conditions, with up to 68% average increase in logarithmic utility in randomly-generated networks with 48 flows.

This document is organized as follows. Section II introduces the optimal CSMA theory, the analytical framework supporting distributed CSMA optimization. Section III derives a distributed protocol for proportional fairness robust to channel asymmetries, heterogeneous traffic and high contention. Sections IV and V respectively evaluate the performance of the derived protocol in isolated scenarios aimed to validate specific system aspects, and in general networks to study the performance of the system as a whole. Section VI discusses related work, and finally, Section VII concludes.

II. The Optimal CSMA framework

A. Network model

Optimal CSMA is an analytical framework proposed in [5] for the optimization of multi-hop wireless networks. Such a framework has been applied in a large number of works for the design of distributed CSMA algorithms maximizing different measures of network performance [5]–[10].

Such works model a wireless network using a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ where $\mathcal{V}$ is the set of nodes and $\mathcal{E} \subseteq \mathcal{V}^2$ is the set of links. Denote as $\mathcal{F} \subseteq \mathcal{E}$ the set of traffic flows, of size $F = |\mathcal{F}|$. It is assumed that a distinct packet queue Q_f is maintained for each flow $f = (i, j)$ at the MAC layer of node i (the *transmitter*, or *source node*). The purpose of such a queue is to temporarily store packets until a transmission opportunity is granted to f.

Depending on the work, interference among links is captured using a *conflict graph* (e.g., [5]), or an *interference matrix* (e.g., [8]). In either case, define an *Independent Set (IS)* in $\mathcal{G}$ as a subset of flows that do not interfere with each other, and thus can successfully transmit simultaneously. An IS is represented by a tuple $m \in \{0, 1\}^F$, where $m_f = 1$ if f belongs to the independent set. Denote as $\mathcal{N}$ the set of all ISs in $\mathcal{G}$.

Assuming unitary modulation rates at all links, and no channel errors, the capacity area of the network is defined as

$$\Gamma = \{\boldsymbol{\gamma} \in [0,1]^F : \exists \boldsymbol{\pi} \in [0,1]^{|\mathcal{N}|}, \\ \forall f \in \mathcal{F}, \gamma_f \leq \sum_{m \in \mathcal{N}} \pi_m \times m_f, \sum_{m \in \mathcal{N}} \pi_m = 1\}$$

i.e., the set of all throughput distributions in the network that are feasible by activating non-interfering links. Note that the area Γ so defined is a convex region, as the convex combination of ISs.

B. Queue-based CSMA optimization

The optimal CSMA framework captures complex interactions among nodes in a multi-hop network using the *continuous-time CSMA model* from [1]. In such a model, the transmitter of a flow f waits for a silent back-off time exponentially distributed with mean $1/\lambda_f$ before transmitting, and uses a transmission duration exponentially distributed with mean μ_f. In the following, we denote such a model as $CSMA(\boldsymbol{\lambda}, \boldsymbol{\mu})$.

The dynamics of a CSMA protocol operating in the network $\mathcal{G}$ can then be captured with a reversible *Continuous Time Markov Chain (CTMC)* $\mathcal{M}$, where the set of states is given by $\mathcal{N}$, and the transmission probabilities depend on the CSMA parameters $(\boldsymbol{\lambda}, \boldsymbol{\mu}) \in {\mathbb{R}_{>0}}^{2\times F}$. Defining $q_f = \log(\lambda_f \times \mu_f) \;\; \forall f \in \mathcal{F}$, the stationary distribution of $\mathcal{M}$ is given by

$$\pi_m^{\boldsymbol{q}} = \frac{\exp\left(\sum_{f\in\mathcal{F}} q_f \times m_f\right)}{\sum_{n\in\mathcal{N}} \exp\left(\sum_{f\in\mathcal{F}} q_f \times n_f\right)} \quad \forall m \in \mathcal{N} \tag{1}$$

Thus, assuming fixed unitary link capacities, and that simultaneous transmissions over interfering links are always avoided by *Carrier Sensing (CS)* [2], the flow throughput distribution can be determined as

$$\gamma_f^{\boldsymbol{q}} = \sum_{m\in\mathcal{N}} \pi_m^{\boldsymbol{q}} \times m_f \qquad \forall f \in \mathcal{F}$$

The work in [5] shows that, for any $\boldsymbol{\gamma}$ in the interior of Γ, there exists $(\boldsymbol{\lambda}, \boldsymbol{\mu}) \in {\mathbb{R}_{>0}}^{2\times F}$ such that

$$\gamma_f \leq \gamma_f^{\boldsymbol{q}} \qquad \forall f \in \mathcal{F}$$

Furthermore, it shows that, if the input rate at all MAC-layer queues is within the CSMA capacity region, all such queues are stabilized by adapting the value of $\boldsymbol{q}$ over time as a scaled version of the queue lengths.

More precisely, time is divided into small intervals indexed by $t \in \mathbb{N}$. Denote as $Q_f[t]$ the queue length of flow f at the beginning of interval t. Also, denote as $\lambda_f[t]$, and $\mu_f[t]$ respectively the medium access rate, and the transmission duration used by flow f during interval t. Defining $q_f[t] = b \times Q_f[t] \;\; \forall f \in \mathcal{F}$ (with b a small positive real value), all MAC-layer queues are stabilized by adapting the CSMA parameters so that

$$\lambda_f[t] \times \mu_f[t] = \exp(q_f[t]) \;\; \forall f \in \mathcal{F} \tag{2}$$

for each interval t. In other words, by periodically adapting the CSMA parameters using rule (2), any throughput distribution $\boldsymbol{\gamma}$ in the interior of Γ is supported.[1]

Finally, [5]–[8] use the adaptation rule (2) to navigate the convex region Γ and derive subgradient methods to approximately solve different optimization problems. For example,

[1]In the case of shared transmitters, local contention within a node is resolved deterministically by serving the flow f with larger $\lambda_f[t]$ at each interval t.

denoting as $S_f[t]$ the throughput received by a flow f during interval t, the proportional-fair throughput maximization problem

$$\max_{\gamma\in\Gamma}\{\sum_{f\in\mathcal{F}}\log(\gamma_f)\} \tag{3}$$

can be solved by injecting $V/q_f[t]$ data into each flow queue Q_f during each interval t (where V is a positive real number), so that the variation of queue length during interval t is given by $\triangle Q_f[t] = ((V/q_f[t]) - S_f[t])$. The idea is that $\triangle Q_f[t]$ captures a subgradient step of the logarithmic function over the area Γ to approach the maximum point (3).

This last step does require some assumptions; *(i)* all queues are assumed non-empty throughout the system execution for the objective in (3) to be fixed over time; *(ii)* timescale separation is needed for the network to converge to its steady-state within one time interval, so that $S_f[t]$ well-approximates the value $\gamma_f^{\boldsymbol{q}[t]}$; and *(iii)* at least $V/q_f[t]$ data from upper layers should be available to be injected into Q_f during each interval t.

While the assumption on non-empty queues has limited impact (since flows without data to transmit do not need to adapt in any case), and the assumption on timescale separation can be relaxed using the ideas on [8], the assumption of $V/q_f[t]$ arrivals is hard to relax in such a design in which the medium access rate of flows is exclusively adapted as a function of queue backlog. Nevertheless, when the assumptions of the model hold, optimal CSMA guarantees near-optimal performance, solving optimization problems such as (3) as an approximation algorithm with arbitrarily bounded accuracy.

III. ROBUST CSMA WITH NETWORK OPTIMIZATION

A. Design overview

The powerful analytical framework of optimal CSMA theory allows the derivation of distributed algorithms with probable performance guarantees. However, as discussed in Section II, a number of assumptions are required to prove optimality. Furthermore, later experimental works have shown that the impact of some of those assumptions can be high, significantly degrading the protocol performance in real scenarios.

In this section, we design a distributed CSMA protocol for proportional fairness, that addresses the main sources of performance degradation in optimal CSMA to deliver high performance across a wide range of networking scenarios. In particular, we overcome the limitations to provide robust operation in the presence of three challenging conditions; channel asymmetries, heterogeneous traffic patterns and high contention.

First, in Section III-B, we introduce a generalized version of the model in [5], that accounts for traffic asymmetries to capture the relation between flow throughput and transmission time. With the use of such a model, we show how to relax the assumption on fixed unitary link capacities from optimal CSMA models by splitting the derivation into two steps: *(i)* In Section III-C we derive a CSMA protocol that maximizes proportional-fair transmission time without any assumptions on the symmetry of channels; *(ii)* Using a problem-reduction technique, we show in Section III-D that the same protocol also maximizes proportional-fair throughput in the same scenarios. Furthermore, the protocol operates in a completely distributed way, and does not require explicit knowledge of channel error rates.

Second, we address performance degradation due to heterogeneous traffic jointly with the optimization of transmission time in Section III-C. Solving this problem is hard in optimal CSMA, since the analytical expressions of subgradient steps used to maximize a given objective function only apply to MAC-layer queues under specific traffic arrival rates. In contrast, we show that the subgradient of network utility can also be captured by the use of a novel structure, termed the *service meter*, whose evolution over time is *only* affected by the service received by a flow. Thus, any backlogged flow can receive optimal adaptation with such a structure regardless of the traffic arrival rates from upper layers.

Third, a simplifying assumption in optimal CSMA models is that CS *always* prevents any simultaneous transmissions at interfering links. While the impact of such an assumption may be limited in small networks, in Section III-E we show that it leads to high performance degradation with a large number of contending flows. Furthermore, we show that the goal itself of optimizing performance as captured by such a model is conflicting with the goal of robustness to interference, such that nominal performance can only be maximized by incurring in high collisions, and vice versa, robustness to high contention can only be attained by reducing medium access. Thus, we propose a mixed design, that adapts over time to deliver near-optimal performance when the network contention levels are low, yet reduces access to avoid interference in scenarios with high contention.

B. A generalized throughput model

Here, we introduce a generalized version of the throughput model in Section II-A that explicitly captures the relation between throughput and transmission time for each traffic flow over links with arbitrary channel capacities. Albeit based on a simple extension, the model extends optimization analysis from fixed unitary capacities to the general case with (potentially) asymmetric channels, and will be used in the next sections to derive a distributed optimal CSMA protocol robust to such conditions.

To this end, we model a wireless network using a *labeled* graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \boldsymbol{c})$ where $\mathcal{V}$ is the set of nodes, $\mathcal{E} \subseteq \mathcal{V}^2$ is the set of links, and the labels in $\boldsymbol{c} \in \mathbb{R}_{>0}^{|\mathcal{E}|}$ are the capacities of each link in $\mathcal{E}$. As before, denote as $\mathcal{F} \subseteq \mathcal{E}$ the set of traffic flows, with $F = |\mathcal{F}|$, and as $\mathcal{N}$ the set of ISs in $\mathcal{G}$.

Such a formulation is flexible enough to accommodate different notions of channel capacity. In general, we assume c_f to be the average transmission rate attained by a flow f in isolation under maximum channel utilization. For example, denoting as r_f the modulation rate used by the transmitter of flow f, and as e_f the error probability over the channel used by flow f, the channel capacity of f's link is given by $c_f = r_f \times (1 - e_f)$.

Under this network model, the set of all feasible transmission-time distributions among flows in $\mathcal{F}$ (constrained over non-interfering links) is given by

$$\Psi = \{\boldsymbol{\psi} \in [0,1]^F : \exists \boldsymbol{\pi} \in [0,1]^{|\mathcal{N}|}, \\ \forall f \in \mathcal{F}, \psi_f \leq \sum_{m \in \mathcal{N}} \pi_m \times m_f, \sum_{m \in \mathcal{N}} \pi_m = 1\}$$

which is convex, as the convex combination of ISs in $\mathcal{N}$.

Given a transmission-time distribution $\boldsymbol{\psi} \in \Psi$, the throughput of a flow f under the channel capacities $\boldsymbol{c}$ is given by $\gamma_f(\boldsymbol{\psi}, \boldsymbol{c}) = \psi_f \times c_f$. Furthermore, the capacity area of $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \boldsymbol{c})$ is defined as

$$\Gamma(\boldsymbol{c}) = \{\boldsymbol{\gamma} \in \mathbb{R}^F_{\geq 0} : \exists \boldsymbol{\psi} \in \Psi, \boldsymbol{\gamma} = \boldsymbol{\gamma}(\boldsymbol{\psi}, \boldsymbol{c})\}$$

which again, is convex as a scaled version of the convex set Ψ. The model in Section II-A can be interpreted as a special case of the model here presented, when $c_f = 1, \ \forall f \in \mathcal{F}$. However, it is direct to see that the throughput of a flow f as captured by the two models can significantly differ depending on the value of c_f.

Finally, we redefine the goal of maximizing a proportional-fair throughput distribution in $\mathcal{G}$ as

$$\max_{\boldsymbol{\psi} \in \Psi} \{\sum_{f \in \mathcal{F}} \log(\gamma_f(\boldsymbol{\psi}, \boldsymbol{c}))\} \tag{4}$$

i.e., adapting the transmission-time distribution to the one that maximizes the network-wide logarithmic utility of throughput. In the following sections, we derive a distributed CSMA algorithm that approximately solves (4) with no explicit knowledge of the values in $\boldsymbol{c}$.

C. Transmission-time optimization under heterogeneous traffic

1) Transmission-time CSMA optimality: Consider the network model in Section III-B. In this section, we derive a distributed CSMA algorithm that solves

$$\max_{\boldsymbol{\psi} \in \Psi} \{\sum_{f \in \mathcal{F}} \log(\psi_f)\} \tag{5}$$

robust to channel asymmetries and heterogeneous traffic arrival patterns. Later, in Section III-D, we will show that the same algorithm also solves (4), with arbitrarily bounded accuracy.

In the definition of (5), as well as in the following analysis, we assume that all flows in $\mathcal{F}$ always have a packet to transmit. This simplifies analysis by considering a fixed objective over time well-defined over the set $\mathcal{F}$, so that an algorithm converging to the optimal point $\boldsymbol{\psi}^*$ can be derived. While in practice queues can become empty, this does not limit the applicability of our method, as we can assume the set $\mathcal{F}$ to dynamically adapt in time to include *only* the set of backlogged flows (which automatically changes the goal defined by (5) as $\mathcal{F}$ changes). Furthermore, as long as queues are non-empty, we make no assumption on the packet arrival process from upper layers which is the fundamental aspect to attain robust CSMA adaptation.

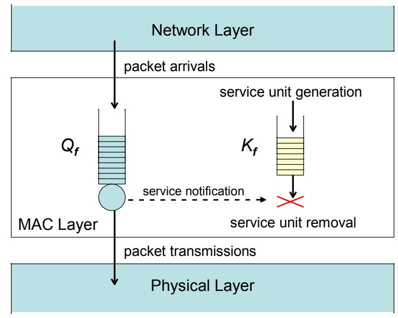


Fig. 1: A combined packet-queue Q_f and service meter K_f, for a given traffic flow f. Solid arrows represent the flow of packets/units, whereas dotted arrows represent the interaction among different system components. In the case of shared transmitters, a different pair (Q_i, K_i) is maintained at the node for each local flow i. The node transmitter notifies the service meter K_f upon completion of an $f's$ transmission, to subtract the amount of service received.

We model the operation of a CSMA protocol over the network $\mathcal{G}$ with the above described *continuous-time CSMA model* $CSMA(\boldsymbol{\lambda}, \boldsymbol{\mu})$. Then, the steady-state distribution of transmission time in the network $\mathcal{G}$ is given by

$$\psi_f^{\boldsymbol{k}} = \sum_{m \in \mathcal{N}} \pi_m^{\boldsymbol{k}} \times m_f \qquad \forall f \in \mathcal{F} \tag{6}$$

where $\boldsymbol{k}$ is defined as $k_f = \log(\lambda_f \times \mu_f) \ \ \forall f \in \mathcal{F}$ and the value of $\pi_m^{\boldsymbol{k}}$ is given by (1). Also, for any transmission-time distribution $\boldsymbol{\psi}$ in the interior of Ψ, there exists $\boldsymbol{k} \in \mathbb{R}^F$ such that

$$\psi_f \leq \psi_f^{\boldsymbol{k}} \qquad \forall f \in \mathcal{F}$$

i.e., as in the case of throughput with fixed unitary link capacities, any transmission-time distribution $\boldsymbol{\psi}$ in the interior of Ψ can be attained by selecting an appropriate choice of CSMA parameters $(\boldsymbol{\lambda}, \boldsymbol{\mu}) \in {\mathbb{R}_{>0}}^{2 \times F}$. Furthermore, due to the convexity of Ψ, it is possible to derive subgradient methods to solve optimization problems such as (5) by adapting the value of $\boldsymbol{k}$ to navigate the region Ψ.

2) A novel structure for subgradient methods: The challenge to solve (5) is to derive an expression of the logarithm's subgradient that can be translated into distributed operations to adapt the CSMA parameters at all network nodes. To this end, the solutions described in Section II use the length of MAC-layer queues as a measure of the service received by each flow, in order to adapt its parameters accordingly.

Unfortunately, the use of queues to regulate CSMA access does not apply well to the problem of optimizing flow transmission time with heterogeneous traffic considered here. First, packets may not be removed from a queue in case of unsuccessful transmission. Thus, queue length variations naturally reflect the amount of throughput received by a flow, but are not suitable to measure transmission time. Second, it requires the assumption that, at any time interval, the packet arrival rates from upper layers are high-enough to maintain the required queue length for optimal adaptation.

Our key contribution is the design of a novel structure that captures the subgradient of the utility function in order to solve (5), without the use of traffic backlog. Instead, it acts like a counter that records the amount of service received by a flow f, using abstract *service units*. We refer to such a device, depicted in Fig. 1, as the *service meter*, denoted as K_f. Since the service meter does not use any packets to provide adaptation, it can be defined to measure service in terms of transmission time, and its operation is not dependent on the packet arrival rate from upper layers.

More precisely, each node periodically updates its state over small time intervals indexed by $t \in \mathbb{N}$. Denote as $K_f[t]$ the value of K_f at the beginning of interval t. To provide for CSMA adaptation, denote as $\lambda_f[t]$ the medium access rate used by f's transmitter during interval t. Also, denote as $T_f[t]$ the fraction of transmission-time by a flow f during interval t. During each interval t, a number of service units are added to the service meter K_f. Upon a data packet transmission from flow f, the corresponding transmission time is subtracted from K_f (even in the case of unsuccessful transmission), such that the "service" received by the service meter during interval t is equal to $T_f[t]$.

Then, the set of service meters can be stabilized by defining $k_f[t] = b \times K_f[t] \ \ \forall f \in \mathcal{F}$ (with a small positive value $b \in \mathbb{R}_{>0}$), and adapting each flow f's channel access rate at the end of each interval t as

$$\lambda_f[t+1] = \frac{\exp(k_f[t+1])}{\mu_f} \tag{7}$$

In other words, any target transmission-time distribution $\boldsymbol{\psi} \in \Psi$ can be attained using (7) while incrementing each K_f at a rate ψ_f.[2]

Furthermore, by limiting the values of $k_f[t]$ within a range $[k_{min}, k_{max}] \subset \mathbb{R}_{>0}$, and adding $V/k_f[t]$ units to K_f during each interval t (with $V \in \mathbb{R}_{>0}$), the evolution of $k_f[\cdot]$ is determined by

$$k_f[t+1] = \left[k_f[t] + b \times \left(\frac{V}{k_f[t]} - T_f[t]\right)\right]_{k_{min}}^{k_{max}} \tag{8}$$

where $[\cdot]_{k_{min}}^{k_{max}} = min(max(\cdot, k_{min}), k_{max})$. Then, $\triangle K_f[t] = ((V/k_f[t]) - T_f[t])$ can be readily interpreted as a subgradient step to approximately solve (5), when using (8) together with (7) to provide for CSMA adaptation, as shown in the following result.

[2]Here, we use the assumption that each flow $f \in \mathcal{F}$ always has at least one packet to transmit. In a practical implementation, if a queue Q_f becomes empty (so that the flow f does not need to be served), remove the corresponding service meter K_f from the system and remove f from $\mathcal{F}$ so that the assumption still holds. Similarly, add a new service meter to the system when a new traffic flow starts.

3) A distributed CSMA algorithm for proportional-fair transmission-time maximization:

Proposition 1. *Refer as Algorithm 1 to a protocol using rules (7) and (8) at all nodes to update their CSMA parameters. Then, Algorithm 1 approximately solves (5) with bounded accuracy* $\log(|\mathcal{N}|)/V$.

Proof: Consider the following optimization problem.

$$\max_{\boldsymbol{\psi}, \boldsymbol{\pi}} \{V \sum_{f \in \mathcal{F}} \log(\psi_f) - \sum_{m \in \mathcal{N}} \pi_m \log(\pi_m)\} \tag{9}$$
$$\text{s.t.} \quad \forall f \in \mathcal{F} \quad \psi_f \le \sum_{m \in \mathcal{N}} \pi_m \times m_f, \quad \sum_{m \in \mathcal{N}} \pi_m = 1$$

We proceed by showing that (8) can be interpreted as a subgradient step of a dual problem of (9), projected onto $[k_{min}, k_{max}]$. First, we derive the Karush-Kuhn-Tucker conditions to solve (9) as

$$V/\psi_f = \nu_f, \forall f \in \mathcal{F}, \tag{10}$$
$$-1 - \log \pi_m + \sum_{f \in \mathcal{F}} (\nu_f - \eta) \times m_f = 0, \forall m \in \mathcal{N}, \tag{11}$$
$$\nu_f \times (\psi_f - \sum_{m \in \mathcal{N}} \pi_m \times m_f) = 0, \forall f \in \mathcal{F}, \tag{12}$$
$$\eta \times (\sum_{m \in \mathcal{N}} \pi_m - 1) = 0, \tag{13}$$
$$\nu_f \ge 0, \forall f \in \mathcal{F} \tag{14}$$

where we omit the intermediate step of deriving the Lagrangian $\mathcal{L}(\boldsymbol{\psi}, \boldsymbol{\pi}; \boldsymbol{\nu}, \eta)$ of (9) for brevity.

For each flow f, define a dual variable $\tilde{k}_f = \nu_f$. Using ideas analogous to [5], [8], (11) and (13) can be satisfied by choosing $\eta = \log(\sum_{m \in \mathcal{N}} \exp(\sum_{f \in \mathcal{F}} \tilde{k}_f \times m_f)) - 1$ and $\boldsymbol{\pi} = \boldsymbol{\pi}^{\tilde{\boldsymbol{k}}}$ (which is equivalent to the adaptation rule (7), through the equality (1)). Furthermore, the subgradient of (10) satisfying (12) is given by

$$\dot{\nu}_f = (V/\nu_f - \sum_{m \in \mathcal{N}} \pi_m^{\tilde{\boldsymbol{k}}} \times m_f) \tag{15}$$

Then, (15) is a subgradient step to solve the dual of problem (9). Since (9) is strictly convex, the subgradient method based on (15) is guaranteed to converge to the solution $\boldsymbol{\nu}^* = (\nu_f^*, f \in \mathcal{F})$. Moreover, if $\boldsymbol{\nu}^* \in [k_{min}, k_{max}]^F$, (15) is equivalent to adaptation rule (8), through the use of (6).[3] This shows that Algorithm 1 solves (9).

It remains to show that Algorithm 1 solves (5) with bounded accuracy $\log(|\mathcal{N}|)/V$. To see this, note that (5) is equivalent to

[3]While this assumes that the network converges to a steady state within one interval, such that the measure $\boldsymbol{T}[t]$ attains the value of $\boldsymbol{\psi}^{\boldsymbol{k}[t]}$, the analysis can be readily extended using the ideas in [8] to relax such an assumption.

$$\max_{\boldsymbol{\psi}\in\Psi}\{V\sum_{f\in\mathcal{F}}\log(\psi_f)\} \tag{16}$$

The bound $\log(|\mathcal{N}|)/V$ can be obtained by comparing (16) to (9) and limiting the term $\sum_{m\in\mathcal{N}}\pi_m\log(\pi_m)$ in (9) with the known entropy bound $\log(|\mathcal{N}|)$. ■

Remark 1. While Proposition 1 relies on the assumption that all queues are non-empty throughout the system operation, and that the set of flows $\mathcal{F}$ is fixed over time, in practice the same results can be applied to dynamic scenarios by observing that optimal adaptation in Proposition 1 is attained from *any* starting point. Then, in the case of changes on the set of backlogged flows, which would imply variations on the optimal point defined by (5), the algorithm continues adapting after each change in the search for the (new) optimal point.

Remark 2. The positive term $V/k_f[t]$ in (8) is not dependent on flow packet arrivals from upper layers, and can be added to K_f even if packet sources have been interrupted, to provide continued adaptation at any backlogged flows up to the last packet transmitted. In Section IV-C we show that this is a fundamental aspect to attain high performance in scenarios with heterogeneous traffic.

For ease of reference, we next provide a description of Algorithm 1. There, we use $Q_f[t]$ to denote the length of f's MAC-layer queue at the beginning of interval t.

Algorithm 1 Distributed CSMA adaptation

The following procedures are executed by the transmitter of each flow $f\in\mathcal{F}$.

During interval t:

1: Run $CSMA(\boldsymbol{\lambda}[t],\boldsymbol{\mu})$ while recording the fraction of transmission time $T_f[t]$ received during interval t

At the end of interval t:

1: **if** $Q_f[t+1]>0$ **then**

2: Set $k_f[t+1]=\left[k_f[t]+b\times\left(\frac{V}{k_f[t]}-T_f[t]\right)\right]_{k_{min}}^{k_{max}}$

3: **end if**

4: Update $\lambda_f[t+1]=\exp(k_f[t+1])/\mu_f$

If Q_f becomes empty during interval t:

1: Reset $k_f[t]=k_{min}$

2: Update $\lambda_f[t]=\exp(k_f[t])/\mu_f$

D. Maximizing proportional-fair throughput over asymmetric channels

In Section III-C we have shown that Algorithm 1 maximizes proportional-fair distributions of transmission time in networks with (or without) channel asymmetries. In the following, we extend our analysis to show that the same protocol also maximizes proportional-fair throughput, thus solving (4), in the same scenarios.

We proceed by reducing the problem of maximizing proportional-fair throughput (4) to the problem of maximizing proportional-fair transmission time (5). To this end, refer to the throughput model defined in Section III-B. Then, assuming no packet collisions, which will be treated separately in Section III-E, we have

$$\begin{aligned}\arg\max_{\boldsymbol{\psi}\in\Psi}\{\sum_{f\in\mathcal{F}}\log(\gamma_f(\boldsymbol{\psi},\boldsymbol{c}))\} &=\\ \arg\max_{\boldsymbol{\psi}\in\Psi}\{\sum_{f\in\mathcal{F}}\log(\psi_f)+\log(c_f))\} &=\\ \arg\max_{\boldsymbol{\psi}\in\Psi}\{\sum_{f\in\mathcal{F}}\log(\psi_f)\}\end{aligned} \tag{17}$$

Equation (17) shows that (5) and (4) are equivalent problems. In the next result, we use this property to show that Algorithm 1 maximizes proportional-fair throughput in wireless networks without any assumptions on the symmetry of channels.

Theorem 1. *For any choice of channel capacities $\boldsymbol{c}\in\mathbb{R}_{>0}^{|\mathcal{E}|}$, Algorithm 1 solves (4) as an approximation algorithm with bounded accuracy $\log(|\mathcal{N}|)/V$.*

Proof: Without loss of generality, let a choice of channel capacities $\boldsymbol{c}\in\mathbb{R}_{>0}{}^{|\mathcal{E}|}$ be given. From Proposition 1, Algorithm 1 solves (5) as an approximation algorithm. Hence, since (5) is equivalent to (4) via (17), Algorithm 1 also solves (4).

It remains to show that the accuracy of Algorithm 1 in solving (4) is bounded by $\log(|\mathcal{N}|)/V$. To this end, denote as $\boldsymbol{\gamma}^*$ the solution to (4). Similarly, denote as $\boldsymbol{\psi}^*$ the solution to (5). Also, denote as $\boldsymbol{\gamma}^\dagger$ and $\boldsymbol{\psi}^\dagger$ respectively the throughput and transmission-time distributions attained by Algorithm 1 in the same network after convergence. Then,

$$\begin{aligned}|\sum_{f\in\mathcal{F}}(\log(\gamma_f^*)-\log(\gamma_f^\dagger))| &=\\ |\sum_{f\in\mathcal{F}}(\log(\psi_f^*\times c_f)-\log(\psi_f^\dagger\times c_f))| &=\\ |\sum_{f\in\mathcal{F}}(\log(\psi_f^*)-\log(\psi_f^\dagger))| &\le \log(|\mathcal{N}|)/V\end{aligned}$$

where the inequality at the last step is given by Proposition 1. ■

Remark 3. Equation (17) implies that the optimal point $\boldsymbol{\psi}^*$ does not change for different values of $\boldsymbol{c}$. Thus, once the algorithm has converged to an optimal point, it does not need to re-converge at every change in $\boldsymbol{c}$, as the same operating point maximizes the network performance across different channel conditions. This property allows Algorithm 1 to natively support modulation rate adaptation without any required extensions, and variations in the channel conditions without continuously tracking the error rate probability at each link.

E. Robust operation under high contention

1) Perfect sensing in $CSMA(\boldsymbol{\lambda},\boldsymbol{\mu})$: The analytical derivation in the previous sections is based on the continuous-time $CSMA(\boldsymbol{\lambda},\boldsymbol{\mu})$ model, which captures the service received by each flow with simple analytical expressions that integrate well into an optimization framework. A required assumption in the model is that of a "perfect" CS implementation that always avoids any simultaneous transmissions at interfering links. While in practice different conditions can make CS fail, the $CSMA(\boldsymbol{\lambda}, \boldsymbol{\mu})$ model is still widely adopted due to its suitability for multi-hop wireless CSMA optimization [5]–[8]. Next, we discuss the performance of algorithms derived with this model, in general scenarios with imperfect sensing.

In a real network, CS may fail to detect an ongoing transmission over interfering links due to either of two situations [4]; (i) propagation delays introduce a detection delay such that two neighbor nodes may not detect each other's transmissions if they decide to transmit at nearly the same time, and (ii) attenuated signals may not be strong enough to be detected if two interfering transmitters are relatively far from each other, a situation widely known as *hidden terminals*. In any of the two cases, interference from other links at a flow's receiver can prevent the successful reception of a transmitted packet (which is referred to as a *collision*).

Prior works use the assumption that the effects of hidden terminals can be limited using RTS-CTS handshakes. In the following analysis, we also rely on this assumption which was validated by experimental work in [13] yielding good results. For the case of collisions with neighbor transmitters, instead, prior work suggests that the effects of imperfect sensing are limited by keeping long back-off times [16]. The idea is that, when capturing access based on the $CSMA(\boldsymbol{\lambda}, \boldsymbol{\mu})$ model, back-off times and transmission durations can be jointly expanded while still guaranteeing an optimal adaptation at all flows. For example, in the case of Algorithm 1, (7) shows that choosing a long transmission duration μ_f, the access rate assigned to each flow f is reduced.

2) The optimization-robustness conflict: While the condition of long transmissions is *necessary* in order to reduce access aggressiveness by multiple flows and limit collisions, in the following we show that such a condition is *insufficient* to guarantee robust operation across multiple system configurations. Furthermore, we show that optimizing the system performance as captured by $CSMA(\boldsymbol{\lambda}, \boldsymbol{\mu})$, and guaranteeing robust operation of CS are two conflicting goals, such that nominally-optimal access can only be attained by incurring in high collisions, and viceversa, collisions can only be limited by reducing network access.

To see this, first note that our solution (as well other protocols derived from the same optimization framework), does not attain optimality in absolute terms, but only asymptotically as an approximation algorithm with bounded accuracy $E(V) = \log(|\mathcal{N}|)/V$. Moreover, $E(V)$ is a monotonically decreasing function of V, and the limit $\lim_{V\to\infty} E(V) = 0$ implies that near-optimal performance is attained by the choice of a large V.

Denote as $\boldsymbol{\psi}^V$, $\boldsymbol{k}^V$, and $\boldsymbol{\lambda}^V$, respectively the transmission-time distribution, the service meter values, and the flow access rates attained by Algorithm 1 at convergence under the parameter assignment V. Then, the condition $\lim_{t\to\infty} \triangle k_f[t] \approx 0$ implies

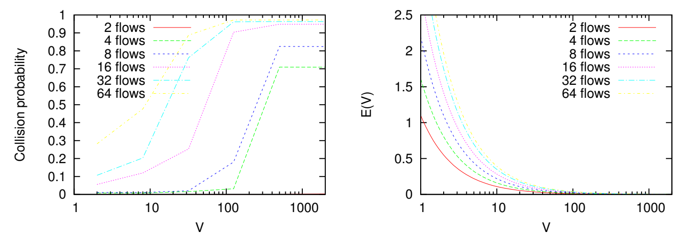
$$((V/k_f^V) - \psi_f^V) \approx 0 \qquad \forall f \in \mathcal{F}$$

where we approximate the transmission-time measure $\lim_{t\to\infty} T_f[t]$ with its expected value ψ_f^V. Furthermore,

$$\lambda_f^V = \exp(k_f^V)/\mu_f \approx \exp(V/\psi_f^V)/\mu_f \quad \forall f \in \mathcal{F} \tag{18}$$

which yields $\lim_{V\to\infty} \lambda_f^V = \infty$. i.e., while optimal performance is attained asymptotically as $V \to \infty$, the target access rate at each flow diverges as $\Theta(\exp(V))$.

In addition, the value of λ_f^V depends on the received service ψ_f^V, which appears as a denominator in (18). Thus, for a fixed value of V, a flow with a lower service attains a higher access rate than other flows. While this is a required feature to provide fairness in asymmetric scenarios where a flow's perceived service is low *relative* to other flows, in a highly congested scenario with symmetric contention, the low service received leads to high access rates at all involved flows, consequently increasing the collision probability.



(a) Collision probability as a function of V and F.

(b) Theoretical bound $E(V)$ for the optimization error, as a function of V and F.

Fig. 2: Performance attained under symmetric contention and different choices of the parameter V.

Fig. 2 shows the collision probability and theoretical error bound under different levels of contention in a symmetric scenario, as a function of V. To obtain the collision probabilities in Fig. 2a, we execute Algorithm 1 in our simulator implementation (more details about the simulator itself are given in Section IV-A). Such relations clearly show a trade-off between nominal performance and robustness to interference, such that reducing the optimization error can only be attained by increasing the collision probability (and vice versa).

3) Balancing robustness with optimization accuracy: While it is not possible to simultaneously attain optimal access and minimize collisions, the relations in Fig. 2 determine, for each level of contention, the maximum value of V such that collisions fall below a given threshold. In this way, the theoretical error in the network optimization is minimized subject to a maximum allowed collision probability. Note that, as CSMA adaptation in Algorithm 1 is only based on transmission time, thus independent from transmission success rates, the measures in Fig. 2a apply to any modulation rate and channel conditions, as long as all flows use a *fixed* average transmission time $\mu^\diamond$ (in our implementation, described in Section IV-A, we attain this by using a combination of packet fragmentation and aggregation at the MAC layer).

Moreover, the curves in Fig. 2 show that the trade-off between nominal performance and robustness highly depends on

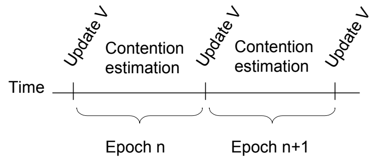


Fig. 3: Alternation of contention estimation and parameter update phases for automatic V adaptation.

the number of contending flows. Thus, the level of contention is an essential metric for the robust configuration of optimal CSMA. As the network topology can change over time, we propose an adaptive system that periodically updates the value of V based on an estimated measure of the network contention level.

The key idea is to exploit the broadcast nature of wireless transmissions to derive an estimation of the network contention level. For example, in a scenario with symmetric contention, each node can estimate the number of contending flows by overhearing packet transmissions from neighbor flows. Then, the relations in Fig. 2 can be used at each node to independently select the configuration of V that yields the desired balance between optimization accuracy and robustness (later on in this section we explain how to ensure a symmetric choice of V in the case of asymmetric scenarios).

More precisely, assume that time is divided into epochs of equal length and indexed by n. During an epoch n, all network flows distributedly estimate a measure Ω_n of the network contention level (measured in number of mutually-contending flows). At the end of epoch n, each node uses Ω_n to select a value V_{n+1} to be used as the configuration V during the next epoch. Fig. 3 shows a diagram of the system operation.

An additional challenge is given by the fact that, for the results in sections III-C and III-D to hold, the value of V should be the same at all network flows. While in a symmetric scenario all flows have the same number of neighbors, in general networks different nodes may experience different contention levels. In such cases, we use the maximum level of contention in the network in order to select V. Denoting as ω_n^i the number of contending flows detected by node i by the start of epoch n, we define the maximum network contention level as $\Omega_n = \max_{i \in \mathcal{V}}\{\omega_n^i\}$.[4] The idea is to satisfy the robustness requirement in the entire network by selecting a value of V that yields low collisions at the point with maximum contention (and thus also in other points).

Furthermore, the value of $\Omega_n = \max_{i \in \mathcal{V}}\{\omega_n^i\}$ can be determined distributedly in a multi-hop network using the lightweight gossiping protocol in Algorithm 2. In our solution, a different instance of such an algorithm is executed at each epoch n. At the end of epoch n, the resulting value of $\hat{\Omega}_n^i$ is used at node i to update its configuration of V. It is easy to show by induction on the set of nodes, that for any connected network, the estimation $\hat{\Omega}_n^i$ at each node i converges to the value of $\max_{j \in \mathcal{V}}\{\omega_n^j\}$ (we omit the proof for brevity).

[4]In such scenarios, we also infer contending flows based on overheard CTS packets, so that ω_n^i includes all contenders to a flow i, either hidden or in transmission range. Our performance evaluation shows that the effects of hidden terminals are significantly reduced by the use of RTS/CTS, and the dominant factor in Optimal CSMA remains the number of contending flows.

Algorithm 2 Maximum contention level estimation

The following procedures are executed by each node $i \in \mathcal{V}$ during an epoch n.

At the start of epoch n:
1: Set $\hat{\Omega}_n^i = \omega_n^i$

Before a packet p is sent by i:
1: Set the field p.epoch = n
2: Set the field p.$\hat{\Omega}$ = $\hat{\Omega}_n^i$

After packet p is received (overheard) by i:
1: **if** p.epoch = n **and** $\hat{\Omega}_n^i <$ p.$\hat{\Omega}$ **then**
2: Set $\hat{\Omega}_n^i$ = p.$\hat{\Omega}$
3: **end if**

At the end of epoch n:
1: Return $\hat{\Omega}_n^i$ as the estimated value of $\max_{j \in \mathcal{V}}\{\omega_n^j\}$.

IV. Evaluation of isolated performance factors

We validate our design by means of extensive simulations. As discussed before, three are the main sources of performance degradation in optimal CSMA networks, each of them affecting the system operation in a different way; channel asymmetries, heterogeneous traffic patterns and collisions under high contention. To precisely evaluate the protocol operation under each of these conditions, in this section we present custom evaluation scenarios that isolate one source of performance degradation at a time, and reveal the performance gains delivered by our design in each case.

In addition, these three performance degradation sources appear combined in real networks, affecting the protocol performance at the same time. Thus, in Section V we extend our evaluation to randomly-generated scenarios that study how the combination of multiple protocol features delivers high performance in general settings.

A. Simulation setup

For our evaluation, we use extensions to Glomosim for optimal CSMA adaptation that have been validated experimentally by prior works [11], [13]. In addition, we implemented custom modules to support independent per-link modulation rate assignment, and extended the BER-table channel model to support multiple rates, as required for the evaluation of scenarios with asymmetric channels in Sections IV-B and V.

To ensure a fixed mean transmission time $\mu^\diamond$ as required by our protocol, we use a combination of packet aggregation and fragmentation at the MAC layer. In our simulations, we set $\mu^\diamond = 2.32$ ms for all flows, which comprises the entire duration of a transmission exchange from the RTS to the last ACK. To implement the epoch-based adaptation of V derived in Section III-E3, we maintain a loose synchronization among nodes by means of data packet transmissions. The total overhead required to implement the solution sums up to 2 bytes per packet.

We evaluate our design alongside 802.11a and an optimal CSMA protocol representative of the ones described in Section

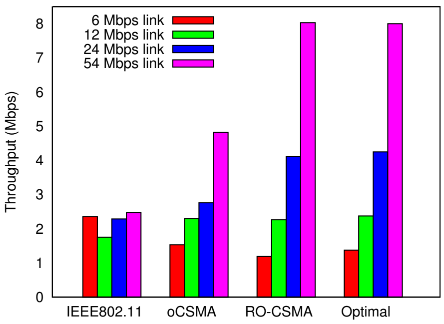


Fig. 4: Throughput distribution attained by different protocols in a scenario with four contending flows and asymmetric link capacities.

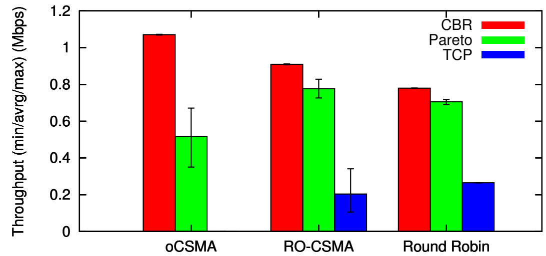


Fig. 5: Results obtained by different protocols in a scenario with competing heterogeneous traffic flows.

II, which provide a baseline for comparison. All protocols operate over the 802.11a PHY layer [3]. The implemented optimal CSMA protocol follows the specifications and configurations from previous work in [13], except for the cases stated otherwise. In the following, we refer as *oCSMA* and *RO-CSMA* respectively to the implementation of the optimal CSMA protocol, and to that of our solutions from Section III. We use measures of proportional fairness, such as logarithmic network utility and throughput product, as the main performance measure for evaluation, and other measures depending on the scenario to better highlight the differences among the three evaluated solutions.

B. Channel asymmetries

In Section III-D, we have shown that our solution maximizes proportional-fair throughput with no assumptions on the symmetry of channels. In this section, we validate such a feature against other protocols in scenarios with multiple contending flows over links with asymmetric modulation rates.

The simulated network comprises 4 flows in contention range, over links at 6 Mbps, 12 Mbps, 24 Mbps, and 54 Mbps. Fig. 4 shows the obtained results in terms of per-flow throughput for our solution compared to 802.11, optimal CSMA and the optimal throughput distribution determined from the effective channel capacities of 5.5 Mbps, 9.5 Mbps, 17 Mbps, and 32 Mbps measured separately at each link in isolation.

The results show a throughput distribution much closer to the optimal by our approach, with an average error of 5.4% against a 53.2% by 802.11 and 22.4% by optimal CSMA. In addition, our design delivers twice the throughput product of optimal CSMA and four times more than 802.11, while at the same time increasing the total throughput (15.6 Mbps against 8.9 Mbps by 802.11 and 11.4 Mbps by optimal CSMA).

Finally, we verify that such performance gains are due to the transmission-time based optimization, described in Section III-C. Due to the symmetry of the evaluated scenario, the transmission-time distribution that maximizes throughput assigns equal (maximum) transmission time to all flows. However, in optimal CSMA the flow over the poorest link receives up to 54% more data transmission time than others, while in 802.11 such difference can raise up to 455%. Instead, with our transmission-time based adaptation, the available channel time is evenly distributed, with a maximum difference among *all* flow pairs of 13%.

C. Traffic heterogeneity

We evaluate next the impact of traffic heterogeneity in system performance. The purpose is to validate the mechanisms derived in Section III-C, designed to deliver optimal adaptation with no assumptions on traffic arrival processes at MAC-layer queues, in situations with multiple competing flows of different traffic types.

To this end, we simulate a network with one *Access Point (AP)* and 9 clients. We generate concurrent downlink flows from the AP to each client, comprising; (i) 3 Constant Bitrate (CBR) flows, which generate an arrival process at MAC-layer queues similar to the one assumed by the optimization models in Section II; (ii) 3 bursty traffic flows following a Pareto ON-OFF model, which interleave idle periods at the source with periods of high packet arrivals to the MAC layer, and; (iii) 3 long-lived TCP flows, which limit the number of in-flow packets using a window-based congestion control mechanism. The bitrate of CBR flows and Pareto ON-OFF flows during ON periods is 1.5 Mbps. The Pareto distribution parameters are $\{\alpha_{\text{OFF}} = 1.5, \alpha_{\text{ON}} = 1.1, b_{\text{OFF}} = 4.17, b_{\text{ON}} = 1.82\}$.

We evaluate this scenario with multiple downlink flows from the same AP, both because it is a common scenario occurring in practice, and also to abstract from the problems of high channel contention with multiple transmitters,[5] which will be studied separately in Section IV-D. Also, with shared transmitters optimal CSMA resolves contention by a deterministic decision that *always* serves the local flow with higher contention aggressiveness (and so does RO-CSMA), which allows us to evaluate these adaptive mechanisms under more strict contention conditions. We compare the performance of optimal CSMA and our solution against that of a *Round Robin* scheduling mechanism, which constitutes an ideal solution for this case as it continuously executes a scheduling cycle where all backlogged flows receive equal service priorities.

We measure the performance of each flow in terms of throughput, and summarize the results by grouping flows of the same type in Fig. 5. For fairness of comparison, the throughput attained by Pareto ON-OFF flows is measured

[5]In this scenario the total number of flows at the MAC layer is 12, with 3 reverse flows for TCP acknowledgments. However, if the AP is shared, the total number of transmitters reduces to 4.

during periods of positive backlog. The results show a highly biased throughput distribution, with a clear dominance by CBR and ON-OFF traffic flows.

With optimal CSMA, all TCP flows completely starve. Indeed, TCP's slow start mechanism begins by placing a transmission unit at the flow's MAC-layer queue, and refrains from injecting more packets until a TCP-ACK is received. However, such a transmission unit is *never* transmitted, since other flows with high bitrate continuously maintain a longer queue (and thus a higher contention aggressiveness by the use of rule (2)). Also the throughput of Pareto ON-OFF flows is significantly lower than for other protocols (27% less on average than for Round Robin and 33% less than for our solution) for similar reasons. When a Pareto transmitter switches to idle, the remaining packets at the MAC layer experience longer service delays as the flow queue gets shorter, rapidly dropping down the flow throughput.

With our solution instead, fairness is restored by increasing the throughput of the lowest performing flows, delivering more than 200 kbps on average to TCP flows against zero throughput by optimal CSMA while increasing the average throughput of Pareto ON-OFF flows by 50%. We observe that such performance gains are due to the use of service meters in the place of queues for contention aggressiveness adaptation. In fact, for any backlogged flow, we measure positive injection rates at the service meter even if the arrival of packets from upper layers has been interrupted. This allows to deliver optimal CSMA adaptation to TCP flows even during slow start, and to maintain high service at all backlogged Pareto flows, even after the source has already entered the OFF state. As a result, we observe high fairness gains, with about 46% increase in network logarithmic utility with respect to optimal CSMA.

D. Collisions under high contention

1) Evaluated scenarios: Finally, we evaluate the protocol features for robustness to interference under high contention presented in Section III-E. Such a mechanism iterates over two main steps. First, each node in the network passively detects surrounding flows by overhearing packet transmissions over the channel. Second, all flows in the channel execute the consensus protocol in Algorithm 2 to determine the maximum level of contention at any point of the network itself, and use that measure to symmetrically update the value of V at all transmitters.

Hence, in the following we divide the evaluation of robustness to interference into two parts. First, we consider symmetric scenarios where all flows have the same view of the wireless channel. This allows to show the precise gains due to the adaptation of V under different contention levels, independently from the features required to attain consensus among flows. Second, we extend the evaluation to the case of multi-hop networks, where different nodes may observe different contention levels depending on their position. This second case evaluates the robustness of the system as a whole, in challenging scenarios where consensus among multiple flows is critical, and for which the effectiveness of Algorithm 2 becomes a central aspect.

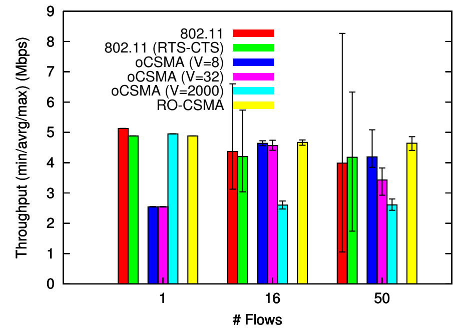


Fig. 6: Per-flow throughput normalized times the number of contending flows in fully connected networks with different sizes.

In all evaluated scenarios, we compare the performance of our solution to that of optimal CSMA with different V choices spanning three orders of magnitude (2, 8, 32, 125, 500, and 2000). Also, we consider a wide-range of network sizes up to 50 flows under symmetric contention and up to 32 flows for randomly generated networks. For brevity, we present a subset of the obtained results in this section. However, the conclusions derived apply to the entire results set. All flows are elastic UDP flows and all transmitters operate at 6 Mbps. The scenarios discussed in Section V extend this study to the case with heterogeneous traffic patterns and asymmetric link modulation rates, using random networks of up to 48 flows.

2) Symmetric contention scenarios: Fig. 6 shows per-flow throughput in a fully-connected network with different number of flows. All measures were normalized by multiplying them to the number of flows in each scenario, to provide for a better comparison.

We observe that the best configuration of optimal CSMA is highly dependent on the network contention level. For small topologies, a high value of V such as 2000 minimizes the nominal optimization error, leading to high access and short silent times. A low V, instead leads to unnecessarily long delays, with up to 50% less throughput in the same scenarios. In contrast, with high contention a large V leads to excessive contention aggressiveness, which in turn raises the collision probability of RTS/CTS packets up to 6 times, and degrades throughput up to 44%. In those cases, a lower value of V restores high performance by decreasing the aggressiveness of all transmitters, thus reducing collisions.

With the automatic V-adaptation mechanism derived in Section III-E, V is adapted to high values in small networks yielding high channel utilization, but gradually lowered under increasingly high contention levels to avoid interference. We verify that the intended value of V is attained at *all nodes* in each of the evaluated scenarios. Furthermore, all flows are quickly detected by passive detection (i.e., overhearing ongoing transmissions), with the worst delay in detection of only 1.5s observed in the network with 50 flows.

Finally, 802.11 with BEB also delivers high total throughput across all evaluated environments. However, in high contention environments, the asymmetries in CW lengths at different transmitters due to frequent collisions yields short-

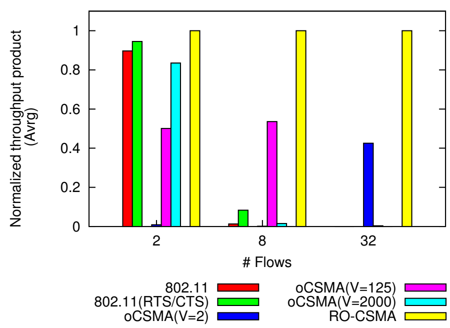


Fig. 7: Average normalized throughput product over 100 random networks for different sizes.

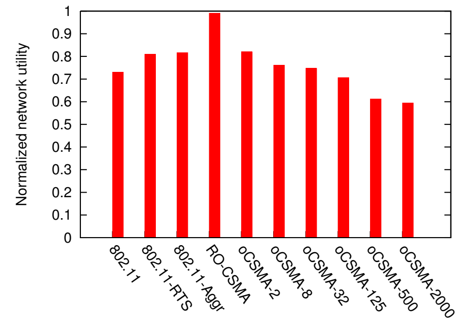


Fig. 8: Normalized sum of the throughput logs attained by different protocols on average over 100 randomly-generated topologies with multiple per-link modulation rates and heterogeneous traffic patterns.

term unfairness, with high disparities in the throughput attained by different flows (with up to 21 times maximum deviation from the mean than for our solution).

3) Randomly generated multi-hop networks: We verified in Section IV-D2 that the automatic adaptation of the V parameter yields high performance gains in networks with symmetric contention. However, in the general case, the density of flows can differ at multiple network points, yielding asymmetric contention levels. Next, we evaluate the performance of the mechanisms presented in Section III-E in randomly-generated networks that introduce additional challenges for the protocol operation. In particular, the operation of Algorithm 2 becomes a central aspect for nodes to agree in a common measure of the network contention level.

We generate random networks using uniformly-distributed node coordinates within a 3000m$\times$3000m terrain. Flows are generated by randomly selecting connected node pairs (within a transmission range of 184m). For a meaningful analysis based on network size, we filter-out disconnected networks (otherwise each network partition can operate independently yielding results similar to networks with smaller size). All results shown in this section are averages obtained over 100 randomly-generated networks. We measure the performance attained by each protocol in terms of throughput products. Since the range of result values is highly dependent on the topology, before averaging the throughput products over all scenarios we normalize them over the maximum value attained by any protocol in each case. The length of epochs for the operation of Algorithm 2 is set to 2.5s.

Fig. 7 shows the results obtained for different network sizes of 2, 8, and 32 flows. As the network size increases, smaller values of V tend to raise performance due to a higher robustness to collisions at high-contention network points. However, we observe that the network density can vary significantly even for a fixed network size, such that no fixed value of V works well in all cases with the same number of flows. Because of this, our approach exhibits higher performance, specially in larger networks spanning a wide-range of possible network densities.

Finally, we evaluate the effectiveness of Algorithm 2 in determining the maximum network contention level Ω_n. In all executed simulation runs, Algorithm 2 converges to the maximum value within 2 epochs (i.e., 5s). We further observe that the first epoch is necessary for each node to passively detect all neighbor flows, such that during the second epoch, updated measures can be shared by all transmitters.

V. Randomly generated networks

In the previous section we evaluated the solutions proposed in Section III separately, in different scenarios with channel asymmetries, heterogeneous traffic and different levels of contention. In this section, we evaluate the joint operation of these three solutions in more challenging scenarios where all the above conditions appear combined, simultaneously affecting the protocol operation in different ways.

To this end, we generate networks using uniformly-distributed random node coordinates within a 3000m$\times$3000m terrain. Flows are generated by randomly selecting connected node pairs (within a transmission range of 184m). All results shown in this section are averages obtained over 100 randomly-generated networks. We consider different network sizes, with 16, 32, and 48 flows, obtaining qualitatively similar results for each of them. Here, we present the results obtained for the case with 48 flows, which is the one with the widest range of contention levels. We randomly assign different traffic types to each flow including CBR, Pareto ON-OFF, and TCP traffic. To increase traffic diversity, we consider 4 possible bitrates for CBR and Pareto ON-OFF flows (0.5 Mbps, 1 Mbps, 1.5 Mbps, and 2 Mbps), randomly assigned among flows of such types. The modulation rate at each transmitter is assigned according to the SNR measured at the receiver in isolation, so that shorter links use higher modulation rates. The V values considered for optimal CSMA are 2, 8, 32, 125, 500, and 2000, which is the range of values used in the solution from Section III-E.

We measure protocol performance in terms of logarithmic network utility. We normalize the obtained measures by the maximum utility attained by any protocol in the same run, and average the normalized results over the 100 considered instances. The results, depicted in Fig. 8, show that our design outperforms other solutions, with a 21%-68% utility increase over optimal CSMA with different configurations, and more than 21% average utility gain over 802.11. Furthermore, the cumulative density function of logarithmic network utility in

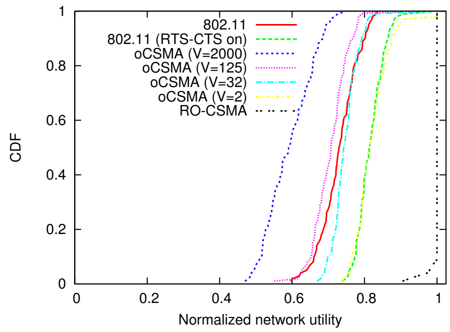


Fig. 9: Cumulative density function of the normalized logarithmic utility attained by different protocols over 100 randomly-generated topologies with 48 flows.

Fig. 9 shows that in more than 90% of the generated network instances, our solution delivers higher utility than the other evaluated protocols.

VI. Related work

There has been extensive research in distributed CSMA optimization. The numerous works in this field come in a variety of flavors depending on the research approach, system model, performance objective, and other aspects. In the following, we present a broad classification that provides an overview of the entire area while contrasting our contributions to prior work.

Analytical works: Multiple analytical works were devoted to the design and study of distributed CSMA algorithms that maximize different performance measures [5]–[10]. Such works differ on the analytical techniques in use, system model, and the level of overhead in the proposed solutions. For example, [5] was the first to show the throughput-optimality of distributed CSMA under the multi-hop model. [6], instead proposed utility-optimal mechanisms based on a fixed-point approximation of the network performance. [7] derives an alternative solution for throughput maximization under slotted time. [8] proposes a generalized version of the algorithm in [5] for utility maximization with no message passing. All these works address network performance optimization through rigorous analytical means within a well-defined set of assumptions. While we also make use of analysis to support our design, our main focus is on addressing sources of high performance degradation for optimal CSMA to deliver robust operation in a wide range of operating settings.

Experimental works: Other works focus on the implementation and experimental evaluation of the systems above described [11]–[13]. The early experiences in [11], [12] mostly focus on implementation aspects, such as the main challenges for system implementation over existing hardware, respectively for the case of wireless networks and wireless sensor networks. The work in [13] evaluates optimal CSMA to identify how different factors affect its performance in practical operational settings. Our work is in part motivated by such studies, which first identified the main sources of performance degradation for optimal CSMA addressed here.

Other works: Like our work, others have also noted the problematic effect of different assumptions in the optimal CSMA models, for example [14]–[18]. However, our approach differs in a number of aspects. For example, while [15] improves the operation of TCP over optimal CSMA via modifications to the transport layer, we propose an extension to optimal CSMA itself that allows optimal adaptation with heterogeneous traffic. And, while [16], [17] propose the use of reservation mechanisms like RTS-CTS to limit the effects of collisions, we are the firsts to study the optimization-robustness conflict under high levels of contention, and propose an adaptive solution.

VII. Conclusion

In this document, we address the main sources of performance degradation in optimal CSMA, to derive a distributed system for proportional fairness in networks with channel asymmetries, heterogeneous traffic, and high contention. We propose a novel approach to design that combines robustness with optimization, to overcome the high performance degradation introduced in optimal CSMA by such factors. Our contributions drive the development of future robust and optimal CSMA, enabling high performance across a broad range of network operating conditions.

References

[1] R. R. Boorstyn, A. Kershenbaum, B. Maglaris, and V. Sahin, "Throughput analysis in multihop CSMA packet radio networks, *IEEE Trans. on Communications*, Vol. COM-35, No. 3, pp. 267-274, March 1987.

[2] L. Kleinrock and F. A. Tobagi, "Packet switching in radio channels: Part I - Carrier Sense Multiple-Access modes and their throughput-delay characteristics," *IEEE Trans. on Communications*, Vol. COM-23, No. 12, pp. 1400-1416, Dec. 1975.

[3] IEEE 802.11 Task Group, "Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications," *IEEE standard*, 802.11-1999, 1999.

[4] M. Garetto, J. Shi, and E. W. Knightly, "Modeling Media Access in Embedded Two-Flow Topologies of Multi-hop Wireless Networks," in *Proc. of ACM MobiCom 2005*, Cologne, Germany, Aug. 2005.

[5] L. Jiang and J. Walrand, "A distributed CSMA algorithm for throughput and utility maximization in wireless networks," in *Proc. of the Allerton Conference*, Monticello, IL, 2008.

[6] P. Marbach and A. Eryilmaz, "A backlog-based CSMA mechanism to achieve fairness and throughput-optimality in multihop wireless networks, in *Proc. of the Allerton Conference*, Monticello, IL, 2008.

[7] J. Ni and R. Srikant, "Distributed CSMA/CA algorithms for achieving maximum throughput in wireless networks, in *Proc. of the ITA Workshop*, San Diego, CA, 2009.

[8] J. Liu, Y. Yi, A. Proutiere, M. Chiang, and H. V. Poor, "Towards utility-optimal random access without message passing," *Wiley Journal of Wireless Communications and Mobile Computing*, vol. 10, no. 1, pp. 115-128, Jan. 2010.

[9] H. Li and N. H. Vaidya, "Optimal CSMA-based Wireless Communication with Worst-case Delay and Non-uniform Sizes," in *Proc. of IEEE INFOCOM*, Toronto, Canada, April 2014.

[10] S. Li, E. Ekici, and N. Shroff, "Throughput-Optimal Queue Length Based CSMA/CA Algorithm for Cognitive Radio Networks," in *IEEE Trans. on Mobile Computing*, vol. 99, 2014.

[11] A. Sridharan, S. Moeller, and B. Krishnamachari, "Implementing backpressure-based rate control in wireless networks, in *Proc. of the ITA Workshop*, San Diego, CA, 2009.

[12] J. Lee, J. Lee, Y. Yi, S. Chong, A. Proutiere, and M. Chiang, "Implementing utility-optimal CSMA, in *Proc. of the Allerton Conference*, Monticello, IL, 2009.

[13] B. Nardelli, J. Lee, K. Lee, Y. Yi, S. Chong, E. W. Knightly, and M. Chiang, "Experimental evaluation of Optimal CSMA," in *Proc. of IEEE INFOCOM*, Shanghai, China, April 2011.

[14] B. Ji, C. Joo, and N. B. Shroff, "Delay-Based Back-Pressure Scheduling In Multi-Hop Wireless Networks," in *Proc. of IEEE INFOCOM*, Shanghai, China, April 2011.

[15] W. Chen, Y. Wang, M. Chen and S. C. Liew, "On the Performance of TCP over Throughput-Optimal CSMA," in *IWQoS 2011*, San Jose, CA, June 2011.

[16] L. Jiang and J. Walrand, "Approaching Throughput-Optimality in a Distributed CSMA Algorithm: Collisions and Stability," in *Proc. of ACM MobiHoc 09*, New Orleans, LA, May 2009.

[17] T. H. Kim, J. Ni, R. Srikant and N. H. Vaidya, "On the achievable throughput of CSMA under imperfect carrier sensing," in *Proc. of IEEE INFOCOM*, Shanghai, China, April 2011.

[18] S. Subramanian, E. L. Pasiliao, J. M. Shea, J. W. Curtis, W. E. Dixon, "Throughput Maximization in CSMA Networks with Collisions and Hidden Terminals," in *Springer Proc. in Mathematics and Statistics*, Vol. 20, pp 195-205, 2012.