Analyzing DISH for Multi-Channel MAC Protocols in Wireless Networks

Tie Luo and Mehul Motani
Electrical & Computer Engineering
National University of Singapore
{tie,motani}@nus.edu.sg

Vikram Srinivasan
Bell Labs Research
India
vikramsr@alcatel-lucent.com

ABSTRACT

For long, node cooperation has been exploited as a *data relaying* mechanism. However, the wireless channel allows for much richer interaction between nodes. One such scenario is in a multi-channel environment, where transmitter-receiver pairs may make incorrect decisions (e.g., in selecting channels) but idle neighbors could help by sharing information to prevent undesirable consequences (e.g., data collisions). This represents a *Distributed Information SHaring* (DISH) mechanism for cooperation and suggests new ways of designing cooperative protocols. However, what is lacking is a theoretical understanding of this new notion of cooperation. In this paper, we view cooperation as a network resource and evaluate the *availability of cooperation* via a metric, p_{co}, the probability of obtaining cooperation. First, we analytically evaluate p_{co} in the context of multi-channel multi-hop wireless networks. Second, we verify our analysis via simulations and the results show that our analysis accurately characterizes the behavior of p_{co} as a function of underlying network parameters. This step also yields important insights into DISH with respect to network dynamics. Third, we investigate the correlation between p_{co} and network performance in terms of collision rate, packet delay, and throughput. The results indicate a near-linear relationship, which may significantly simplify performance analysis for cooperative networks and suggests that p_{co} be used as an appropriate performance indicator itself. Throughout this work, we utilize, as appropriate, three different DISH contexts — model-based DISH, ideal DISH, and real DISH — to explore p_{co}.

Categories and Subject Descriptors

C.4 [**Performance Of Systems**]: *Performance attributes*; C.2.1 [**Computer-Communication Networks**]: Network Architecture and Design—*Wireless communication*

General Terms

Theory, Performance

Keywords

Distributed information sharing, cooperative communication

1. INTRODUCTION

Cooperative diversity is not a new concept in wireless communications. Key ideas and results in cooperative communications can be traced back to the 1970s to van der Meulen [1] and Cover & El Gamal [2], whose works have spurred numerous studies on this topic from an information-theoretic perspective (e.g., [3–6]) or a protocol-design perspective (e.g., [7–10]). To date, cooperation has been intensively studied in various contexts. However, to the best of our knowledge, it has always been used as a *data relaying* mechanism where intermediate nodes help relay packets from a transmitter to a receiver. In fact, the wireless channel allows for much richer interaction among nodes. Consider a scenario where orthogonal frequency channels are available. A node wishes to select a conflict-free channel to transmit data, but may often fail to achieve this due to lack of sufficient information about channel usage. In this case, other nodes in the neighborhood may possess the information in need and thus could help by sharing this information.

This shows that cooperation can be used as a *Distributed Information SHaring* (DISH) mechanism, in addition to mere data relaying. In [11, 12], we proposed a multi-channel MAC protocol based on this idea, where performance enhancement was demonstrated via simulations. In this paper, we develop a theoretical treatment of this new notion of cooperation, in particular, the *availability of cooperation*. The benefit of DISH is that it can remove the need of using multiple transceivers [13–17] and time synchronization [18–23] in designing multi-channel MAC protocols. This motivates us to understand DISH from a theoretical perspective.

In this paper, we define a metric p_{co} which characterizes the availability of cooperation as *the probability of obtaining cooperation* (see Def. 3 for a more precise definition). We analytically evaluate this metric in multi-channel multi-hop wireless networks with randomly distributed nodes, and verify the analysis via simulations. We also carry out a detailed investigation of p_{co} with three different contexts of DISH: model-based DISH, ideal DISH, and real DISH, in order to obtain meaningful findings.

1.1 Summary of Contributions

Our aim in this paper is to understand DISH and the availability of cooperation (p_{co}) from an analytical perspective. More specifically, we provide an analysis which accurately characterizes the availability of cooperation as a function of the underlying network parameters. This analysis reveals *what* underlying factors and *how* these factors affect cooperation, and can provide guidelines to provisioning the network to increase performance.

Throughput analysis for multi-hop networks is difficult (and still an open problem in general), and it gets even more complicated in a multi-channel context with DISH. Our approach in this paper is to first look at p_{co} and then correlate it with network performance. The results indicate that there is a simple relationship between p_{co} and several performance metrics.

The specific findings of this study are:

1. The availability of cooperation is high ($p_{co} > 0.7$) in typical cases, which suggests that DISH is feasible to

MobiHoc'08, May 26–30, 2008, Hong Kong SAR, China.

use in multi-channel MAC protocols.

2. The performance degradation due to an increase in node density can be alleviated due to the simultaneously increased availability of cooperation.

3. The metric p_{co} will increase for larger packet sizes for a given *bit* arrival rate, but will decrease for larger packet sizes for a given *packet* arrival rate.

4. Node density and traffic load have opposite effects on p_{co} but node density is the dominating factor. This implies an improved scalability for DISH networks as p_{co} increases with node density.

5. p_{co} is strongly correlated to network performance and has a near-linear dependence with metrics such as throughput and delay. This may significantly simplify performance analysis for cooperative networks, and suggests that p_{co} be used as an appropriate performance indicator itself.

2. RELATED WORK

There are three other studies most related to this work. One is CAM-MAC [11] which uses cooperation in the new way that we call DISH in this paper. It is a cooperative multi-channel MAC protocol requiring only a single transceiver and no synchronization. In this protocol, there is a control channel for transmitter-receiver pairs to perform handshakes in order to reserve data channels, while nodes in the neighborhood may send cooperative messages to invalidate the handshake if the selected channel or receiver is busy. As it is the only work we are aware of that explicitly uses DISH, our system model will use a protocol framework by abstracting the work in [11].

The second work, CoopMAC [10], is also a cooperative MAC protocol which exploits data relaying as many other protocols do, such as [7–9]. A protocol analysis is provided in the paper and it requires computing the probability that a *relay node* is available. This probability is different from p_{co} in that it is determined by the *static locations* of nodes, i.e., whether a node exists in a specific region. The probability is computed via geometric analysis (nodes are assumed to be uniformly distributed). On the other hand, p_{co} is determined not only by static node locations, but also by *dynamic node behavior*, e.g, a node must have acquired the specific information at a specific moment. The second main difference between the protocol analysis of CoopMAC and our work is that the problem context of CoopMAC is a *wireless LAN* with *a single channel*, whereas this paper assumes a multi-hop network with multiple channels.

The last work is by Han et al. [24] who considered a multi-channel MAC protocol adopting ALOHA on the control channel to reserve data channels. A queueing-theoretic approach was taken to calculate throughput for the protocol. However, there are some noteworthy limitations. First, only a *single-hop* scenario was considered. Second, each node was assumed to be able to communicate on the control channel and a data channel simultaneously. This essentially requires *two transceivers* per node, and consequently leads to collision-free data channels, which oversimplifies the problem. Third, a unique virtual queue was assumed to store the packets arriving at *all* nodes for the ease of *centralized transmission scheduling*, and the precise status of the queue was assumed to be known to the entire network. This assumption is impractical and eventually results in a throughput upper bound. Fourth, the access to the control channel adopts the ALOHA algorithm, rather than the more practical and sophisticated mechanism of CSMA/CA.

3. SYSTEM MODEL

We consider a static and connected ad hoc network in which each node is equipped with a single half-duplex transceiver that can dynamically switch between a set of orthogonal frequency channels but can only use one at a time. One channel is designated as a control channel and the others are designated as data channels. Nodes are placed in a plane area according to a two-dimensional Poisson point process.

We consider a class of multi-channel MAC protocols with their common framework described below. A transmitter-receiver pair uses a McRTS/McCTS handshake on the control channel to set up communication (like 802.11 RTS/CTS) for their subsequent DATA/ACK handshake on a data channel. To elaborate, a transmitter sends a McRTS on the control channel using CSMA/CA, i.e., it sends McRTS after sensing the control channel to be idle for a random period (addressed below) of time. The intended receiver, after successfully receiving McRTS, will send a McCTS and then switch to a data channel (the McRTS informs the receiver of the data channel). After successfully receiving the McCTS, the transmitter will also switch to its selected data channel, and otherwise it will backoff on the control channel for a random period (addressed below) of time. Hence it is possible that only the receiver switches to the data channel. After switching to a data channel, the transmitter will send a DATA and the receiver will respond with a ACK upon successful reception. Then both of them switch back to the control channel.

In the above we have mentioned two random periods of time. Our model does not specify them but assumes that these periods are designed such that idle intervals on the control channel are well randomized. Specifically, when a node is on the control channel, it sends control messages (an aggregated stream of McRTS and McCTS) according to a Poisson process.

Note that we use McRTS, McCTS, DATA and ACK to refer to different packets (frames) without assuming specific frame formats. Since, logically, they must make a protocol functional, we assume that McRTS carries channel usage information (e.g., “who will use which channel for how long”) and, for simplicity, McCTS is the same as McRTS.

We assume that, after switching to a data channel, a node will stay on that channel for a period of T_d, where T_d is the duration of a successful data channel handshake. We ignore channel switching delay as it will not fundamentally change our results if it is negligible compared to T_d (the delay is $80\mu s$ [23] while T_d is more than $6ms$ for a 1.5KB data packet on a 2Mbps channel). We also ignore SIFS and propagation delay for the same reason, provided that they are smaller than the transmission time of a control message.

We assume a uniform traffic pattern — all nodes have the equal data packet arrival rate, and for each data packet to send, a node chooses a receiver equally likely among its neighbors. We also assume a stable network — all data packets can be delivered to destinations within finite delay. In addition, packet reception fails if and only if packets collide with each other (i.e., no capture effect), transmission and interference ranges are equal, and the probability that *neighboring* nodes simultaneously *start* sending control messages is zero (no time synchronization).

We do not assume a specific channel selection strategy; how a node selects data channels will affect how often conflicting channels are selected, but will not affect p_{co}. This is because, intuitively, we only care about the availability of cooperation (p_{co}) when a multi-channel coordination problem (a precise definition is given in Def. 1), which includes channel conflicting problem, *has been* created.

We do not assume a concrete DISH mechanism, i.e., nodes do not physically react upon a multi-channel coordination problem, because analyzing the availability of cooperation does not require the use of this resource. In fact, assuming one of the (numerous possible) DISH mechanisms will lose generality. Nevertheless, we will show in Section 5 that, when an ideal or a real DISH mechanism is used, the results do not fundamentally change. This could be an overall effect from contradicting factors which will be explained therein.

The following lists all parameters that are assumed known:

- n: node density. In a multi-hop network, it is the average number of nodes per R^2 where R is the transmission range. In a single-hop network, it is the total number of nodes.
- λ: the average data packet arrival rate at each node, including retransmissions.
- T_d: the duration of a data channel handshake.
- b: the transmission time of a control message. $b \ll T_d$.

4. ANALYSIS

4.1 Problem Formulation and Analysis Outline

We first formally define p_{co}, which depends on two concepts called the *MCC problem* and the *cooperative node*.

Definition 1 (MCC Problem) *A multi-channel coordination (MCC) problem is either a* channel conflict *problem or a* deaf terminal *problem. A channel conflict problem is created when a node, say y, selects a channel to use (transmit or receive packets) but the channel is already in use by a neighboring node, say x. A deaf terminal problem is created when a node, say y, initiates communication to another node, say x, that is however on a different channel. In either case, we say that an MCC problem is created by x and y.*

In a protocol that transmits `DATA` without requiring `ACK`, a channel conflict problem does not necessarily indicate an impending data collision. We do not consider such a protocol.

Definition 2 (Cooperative Node) *A node that identifies an MCC problem created by two other nodes, say x and y, is called a cooperative node with respect to x and y.*

See Fig. 1 for a visualization based on our system model.

Definition 3 (p_{co}) *p_{co} is the probability for two arbitrary nodes that create an MCC problem to obtain cooperation, i.e., there is at least one cooperative node with respect to these two nodes.*

Note that, if there are multiple cooperative nodes and a DISH mechanism allows those nodes to send cooperative messages concurrently, then a collision results. However, this collision *still indicates an MCC problem* and thus cooperation is still deemed obtained. CAM-MAC [11] also implements this.

We distinguish the receiving of control messages. A transmitter receiving `McCTS` from its intended receiver is referred to as *intentional receiving*, and the other cases of receiving are referred to as *overhearing*, i.e., any node receiving `McRTS` (hence an intended receiver may also be a cooperative node) or any node other than the intended transmitter receiving `McCTS`.

Our notation is listed in Table 1. Overall, we will determine p_{co} by following the order of $p_{co}^{xy}(v) \to p_{co}^{xy} \to p_{co}$.

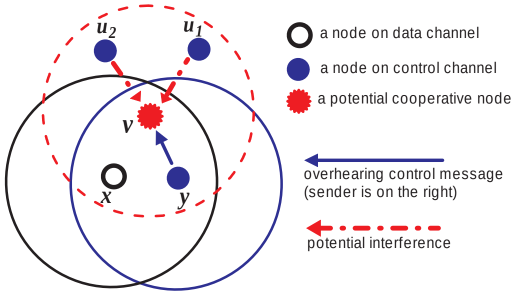

u2
u1
v
x
y
a node on data channel
a node on control channel
a potential cooperative node
overhearing control message
(sender is on the right)
potential interference

Figure 1: Illustration of an MCC problem and a cooperative node. Node x is performing a data channel channel handshake on CH_x, and y has just sent a control message during a control channel handshake. If this control message is an `McRTS` addressed to x, then a *deaf terminal* problem is created. If this control message indicates that y selects CH_x (recall that a control message carries channel usage information), a *channel conflict* problem is created. In either case, if a third node v identifies this problem (by overhearing x's and y's control messages successively), it is a cooperative node.

Consider $p_{co}^{xy}(v)$ first. Fig. 1 illustrates that node v is cooperative if and only if it successfully overhears x's and y's control messages successively. Hence $\forall v \in \mathcal{N}_{xy}$,

$$\begin{aligned} p_{co}^{xy}(v) &= \Pr[\mathcal{O}(v \leftarrow x), \mathcal{O}(v \leftarrow y)] \\ &= \Pr[\mathcal{O}(v \leftarrow x)] \cdot \Pr[\mathcal{O}(v \leftarrow y) | \mathcal{O}(v \leftarrow x)]. \end{aligned} \quad (1)$$

Consider $\mathcal{O}(v \leftarrow i)$. For v to successfully overhear i's control message which is being sent during interval $[s_i, s_i + b]$, v must be silent on the control channel and not be interfered, i.e.,

$$\begin{aligned} \Pr[\mathcal{O}(v \leftarrow i)] = &\Pr[\mathcal{S}_v(s_i, s_i + b), \bigcap_{u \in \mathcal{N}_v \setminus \{i\}} \mathcal{I}_u(s_i, s_i + b)], \\ &\forall v \in \mathcal{N}_i. \end{aligned} \quad (2)$$

Now we outline our analysis as below.

- Section 4.2: solves (2), with p_{ctrl} and λ_c introduced.
- Section 4.3: solves p_{ctrl} and λ_c.
- Section 4.4: solves (1) and then the target metric p_{co}.
- Section 4.5: special-case study in single-hop networks.

4.2 Solving Equation (2)

Proposition 1 *If node u is on a data channel at t_1, then the probability that u does not introduce interference to the control channel during $[t_1, t_2]$, where $t_2 - t_1 = \Delta t < T_d$, is given by*

$$\Pr[\mathcal{I}_u(t_1, t_2) | \overline{\mathcal{C}_u(t_1)}] = 1 - \frac{\Delta t}{T_d} + \frac{1 - e^{-\lambda_c \Delta t}}{\lambda_c T_d}.$$

PROOF. By the total probability theorem,

$$\begin{aligned} l.h.s. = &\Pr[\Omega_u(t_1, t_2)] \times \Pr[\mathcal{I}_u(t_1, t_2) | \Omega_u(t_1, t_2)] \\ &+ \Pr[\overline{\Omega_u(t_1, t_2)}] \times 1. \end{aligned}$$

Let t_{sw} be the time when node u switches to the control channel (see Fig. 2). It is uniformly distributed in $[t_1, t_1 + T_d]$

Table 1: Notation

Probabilities	p_{co}^{xy}	the probability that at least one cooperative node with respect to x and y exists
	$p_{co}^{xy}(v)$	the probability that node v is a cooperative node with respect to x and y
	p_{ctrl}	the probability that a node is on the control channel at an arbitrary point in time
	p_{succ}	the probability that a control channel handshake (initiated by a McRTS) is successful
	p_{oh}	the probability that an arbitrary node successfully overhears a control message
Events	$\mathcal{C}_v(t)$	node v is on the control channel at time t
	$\mathcal{O}(v \leftarrow i)$	node v successfully overhears node i's control message, given that i sends the message
	$\mathcal{S}_v(t_1, t_2)$	node v is silent (not transmitting) on the control channel during interval $[t_1, t_2]$
	$\mathcal{I}_v(t_1, t_2)$	node v does not introduce interference to the control channel during interval $[t_1, t_2]$, i.e., it is on a data channel or is silent on the control channel.
	$\Omega_u(t_1, t_2)$	node u, which is on a data channel at t_1, switches to the control channel in $[t_1, t_2]$
Others	$\mathcal{N}_i, \mathcal{N}_{ij}, \mathcal{N}_{v\backslash i}$	$\mathcal{N}_i$ is the set of node i's neighbors, $\mathcal{N}_{ij} = \mathcal{N}_i \cap \mathcal{N}_j$, $\mathcal{N}_{v\backslash i} = \mathcal{N}_v \backslash \mathcal{N}_i \backslash \{i\}$ (v's but not i's neighbors)
	$K_{ij}, K_{v\backslash i}$	$K_{ij} = \lvert\mathcal{N}_{ij}\rvert$, $K_{v\backslash i} = \lvert\mathcal{N}_{v\backslash i}\rvert$
	s_i	the time when node i starts to send a control message
	$\lambda_c, \lambda_{rts}, \lambda_{cts}$	the average rates of a node sending control messages, McRTS, and McCTS, respectively, *when it is on the control channel.* Clearly, $\lambda_c = \lambda_{rts} + \lambda_{cts}$.

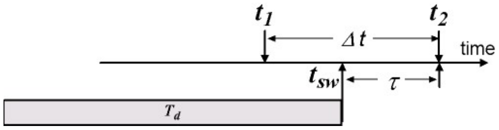


Figure 2: A node switches to the control channel after data channel handshaking.

because the time when u started its data channel handshake is unknown, and hence

$$\Pr[\Omega_u(t_1, t_2)] = \frac{\Delta t}{T_d}. \tag{3}$$

Since control channel traffic is Poisson with rate λ_c,

$$\Pr[\mathcal{I}_u(t_1, t_2) | \Omega_u(t_1, t_2)] = \Pr[\mathcal{S}_u(t_{sw}, t_2) | \Omega_u(t_1, t_2)] = \mathbb{E}[e^{-\lambda_c \tau}]$$

where $\tau = t_2 - t_{sw}$ is uniformly distributed in $[0, \Delta t]$ by the same argument leading to (3). Hence

$$\mathbb{E}[e^{-\lambda_c \tau}] = \int_0^{\Delta t} e^{-\lambda_c \tau_0} \frac{1}{\Delta t} d\tau_0 = \frac{1 - e^{-\lambda_c \Delta t}}{\lambda_c \Delta t},$$

and then by substitution the proposition is proven. □

Proposition 2 *If node v is overhearing a control message from node i during $[s_i, s_i + b]$, then the probability that a node $u \in \mathcal{N}_v$ does not interfere with v is given by*

$$\Pr[\mathcal{I}_u(s_i, s_i + b)] = \begin{cases} 1, & u \in \mathcal{N}_{vi}; \\ p_{ni\text{-}oh}, & u \in \mathcal{N}_{v\backslash i}. \end{cases}$$

where

$$p_{ni\text{-}oh} = \ p_{ctrl} \cdot e^{-2\lambda_c b} + (1 - p_{ctrl}) \cdot (1 - \frac{2b}{T_d} + \frac{1 - e^{-2\lambda_c b}}{\lambda_c T_d}).$$

Proposition 3 *If node i (transmitter) is intentionally receiving McCTS from node j (receiver) during $[s_j, s_j + b]$, then the probability that a node $u \in \mathcal{N}_i$ does not interfere with i is given by*

$$\Pr[\mathcal{I}_u(s_j, s_j + b)] = \begin{cases} 1, & u \in \mathcal{N}_{ij}; \\ p_{ni\text{-}cts}, & u \in \mathcal{N}_{i\backslash j}. \end{cases}$$

where $p_{ni\text{-}cts}$ =

$$(1 - p_{ctrl})[1 - \frac{b}{T_d}(1 + \frac{b}{T_d} - \frac{1 - e^{-\lambda_c b}}{\lambda_c T_d} - e^{-\lambda_c b})] + p_{ctrl}.$$

Then we can solve (2), based on Prop. 2, to be

$$\Pr[\mathcal{O}(v \leftarrow i)] \approx p_{ctrl}\, p_{ni\text{-}oh}^{K_{v\backslash i}}. \tag{4}$$

See appendix for the proof of Prop. 2, Prop. 3 and Eq. (4).

4.3 Solving p_{ctrl} and λ_c

For p_{ctrl}, consider a sufficiently long period T_0. On the one hand, the number of arrival data packets at each node is λT_0. On the other hand, each node spends total time of $(1 - p_{ctrl})T_0$ on data channels, a factor η of which is used for sending arrival data packets. Since the network is stable (incoming traffic is equal to outgoing traffic), we establish a balanced equation:

$$\lambda T_0 T_d = \eta\ (1 - p_{ctrl}) T_0.$$

To determine η, noticing that a node switches to data channels either as a transmitter (with an average rate of λ) or as a receiver (with an average rate of λ_{cts}), we have $\eta = \lambda/(\lambda + \lambda_{cts})$. Substituting this into the above yields

$$p_{ctrl} = 1 - (\lambda + \lambda_{cts}) T_d. \tag{5}$$

For λ_c (together with λ_{cts}), we need two lemmas.

Lemma 1 *For a Poisson random variable K with mean value $\overline{K}$, and $0 < p < 1$,*

$$\mathbb{E}[p^K] = e^{-(1-p)\overline{K}}.$$

PROOF.

$$\mathbb{E}[p^K] = \sum_0^{\infty} p^k \Pr(K = k) = e^{-\overline{K}} \sum_0^{\infty} \frac{(p\overline{K})^k}{k!} = e^{-(1-p)\overline{K}}.$$

□

Lemma 2 *For three random distributed nodes v, i and j,*

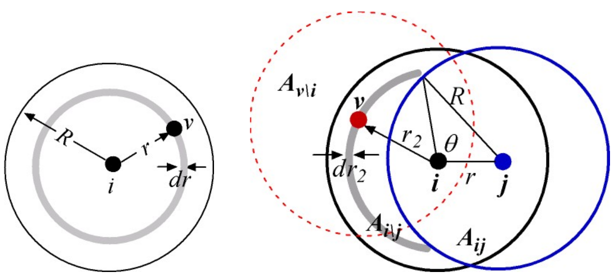
(a) $\mathbb{E}[K_{v\backslash i} | v \in \mathcal{N}_i] \approx 1.30n$.

(b) $\mathbb{E}[K_{v\backslash i} | v \in \mathcal{N}_{ij}] \approx 1.19n$.

(c) $\mathbb{E}[K_{ij}] \approx 1.84n$.

PROOF. Let $A_s(\gamma)$ be the intersection area of two circles with a distance of γ between their centers, and $\gamma < R$, where R is the circles' radius. It can be derived from [25] that

$$A_s(\gamma) = 2R^2 \arccos \frac{\gamma}{2R} - \gamma \sqrt{R^2 - \frac{\gamma^2}{4}}.$$



(a) $v \in \mathcal{N}_i$.

(b) $v \in \mathcal{N}_{i\backslash j}$.

Figure 3: Deriving the pdf of distance $||vi||$.

Let $A_c(\gamma)$ be the complementary area of $A_s(\gamma)$, i.e., $A_c(\gamma) = \pi R^2 - A_s(\gamma)$, and let A_{ij} and $A_{v\backslash i}$ be the areas where $\mathcal{N}_{ij}$ and $\mathcal{N}_{v\backslash i}$ are located, respectively.

(a) See Fig. 3(a). Letting $\gamma = ||vi||$ where $v \in \mathcal{N}_i$, and $f(r)$ be its probability density function (pdf), we have $f(r)dr = 2\pi r dr/(\pi R^2)$, which gives $f(r) = 2r/R^2$. Thus

$$\mathbb{E}[A_{v\backslash i}|v \in \mathcal{N}_i] = \int_0^R A_c(r) f(r) dr \approx 1.30R^2,$$

and hence $\mathbb{E}[K_{v\backslash i}|v \in \mathcal{N}_i] \approx n \cdot 1.30R^2/R^2 = 1.30n$.

(b) Let $\gamma_1 = ||vi||$ where $v \in \mathcal{N}_{ij}$, and $f_1(r_1)$ be its pdf. To solve $f_1(r_1)$, we consider $v \in \mathcal{N}_{i\backslash j}$ instead (see Fig. 3(b)):

$$\begin{aligned} \because\ \mathbb{E}[A_{v\backslash i}|v \in \mathcal{N}_i] &= p_1 \cdot \mathbb{E}[A_{v\backslash i}|v \in \mathcal{N}_{ij}] \\ &\quad + (1-p_1) \cdot \mathbb{E}[A_{v\backslash i}|v \in \mathcal{N}_{i\backslash j}] \\ \text{where } p_1 &\triangleq \Pr[v \in \mathcal{N}_{ij}|v \in \mathcal{N}_i] = \frac{A_s(r)}{\pi R^2}, \\ \therefore\ \mathbb{E}[A_{v\backslash i}|v \in \mathcal{N}_{ij}] &= p_1^{-1}\mathbb{E}[A_{v\backslash i}|v \in \mathcal{N}_i] \\ &\quad - (p_1^{-1} - 1) \cdot \mathbb{E}[A_{v\backslash i}|v \in \mathcal{N}_{i\backslash j}]. \end{aligned} \quad (6)$$

To determine $\mathbb{E}[A_{v\backslash i}|v \in \mathcal{N}_{i\backslash j}]$, let $\gamma_2 = ||vi||$ where $v \in \mathcal{N}_{i\backslash j}$, and $f_2(r_2)$ be its pdf. It is determined by

$$f_2(r_2)dr_2 = \frac{2(\pi - \theta) r_2 dr_2}{A_{i\backslash j}} \text{ and } \cos\theta = \frac{r_2^2 + r^2 - R^2}{2r_2 r}.$$

Therefore

$$\mathbb{E}[A_{v\backslash i}|v \in \mathcal{N}_{i\backslash j}] = \int_{R-r}^{R} A_c(r_2) f_2(r_2) dr_2$$
$$= \int_{R-r}^{R} \frac{2r_2 A_c(r_2)}{A_c(r)} (\pi - \arccos\frac{r_2^2 + r^2 - R^2}{2r_2 r}) dr_2.$$

Substituting this and $\mathbb{E}[A_{v\backslash i}|v \in \mathcal{N}_i] \approx 1.30R^2$ (by case (a)) into (6) solves $\mathbb{E}[A_{v\backslash i}|v \in \mathcal{N}_{ij}]$, which we denote by $M(r)$. Then we have

$$\mathbb{E}[K_{v\backslash i}|v \in \mathcal{N}_{ij}] = \frac{n}{R^2}\int_0^R M(r) f(r) dr \approx 1.19n.$$

(c) Proven by noticing A_{ij} is complementary to the area corresponding to case (a). □

These lemmas enable us to prove (see appendix) that

$$\begin{aligned} p_{oh} &\approx p_{ctrl}\ \exp[-1.30n(1 - p_{ni\text{-}oh})], \\ p_{succ} &\approx p_{oh} \exp[-1.30n(1 - p_{ni\text{-}cts})]. \end{aligned} \quad (7)$$

Now we solve λ_c (together with λ_{cts}). From the perspective of a transmitter, the average number of successful control channel handshakes that it initiates per second is $p_{ctrl}\lambda_{rts}p_{succ}$. Since each successful control channel handshake leads to transmitting one data packet, we have $p_{ctrl}\lambda_{rts}p_{succ} = \lambda$.

From the perspective of a receiver, it sends a `McCTS` when it successfully receives (overhears) a `McRTS` addressed to it, and hence $\lambda_{cts} = \lambda_{rts}\, p_{oh}$. Then combining these with $\lambda_c = \lambda_{rts} + \lambda_{cts}$ yields

$$\lambda_c = \frac{\lambda(1 + p_{oh})}{p_{ctrl}\, p_{succ}} \text{ and } \lambda_{cts} = \frac{\lambda\ p_{oh}}{p_{ctrl}\, p_{succ}} \quad (8)$$

where p_{oh} and p_{succ} are given in (7).

4.4 Solving Equation (1) and Target Metric p_{co}

Based on the proof of (4), it can be derived that

$$\Pr[\mathcal{O}(v \leftarrow y)|\mathcal{O}(v \leftarrow x)] \approx p_{ctrl}^{\star}\ p_{ni\text{-}oh}^{K_{v\backslash y}}, \quad (9)$$

$$\text{where} \quad p_{ctrl}^{\star} \triangleq \Pr[\mathcal{C}_v(s_y)|\mathcal{O}(v \leftarrow x)].$$

Note that $p_{ctrl}^{\star} \neq p_{ctrl}$, because s_y is not an *arbitrary* time for v due to the effect form $\mathcal{O}(v \leftarrow x)$. The reason is that $\mathcal{O}(v \leftarrow x)$ implies $\mathcal{C}_v(s_x)$, and thus for $\mathcal{C}_v(s_y)$ to happen, v must stay *continuously* on the control channel during $[s_x, s_y]$ (otherwise, a switching will lead to v staying on the data channel for T_d, but $s_x + T_d > s_y$ since x's data communication is still ongoing at s_y, and hence $\mathcal{C}_v(s_y)$ can never happen).

It can be proven (see appendix for the proof) that

$$p_{ctrl}^{\star} = \frac{(w\lambda_c - \frac{1-w}{T_d})\ g(\lambda_c + \lambda_w) + \frac{1-w}{T_d}\ g(\lambda_w)}{1 - w + (w\lambda_c - \frac{1-w}{T_d})\ g(\lambda_c)} \quad (10)$$

where

$$g(x) = \frac{1 - e^{-xT_d}}{x}, \quad w = \frac{p_{ctrl} - p_{oh}}{1 - p_{oh}}, \text{ and}$$
$$\lambda_w = \lambda_{rts}p_{succ} + \lambda_{cts}.$$

Combining (4) and (9) reduces (11) to

$$p_{co}^{xy}(v) \approx p_{ctrl}\ p_{ctrl}^{\star}\ p_{ni\text{-}oh}^{K_{v\backslash x} + K_{v\backslash y}},\ \forall v \in \mathcal{N}_{xy}. \quad (11)$$

Let $p_{co}^{xy}(\star)$ be the average of $p_{co}^{xy}(v)$ over all $v \in \mathcal{N}_{xy}$, i.e., $p_{co}^{xy}(\star)$ is the probability that an arbitrary node in $\mathcal{N}_{xy}$ is cooperative with respect to x and y, Using Lemma 2-(b),

$$p_{co}^{xy}(\star) \approx p_{ctrl}\ p_{ctrl}^{\star}\ \exp[-2.38n(1 - p_{ni\text{-}oh})]. \quad (12)$$

By the definition of p_{co}^{xy} in Table 1,

$$p_{co}^{xy} \approx 1 - \prod_{v \in \mathcal{N}_{xy}} [1 - p_{co}^{xy}(v)] \approx 1 - [1 - p_{co}^{xy}(\star)]^{K_{xy}}, \quad (13)$$

where the events corresponding to $1 - p_{co}^{xy}(v)$, i.e., nodes not being cooperative with respect to x and y, are regarded as independent of each other, as an approximation.

Thus p_{co} is determined by averaging p_{co}^{xy} over all (x, y) pairs that are possible to create MCC problems. It can be proven that these pairs are neighboring pairs (x, y) satisfying (d_i denoting the degree of a node i)

(a) $d_x \geq 2$, $d_y \geq 2$, but not $d_x = d_y = 2$, or

(b) $d_x = d_y = 2$, but x and y are not on the same three-cycle (triangle).

This condition is satisfied by all neighboring pairs in a *connected* random network, because the connectivity requires a sufficiently high node degree ($5.18 \log N$ where N is the total number of nodes [26]) which is much larger than 2. Therefore, taking expectation of (13) over all neighboring pairs using Lemma 1 and Lemma 2-(c),

$$\begin{aligned} p_{co} &= 1 - \exp[-p_{co}^{xy}(\star)\overline{K_{xy}}] \\ &\approx 1 - \exp[-1.84n\ p_{co}^{xy}(\star)]. \end{aligned} \quad (14)$$

This completes the analysis.

Procedure 1 PKT-ARRIVAL
Called when a data packet arrives

```
enqueue the packet, $L_{queue}$++
if $S_{ctrl} = FREE \wedge S_{node} = IDLE \wedge L_{queue} = 1$ then
    call ATTEMPT-RTS
end if
```

4.5 Special Case: Single-Hop Networks

Now that all nodes are in the communication range of each other, we have $p_{ni\text{-}oh} = p_{ni\text{-}cts} = 1$ according to Prop. 2 and 3, which leads to $p_{succ} = p_{oh} = p_{ctrl}$ according to (7), and $p_{co}^{xy}(v) = p_{ctrl}\, p_{ctrl}^{\star}$ according to (11). Hence (13) reduces to

$$p_{co}^{xy} = 1 - (1 - p_{ctrl}\, p_{ctrl}^{\star})^{K_{xy}},$$

where K_{xy} is the number of all possible cooperative nodes with respect to x and y, leading to $K_{xy} = n - 4$. So, as the average of p_{co}^{xy},

$$p_{co} = 1 - (1 - p_{ctrl}\, p_{ctrl}^{\star})^{n-4}, \tag{15}$$

where p_{ctrl} is given below, by solving the equations in Section 4.3,

$$p_{ctrl} = \frac{1}{2}(1 - \lambda T_d + \sqrt{1 + \lambda T_d(\lambda T_d - 6)}\ ),$$

and $p_{ctrl}^{\star}$ is given below, by reducing (10) with $w = 0$,

$$p_{ctrl}^{\star} = \frac{g(\lambda_w) - g(\lambda_c + \lambda_w)}{T_d - g(\lambda_c)}$$

$$\text{where} \quad \lambda_c = \frac{1}{2}\left(\frac{1 - \sqrt{1 + \lambda T_d(\lambda T_d - 6)}}{\lambda T_d^2} - \frac{3}{T_d}\right),$$

$$\lambda_w = \frac{1 - \sqrt{1 + \lambda T_d(\lambda T_d - 6)}}{T_d} - \lambda.$$

5. INVESTIGATING P_{CO} WITH DISH

We verify the analysis in both single-hop and multi-hop networks and identify key findings therein. We also investigate the correlation between p_{co} and network performance.

5.1 Protocol Design and Simulation Setup

5.1.1 Model-Based DISH

This is a multi-channel MAC protocol based on the protocol framework described in Section 3. Key part of its pseudocode is listed below, where S_{ctrl} is the control channel status (FREE/BUSY) detected by the node running the protocol, S_{node} is the node's state (IDLE/TX/RX, etc.), L_{queue} is the node's current queue length, and they are initialized as FREE, IDLE and 0, respectively. The frame format of McRTS and McCTS is shown in Fig. 4, where we can see that they carry channel usage information. A node that overhears McRTS or McCTS will cache the information in a *channel usage table* shown in Fig. 5, where Until is converted from Duration by adding the node's own clock.

FC	Tx	Rx	Ch	Duration	CRC
2	6	6	1	2	2

FC: frame control Tx: address of transmitter
Ch: channel index Rx: address of receiver

Figure 4: Frame format of McRTS and McCTS.

Tx	Rx	Ch	Until
T1	R1	1	00:15:36
T2	R2	3	00:16:01

Figure 5: Channel usage table.

Procedure 2 ATTEMPT-RTS
Called by PKT-ARRIVAL or CHECK-QUEUE

```
construct a set $\mathcal{F}$ of free channel indexes using channel usage table
if $\mathcal{F} \neq \phi$ then
    send McRTS with CH:=RANDOM($\mathcal{F}$)
else
    Timer ← min(until − $now$)
    while $S_{ctrl} = FREE \wedge$ Timer not expired do
        wait {carrier sensing remains on}
    end while
    if Timer expired then
        call CHECK-QUEUE
    else
        call PASSIVE {receive a control message}
    end if
end if
```

Procedure 3 CHECK-QUEUE
Called when $S_{ctrl} = FREE \wedge S_{node} = IDLE$ changes from $FALSE$ to $TRUE$

```
if $L_{queue} > 0$ then
    Timer ← RANDOM$(0, 10b)$ {FAMA [27,28]}
    while $S_{ctrl} = FREE \wedge$ Timer not expired do
        wait {carrier sensing remains on}
    end while
    if Timer expired then
        call ATTEMPT-RTS
    else
        call PASSIVE {receive a control message}
    end if
end if
```

As is based on the system model, this protocol does not use a concrete DISH mechanism, i.e., cooperation is treated as a resource while not actually utilized.

5.1.2 Ideal DISH

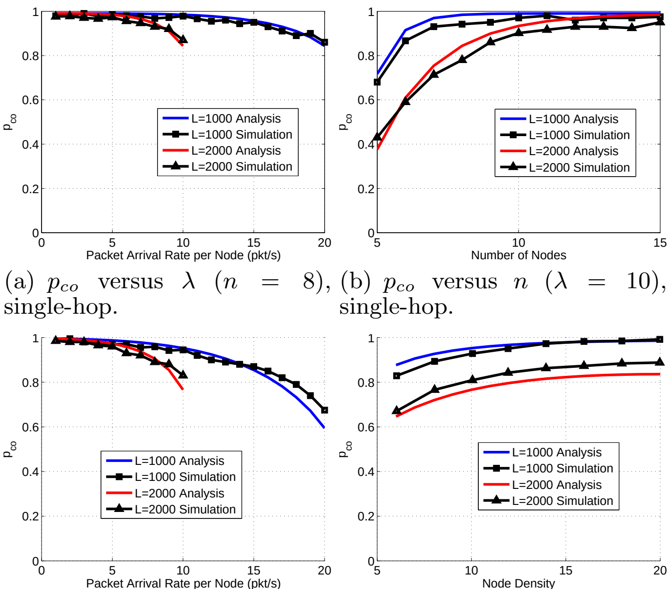
This protocol is by adding an ideal cooperating mechanism to the model-based DISH. Each time when an MCC problem is created by nodes x and y and if at least one cooperative node is available, the node that is on the control channel, i.e., node y, will be informed without any message physically sent, and then back off to avoid the MCC problem.

5.1.3 Real DISH

In this protocol, cooperative nodes will physically send cooperative messages to inform a transmitter or receiver of the MCC problem so that it will backoff. We design this real DISH by adapting CAM-MAC [11]. The only change that we made is that, since in CAM-MAC a transmitter will send a PRA and a CFA, and a receiver will send a PRB and a CFB, during the control channel handshake, we change these control packet sizes such that $\|PRA\| + \|CFA\| = \|McRTS\| = \|PRB\| + \|CFB\| = \|McCTS\|$, where $\|\cdot\|$ gives the size of a packet.

5.1.4 Simulation Setup

There are six channels of data rate 1Mbps each. Data packets arrive at each node as a Poisson process. The uniform traffic pattern as in the model is used. Traffic load λ (pkt/s), node density n $(1/R^2)$, and packet size L (byte) will vary in simulations. In multi-hop networks, the network area is 1500m×1500m and the transmission range is 250m. Each simulation is terminated when a total of 100,000 data packets



(a) p_{co} versus λ ($n = 8$), single-hop.

(b) p_{co} versus n ($\lambda = 10$), single-hop.

(c) p_{co} versus λ ($n = 10$), multi-hop.

(d) p_{co} versus n ($\lambda = 10$), multi-hop.

Figure 6: Impact of traffic load and node density, with different packet sizes. The value ranges of X axes are chosen such that the network is stable.

are sent over the network, and each set of results is averaged over 15 randomly generated networks.

5.2 Investigation with Model-Based DISH

The p_{co} obtained via analysis and simulations are compared in Fig. 6. We see a close match between them, with a deviation of less than 5% in almost all single-hop scenarios, and less than 10% in almost all multi-hop scenarios. Particularly, the availability of cooperation is observed to be at a high level ($p_{co} > 0.7$ in most cases), which suggests that a large percentage of MCC problems would be avoided by exploiting DISH, and DISH is feasible to use in multi-channel MAC protocols. (**Finding 1**)

Specifically, Fig. 6(a) and Fig. 6(c) consistently show that, in both single-hop and multi-hop networks, p_{co} monotonically decreases as λ increases. The reasons are two folds. First, as traffic grows, each node spends more time on data channels for data transmission and reception, which reduces p_{ctrl} and hence the chance of overhearing control messages (p_{oh}), resulting in lower p_{co}. Second, as the control channel is the rendezvous to set up all communications, larger traffic intensifies the contention and introduces more interference to the control channel, which is hostile to messages overhearing and thus also reduces p_{co}.

Fig. 6(b) and Fig. 6(d) show that p_{co} monotonically increases as n increases, and is concave. The increase of p_{co} is because MCC problems are more likely to have cooperative nodes under a larger node population, while the deceleration of the increase is because more nodes also generate more interference to the control channel.

An important message conveyed by this observation is that, although a larger node density creates more MCC problems (e.g., more channel conflicts as data channels are more likely to be busy), it also boosts the availability of cooperation which avoids more MCC problems. This implies that the performance degradation can be mitigated. (**Finding 2**)

In both single-hop and multi-hop networks, a larger packet size L corresponds to a lower p_{co}. However, note that this is observed under the same *packet* arrival rate (pkt/s), which means actually a larger *bit* arrival rate for a larger L, and can be explained by the previous scenarios of p_{co} versus λ. Now if we consider the same *bit* arrival rate, by examining the two analysis curves in Fig. 6(c) where we compare p_{co} with respect to the same $\lambda \cdot L$ product, e.g., ($\lambda = 5, L = 2000$) versus ($\lambda = 10, L = 1000$), and ($\lambda = 10, L = 2000$) versus ($\lambda = 20, L = 1000$), then we will see that a larger L corresponds to a *higher* p_{co}, which is contrary to the observation under the same *packet* arrival rate. The explanation is that, for a given bit arrival rate, increasing L reduces the number of packets and hence *fewer control channel handshakes* are required, thereby alleviating control channel interference. (**Finding 3**)

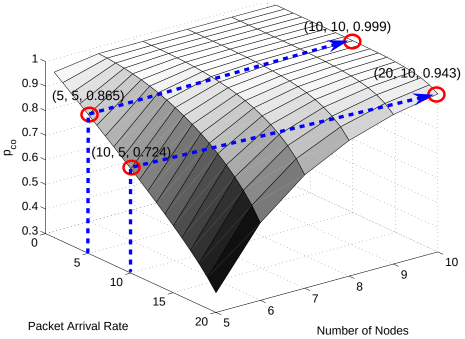


Figure 7: p_{co} versus λ and n. Each of the two arrows indicates a multiplicative increase of λ and n with the same factor (two).

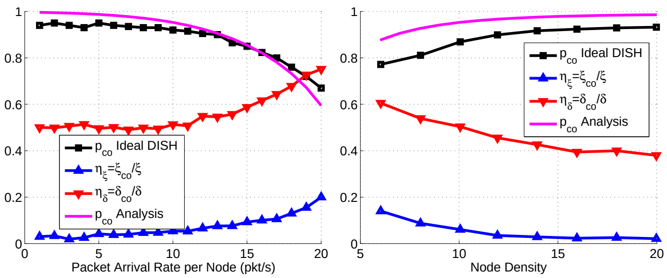
The above results indicate that node density and traffic load affect the availability of cooperation in *opposite* ways. This section aims to find which one dominates over the other. In Fig. 7, we plot the relationship of p_{co} versus λ and n, given $L = 1000$ and based on the *analytical* result for single-hop networks. We multiplicatively increase λ and n with the same factor (two), and find that, when increasing (λ, n) from (5,5) to (10,10), p_{co} keeps *increasing* from 0.865 to 0.999, and when increasing (λ,n) from (10,5) to (20,10), p_{co} keeps *increasing* from 0.724 to 0.943.

This investigation shows that n is the dominating factor over λ that determines the variation of p_{co}. This implies that DISH networks should have better scalability than non-DISH networks, since p_{co} increases when both traffic load and node density scale up. (**Finding 4**)

5.3 Investigation with Ideal DISH

The results of comparison are shown in Fig. 8, where p_{co} with ideal DISH well matches p_{co} of analysis. This confirms **Findings 1-3**, and we speculate the reasons to be as follows. With ideal DISH, a transmitter may be informed of a deaf terminal problem and thus will backoff for a fairly long time, which leads to *fewer* `McRTS` being sent. On the other hand, a node may also be informed of a channel conflict problem and thus will re-select channel and retry shortly, which leads to *more* control messages being sent. Empirically, the latter case has more significant effect, which means that, overall, there will be an *increase* of control messages being sent. This boosts interference and thus would reduce p_{co}. However, nodes will also stay *longer* on the control channel due to less use of conflicting data channels, which would elevate p_{co}. Consequently, p_{co} does not change noticeably.

We investigate how p_{co} correlates to network performance



(a) Impact of λ ($n = 10$).   (b) Impact of n ($\lambda = 10$).

Figure 8: Investigating p_{co} with ideal DISH in stable networks. This includes (i) verification of analysis, and (ii) correlation between p_{co} and (η_ξ, η_δ) (ratio of data collision, ratio of packet delay). $L = 1000$ byte.

— specifically, data channel collision rate ξ, packet delay δ, and aggregate throughput S. We consider both stable networks and saturated networks under multi-hop scenarios.

In stable networks, we measure (ξ, δ) and (ξ_{co}, δ_{co}) when without and with cooperation (ideal DISH), respectively. Then we compute $\eta_\xi = \xi_{co}/\xi$ and $\eta_\delta = \delta_{co}/\delta$ to compare to p_{co} with ideal DISH. The first set of results, by varying traffic load λ, is shown in Fig. 8(a). We observe that the two *ascending* and *convex* curves of η_ξ and η_δ approximately *reflect* the *descending* and *concave* curve of p_{co}, which hints at a *linear* or *near-linear* relationship between p_{co} and these two performance ratios. That is, $\eta_\xi + p_{co} \approx c_1$, $\eta_\delta + p_{co} \approx c_2$, where c_1 and c_2 are two constants. The second set of results, by varying node density n, is shown in Fig. 8(b). On the one hand, η_ξ and η_δ decreases as n increases, which is contrary to Fig. 8(a). This confirms our earlier observations: n is amicable whereas λ is hostile to p_{co} (the smaller η_ξ and η_δ, the better performance cooperation offers). On the other hand, the correlation between p_{co} and the performance ratios is found again: as p_{co} increases on a concave curve, it is reflected by η_ξ and η_δ which decrease on two convex curves.

In saturated networks, we vary node density n and measure aggregate throughput without and with cooperation (ideal DISH), as S and S_{co}, respectively. Then we compute $\eta_S = S/S_{co}$ (note that this definition is inverse to η_ξ and η_δ, such that $\eta_S \in [0, 1]$) to compare to p_{co} with ideal DISH. The results are summarized in Fig. 9. We see that (i) p_{co} grows with n, which conforms to Finding 2, and particularly, (ii) the declining and convex curve of η_S reflects the rising and concave curve of p_{co}, which is consistent with the observation in stable networks. In addition, here p_{co} is lower than the p_{co} in stable networks. This is explained by our earlier result that higher traffic load suppresses p_{co}.

In summary, the experiments in stable networks and saturated networks both demonstrate a strong correlation (*linear* or *near-linear* mapping) between p_{co} and network performance ratio in terms of typical performance metrics. This may significantly simplify performance analysis for cooperative networks via bridging the *nonlinear* gap between network parameters and p_{co}, and also suggests that p_{co} be used as an appropriate performance indicator itself. (**Finding 5**)

The explanation to this linear or near-linear relationship should involve intricate network dynamics. We speculate that the rationale might be that (i) MCC problems are an essential performance *bottleneck* to multi-channel MAC performance, and (ii) p_{co} is equivalent to the *ratio* of MCC problems that can be avoided by DISH. In any case, we reckon that this observation may spur further studies and lead to more thought-provoking results.

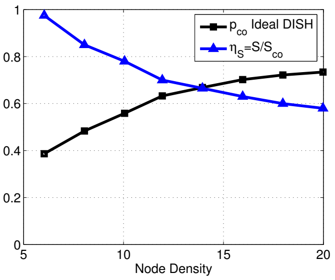


Figure 9: Investigating p_{co} with ideal DISH in saturated networks: correlation between p_{co} and η_S (throughput ratio). $L = 1000$ byte.

5.4 Investigation with Real DISH

Due to space constraints, we discuss the results without presenting figures because the results were found to be similar to those with ideal DISH.

The deviation between simulations and analysis, which maximally reached ∼15%, was found to be slightly larger than that with model-based or ideal DISH. However, the overall trends still match, and **Findings 1-4** are confirmed. For explanation, we speculate that, in real DISH, there are much more control messages (including cooperative messages) being sent, which result in more control channel interference and thus would diminish p_{co}. But on the other hand, nodes stay longer on the control channel due to the same reason as for the case of ideal DISH (i.e., less use of conflicting data channels), which would increase p_{co}. As a result, p_{co} with real DISH does not deviate significantly from the analysis.

The near-linear relationship between p_{co} and $(\eta_\xi, \eta_\delta, \eta_S)$ was observed. This confirms **Finding 5**. As a possible explanation, the remark following Finding 5 in Section 5.3 applies. Although the absolute values of these quantities were found to differ from those with ideal DISH (ranging between 3-18%), this is not the primary concern.

6. CONCLUSION

Distributed Information SHaring (DISH) represents a mechanism different from data relaying to exploit cooperative diversity. This paper gives the first theoretical treatment of this new notion of cooperation, by addressing the availability of cooperation via a metric p_{co}. Instead of directly analyzing throughput which is an open problem in general and rendered much more complicated when considered together with multiple channels and DISH, our approach is to analyze p_{co} first and then correlate p_{co} with performance metrics including throughput. We conduct analysis in a multi-hop multi-channel wireless network and, to verify its validity and study its implications, investigate p_{co} with three different contexts of DISH: model-based DISH, ideal DISH, and real DISH. The investigation validates that our analysis accurately captures the interaction among network parameters, which allows us to draw important findings of p_{co} with respect to network dynamics. It also reveals a near-linear relationship between p_{co} and network performance, which may greatly aid in performance analysis for cooperative networks and also suggests p_{co} to be a proper performance indicator itself.

This work is the first that explicitly presents DISH, together with a detailed study offering meaningful insights into understanding cooperation. Based on our findings, we conclude that p_{co} is a useful metric capable of characterizing the performance of *DISH networks*. We contend that DISH is useful and practical enough to be a part of future cooperative communication networks.

7. REFERENCES

[1] E. C. van der Meulen, “Three-terminal communication channels,” *Advances in Applied Probability*, vol. 3, no. 1, pp. 120–154, 1971.

[2] T. M. Cover and A. E. Gamal, “Capacity theorems for the relay channel,” *IEEE Trans. Infomation Theory*, vol. 25, pp. 572–84, Sept. 1979.

[3] A. Host-Madsen, “Capacity bounds for cooperative diversity,” *IEEE Trans. on Information Theory*, vol. 52, no. 4, pp. 1522–1544, Apr. 2006.

[4] J. N. Laneman and G. W. Wornell, “Distributed space-time-coded protocols for exploiting cooperative diversity in wireless networks,” *IEEE Transactions on Information Theory*, vol. 49, no. 10, October 2003.

[5] A. Sendonaris, E. Erkip, and B. Aazhang, “User cooperation diversity–part I: System description,” *IEEE Trans. Commun.*, vol. 51, no. 11, December 2003.

[6] ——, “User cooperation diversity–part II: Implementation aspects and performance analysis,” *IEEE Trans. Commun.*, vol. 51, no. 11, Dec. 2003.

[7] H. Zhu and G. Cao, “rDCF: A relay-enabled medium access control protocol for wireless ad hoc networks,” in *IEEE Infocom*, 2005.

[8] A. Azgin, Y. Altunbasak, and G. AlRegib, “Cooperative MAC and routing protocols for wireless ad hoc networks,” in *IEEE GLOBECOM*, 2005.

[9] S. Moh, C. Yu, S.-M. Park, and H.-N. Kim, “CD-MAC: Cooperative diversity MAC for robust communication in wireless ad hoc networks,” in *IEEE ICC*, June 2007, pp. 3636–3641.

[10] P. Liu, Z. Tao, S. Narayanan, T. Korakis, and S. S. Panwar, “CoopMAC: A cooperative MAC for wireless LANs,” *IEEE Journal On Selected Areas In Communications*, vol. 25, no. 2, pp. 340–354, 2007.

[11] T. Luo, M. Motani, and V. Srinivasan, “CAM-MAC: A cooperative asynchronous multi-channel MAC protocol for ad hoc networks,” in *IEEE Broadnets*, San Jose, CA, USA, October 2006.

[12] ——, “Altruistic cooperation for energy-efficient multi-channel MAC protocols,” in *ACM MobiCom*, Montreal, QC, Canada, 2007, extended abstract.

[13] S.-L. Wu, C.-Y. Lin, Y.-C. Tseng, and J.-P. Sheu, “A new multi-channel MAC protocol with on-demand channel assignment for multi-hop mobile ad hoc networks,” in *I-SPAN*, 2000.

[14] A. Nasipuri, J. Zhuang, and S. R. Das, “A multichannel CSMA MAC protocol for multihop wireless networks,” in *WCNC*, 1999.

[15] N. Jain, S. R. Das, and A. Nasipuri, “A multichannel CSMA MAC protocol with receiver-based channel selection for multihop wireless networks,” in *IEEE ICCCN*, 2001.

[16] A. Adya, P. Bahl, J. Padhye, and A. Wolman, “A multi-radio unification protocol for IEEE 802.11 wireless networks,” in *IEEE Broadnets*, 2004.

[17] R. Maheshwari, H. Gupta, and S. R. Das, “Multichannel MAC protocols for wireless networks,” in *IEEE SECON*, 2006.

[18] J. Chen, S. Sheu, and C. Yang, “A new multichannel access protocol for IEEE 802.11 ad hoc wireless LANs,” in *PIMRC*, 2003.

[19] J. So and N. Vaidya, “Multi-channel mac for ad hoc networks: Handling multi-channel hidden terminals using a single transceiver,” in *ACM MobiHoc*, 2004.

[20] J. Zhang, G. Zhou, C. Huang, S. H. Son, and J. A. Stankovic, “TMMAC: an energy efficient multi-channel MAC protocol for ad hoc networks,” in *IEEE ICC*, 2007.

[21] A. Tzamaloukas and J. Garcia-Luna-Aceves, “Channel-hopping multiple access,” in *IEEE ICC*, 2000.

[22] ——, “Channel-hopping multiple access with packet trains for ad hoc networks,” in *IEEE Device Multimedia Communications*, 2000.

[23] P. Bahl, R. Chandra, and J. Dunagan, “SSCH: Slotted seeded channel hopping for capacity improvement in IEEE 802.11 ad-hoc wireless networks,” in *ACM MobiCom*, 2004.

[24] Y. S. Han, J. Deng, and Z. J. Haas, “Analyzing multi-channel medium access control schemes with ALOHA reservation,” *IEEE Trans. on Wireless Comm.*, 2005.

[25] E. W. Weisstein, “Circular segment—from mathworld,” http://mathworld.wolfram.com/CircularSegment.html.

[26] F. Xue and P. R. Kumar, “The number of neighbors needed for connectivity of wireless networks,” *Wireless Networks*, vol. 10, no. 2, pp. 169–181, 2004.

[27] C. L. Fullmer and J. J. Garcia-Luna-Aceves, “Floor acquisition multiple access (FAMA) for packet-radio networks,” in *SIGCOMM*, New York, NY, USA, 1995.

[28] J. J. Garcia-Luna-Aceves and C. L. Fullmer, “Floor acquisition multiple access (FAMA) in single-channel wireless networks,” *Mobile Networks and Applications*, vol. 4, no. 3, pp. 157–174, 1999.

APPENDIX

A. PROOFS AND DERIVATIONS

A.1 Proof of Prop. 2

PROOF. In the case of $u \in \mathcal{N}_{vi}$, no matter u is on the control channel at s_i, or is on a data channel at s_i but switches to the control channel before $s_i + b$, it will sense a busy control channel (due to CSMA) and thus keep silent.

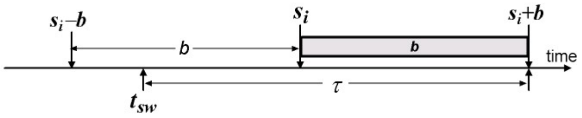


Figure 10: The vulnerable period of v is $[s_i - b, s_i + b]$, in which node $u \in \mathcal{N}_{v\setminus i}$ should not start transmission on the control channel.

In the case of $u \in \mathcal{N}_{v\setminus i}$, see Fig. 10. Note that the vulnerable period of v is $[s_i - b, s_i + b]$ instead of $[s_i, s_i + b]$, because a transmission started within $[s_i - b, s_i]$ will end within $[s_i, s_i + b]$. Therefore, by the total probability theorem,

$$\begin{aligned} p_{ni\text{-}oh} &= \Pr[\mathcal{C}_u(s_i - b)] \cdot \Pr[\mathcal{I}_u(s_i - b, s_i + b) | \mathcal{C}_u(s_i - b)] \\ &\quad + \Pr[\overline{\mathcal{C}_u(s_i - b)}] \cdot \Pr[\mathcal{I}_u(s_i - b, s_i + b) | \overline{\mathcal{C}_u(s_i - b)}] \\ &= p_{ctrl} \cdot e^{-2\lambda_c b} + (1 - p_{ctrl}) \cdot (1 - \frac{2b}{T_d} + \frac{1 - e^{-2\lambda_c b}}{\lambda_c T_d}), \end{aligned}$$

where $\Pr[\mathcal{I}_u(s_i - b, s_i + b) | \overline{\mathcal{C}_u(s_i - b)}]$ is solved by Prop. 1. □

A.2 Proof of Prop. 3

PROOF. The case of $u \in \mathcal{N}_{ij}$ follows the same line as the proof for Prop. 2. In the case of $u \in \mathcal{N}_{i\setminus j}$, the only difference from Prop. 2 is that now we are implicitly given the fact that

i was transmitting McRTS during $[s_j - b, s_j]$. This excludes i's any neighbor u interfering in $[s_j - b, s_j]$. Therefore i's vulnerable period is $[s_j, s_j + b]$ instead of $[s_j - b, s_j + b]$ as compared to Prop. 2. So

$$\begin{aligned} p_{ni\text{-}cts} =& \Pr[\mathcal{C}_u(s_j - b)] \cdot \Pr[\mathcal{I}_u(s_j, s_j + b) | \mathcal{C}_u(s_j - b)] \\ &+ \Pr[\overline{\mathcal{C}_u(s_j - b)}] \cdot \Pr[\mathcal{I}_u(s_j, s_j + b) | \overline{\mathcal{C}_u(s_j - b)}]. \end{aligned}$$

Note that we condition on $\mathcal{C}_u(s_j - b)$ instead of $\mathcal{C}_u(s_j)$, because s_j is not an *arbitrary* time due to i's McRTS transmission during $[s_j - b, s_j]$, which leads to $\Pr[\mathcal{C}_u(s_j)] \neq p_{ctrl}$.

First, $\Pr[\mathcal{I}_u(s_j, s_j + b) | \mathcal{C}_u(s_j - b)] = 1$. This is because, as $\mathcal{C}_u(s_j - b) \Leftrightarrow \mathcal{S}_u(s_j - b, s_j)$ which is easy to show, u will successfully overhear i's McRTS, and hence will keep silent in the next period of b to avoid interfering with i receiving McCTS.

Next consider $\Pr[\mathcal{I}_u(s_j, s_j + b) | \overline{\mathcal{C}_u(s_j - b)}]$ where u is on a data channel at $s_j - b$. If u switches to the control channel (i) before s_j, it will be suppressed by i's McRTS transmission until s_j, and thus the vulnerable period of i receiving McCTS is $[s_j, s_j + b]$, (ii) within $[s_j, s_j + b]$, this has been solved by Prop. 1, or (iii) after $s_j + b$, the probability to solve is obviously 1. Therefore,

$$\begin{aligned} &\Pr[\mathcal{I}_u(s_j, s_j + b) | \overline{\mathcal{C}_u(s_j - b)}] = \Pr[\Omega_u(s_j - b, s_j)]\ e^{-\lambda_c b} \\ &\quad + \Pr[\Omega_u(s_j, s_j + b)]\,(1 - \frac{b}{T_d} + \frac{1 - e^{-\lambda_c b}}{\lambda_c T_d}) \\ &\quad + \{1 - \Pr[\Omega_u(s_j - b, s_j)] - \Pr[\Omega_u(s_j, s_j + b)]\} \times 1. \end{aligned}$$

According to (3), $\Pr[\Omega_u(s_j - b, s_j)] = \Pr[\Omega_u(s_j, s_j + b)] = b/T_d$. Then by substitution the proposition is proven. □

A.3 Derivation of Equation (4)

Proof. Based on the proof for the case $u \in \mathcal{N}_{vi}$ in Prop. 2, it is easy to show that $\mathcal{S}_v(s_i, s_i + b) \Leftrightarrow \mathcal{C}_v(s_i)$. Hence

$$\Pr[\mathcal{S}_v(s_i, s_i + b)] = \Pr[\mathcal{C}_v(s_i)] = p_{ctrl}.$$

Treating events $\mathcal{S}_v(s_i, s_i + b)$ (node v is silent on the control channel) and $\mathcal{I}_u(s_i, s_i + b)$ (node u does not interfere the control channel) being independent of each other, as an approximation, we have

$$\Pr[\mathcal{O}(v \leftarrow i)] \approx p_{ctrl} \prod_{u \in \mathcal{N}_{v \setminus i}} p_{ni\text{-}oh} = p_{ctrl}\ p_{ni\text{-}oh}^{K_{v \setminus i}}.$$

□

A.4 Derivation of Equation (7)

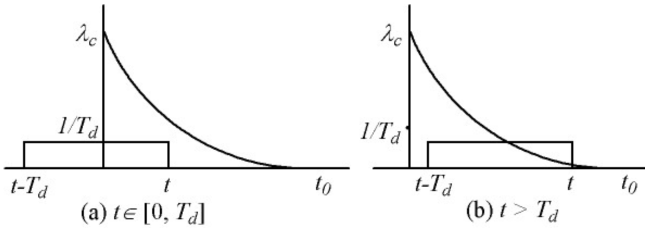
Proof. Taking the expectation of $\Pr[\mathcal{O}(v \leftarrow i)]$ (given by (4)) over all neighboring (v, i) pairs using Lemma 1 and Lemma 2-(a):

$$p_{oh} \approx p_{ctrl}\ \mathbb{E}[p_{ni\text{-}oh}^{K_{v \setminus i}}] \approx p_{ctrl}\ \exp[-1.30n(1 - p_{ni\text{-}oh})].$$

To solve for p_{succ}, notice that for a control channel handshake to be successful, (i) the McRTS must be successfully received by the receiver, with probability p_{oh}, and (ii) the McCTS must be successfully received by the transmitter, with probability $\mathbb{E}[p_{ni\text{-}cts}^{K_{i \setminus j}}]$ based on Prop. 3 (assuming that $p_{ni\text{-}cts}$ holds for nodes in $\mathcal{N}_{i \setminus j}$ independently, as an approximation). Therefore,

$$p_{succ} \approx p_{oh} \mathbb{E}[p_{ni\text{-}cts}^{K_{i \setminus j}}] \approx p_{oh} \exp[-1.30n(1 - p_{ni\text{-}cts})].$$

□



(a) $t \in [0, T_d]$ (b) $t > T_d$

Figure 11: The convolution of $\frac{1}{T_d}$ $(t \in [0, T_d])$ and $\lambda_c e^{-\lambda_c t}$ $(t > 0)$.

A.5 Derivation of Equation (10)

Proof. Recall that node v must stay continuously on the control channel during $[s_x, s_y]$. Let $\tau_c = s_y - s_x$ and suppose v switches to a data channel at $s_x + \tau_w$, then we need $\tau_w > \tau_c$. Hence $p_{ctrl}^{\star} = \Pr(\tau_w > \tau_c)$, where $\tau_c \in [0, T_d]$.

Denote by $f_{\tau_c}(t)$ the pdf of an unbounded τ_c ($s_y \in (s_x, \infty)$). The fact that a MCC problem is created by x and y (at s_y) implies that y missed x's control message (at s_x). This is due to one of the following: (i) y is on the control channel at s_x but interfered, in which case $f_{\tau_c}(t)$ is $\lambda_c e^{-\lambda_c t}$ (ignoring the short interference period which is in the magnitude of b, while τ_c is in the magnitude of T_d), (ii) y is on a data channel at s_x, in which case y must switch to the control channel before s_y. Again see Fig. 2, where t_1 and t_2 are now s_x and s_y, respectively. Let $\tau_1 = t_{sw} - s_x$ and $\tau_2 = s_y - t_{sw}$, then $\tau_c = \tau_1 + \tau_2$. Note that τ_1 is uniformly distributed in $[0, T_d]$, τ_2 is exponentially distributed with the mean of $1/\lambda_c$, and τ_1 and τ_2 can be regarded as independent. Therefore, $f_{\tau_c}(t)$ is the convolution of $\frac{1}{T_d}$ $(t \in [0, T_d])$ and $\lambda_c e^{-\lambda_c t}$ $(t > 0)$, which can be calculated by referring to Fig. 11, to be $f_{\tau_c}^d(t) =$

$$\frac{1 - e^{-\lambda_c t}}{T_d}[u(t) - u(t - T_d)] + \frac{e^{-\lambda_c t}}{T_d}(e^{\lambda_c T_d} - 1)u(t - T_d).$$

where $u(\cdot)$ is the unit step function.

A weighted sum of the above cases (i) and (ii) gives

$$f_{\tau_c}(t) = w\ \lambda_c e^{-\lambda_c t} + (1 - w)\ f_{\tau_c}^d(t)$$

where w is the weight for case (i). To determine w, note that the probability of case (ii) is $1 - p_{ctrl}$, and the probability of case (i) is $p_{ctrl}(1 - p_{ni\text{-}oh}^{K_{y \setminus x}})$ (using (4)) whose mean is $p_{ctrl}(1 - \exp[-1.30n(1 - p_{ni\text{-}oh})])$. Therefore

$$w = \frac{p_{ctrl}[1 - e^{-1.30n(1 - p_{ni\text{-}oh})}]}{p_{ctrl}[1 - e^{-1.30n(1 - p_{ni\text{-}oh})}] + (1 - p_{ctrl})} = \frac{p_{ctrl} - p_{oh}}{1 - p_{oh}}.$$

Finally we compute $p_{ctrl}^{\star} = \Pr(\tau_w > \tau_c)$ using $f_{\tau_c}(t)$. Recall that $f_{\tau_c}(t)$ is the pdf of an unbounded τ_c but τ_c is in fact bounded within $[0, T_d]$, therefore its actual pdf is $f_{\tau_c}(t) / \int_0^{T_d} f_{\tau_c}(t)dt$. Assuming that τ_w is exponentially distributed with mean $1/\lambda_w$, we have

$$p_{ctrl}^{\star} = \mathbb{E}_{\tau_c \in [0, T_d]}\ \Pr(\tau_w > \tau_c) = \int_0^{T_d} e^{-\lambda_w t} \frac{f_{\tau_c}(t)}{\int_0^{T_d} f_{\tau_c}(t)dt} dt$$

which reduces to (10). For λ_w, noticing that it is the average rate of a node on the control channel switching to data channels, which happens when a node successfully initiates a control channel handshake via McRTS or sends a McCTS, we have $\lambda_w = \lambda_{rts} p_{succ} + \lambda_{cts}$. □