The κ Factor: Inferring Protocol Performance Using Inter-link Reception Correlation

Kannan Srinivasan†, Mayank Jain†, Jung Il Choi†, Tahir Azim†,
Edward S Kim*, Philip Levis†, and Bhaskar Krishnamachari*
{srikank, mayjain, jungilchoi, tazim}@stanford.edu,
e.skim@usc.edu, pal@cs.stanford.edu, bkrishna@usc.edu
†Stanford University *University of Southern California
Stanford, CA-94305 Los Angeles, CA-90089

ABSTRACT

This paper explores metrics that capture to what degree packet reception on different links is correlated. Specifically, it explores metrics that shed light on when and why opportunistic routing and network coding protocols perform well (or badly). It presents a new metric, κ that, unlike existing widely used metrics, has no bias based on the packet reception ratios of links. This lack of bias makes κ a better predictor of performance of opportunistic routing and network coding protocols. Comparing Deluge and Rateless Deluge, Deluge's network coding counterpart, we find that κ can predict which of the two is best suited for a given environment. For example, irrespective of the packet reception ratios of the links, if the average κ of the link pairs is very high (close to 1.0), then using a protocol that does not code works better than using a network coding protocol.

Measuring κ on several 802.15.4 and 802.11 testbeds, we find that it varies significantly across network topologies and link layers. κ can be a metric for quantifying what kind of a network is present and help decide which protocols to use for that network.

Categories and Subject Descriptors: C.2.1 [Network Architecture and Design]:Wireless communications

General Terms: Measurement, Design, Performance.

Keywords: 802.15.4, Wireless measurement study, Low power wireless networks, Wireless protocol design.

1. INTRODUCTION

One of the principal opportunities that wireless provides is the ability to send a single packet to multiple receivers. Data dissemination protocols such as Deluge [8] and broadcast protocols such as RBP [17] can broadcast data to multiple neighbors at once. Routing protocols can snoop, or use opportunistic receptions [3], in order to forward packets even when delivery to the primary intended recipient fails.

In these schemes, correlations in packet reception on different links can greatly impact protocol performance. For example, if all links receive the same packets (i.e their reception is completely correlated,) then opportunistic routing protocols will not work better than the shortest path protocols, as there is no spatial diversity to exploit. At the other extreme, if the links are negatively correlated – a reception at one implies a failure at the other – then opportunistic routing can be of great benefit.

In prior work, researchers have often concluded that packet reception on different links is largely independent [16, 15]. Miu et al. use the cross conditional measure, $P(A = 0|B = 0) - P(A = 0)$, as a metric for inter-link correlation between links A and B [15]. We refer to this metric as χ. χ tries to capture correlation in link losses: if the losses on links A and B are independent then it's zero. This metric was subsequently used by Reis et al. [16] and Laufer et al. [12] to conclude that most of the link pairs have independent packet reception. Section 3 discusses properties desired of a correlation metric and shows that χ does not satisfy them: its value depends highly on the packet reception ratios (PRRs) of the two links. Section 5 further shows that χ is not indicative of how well opportunistic routing protocols work.

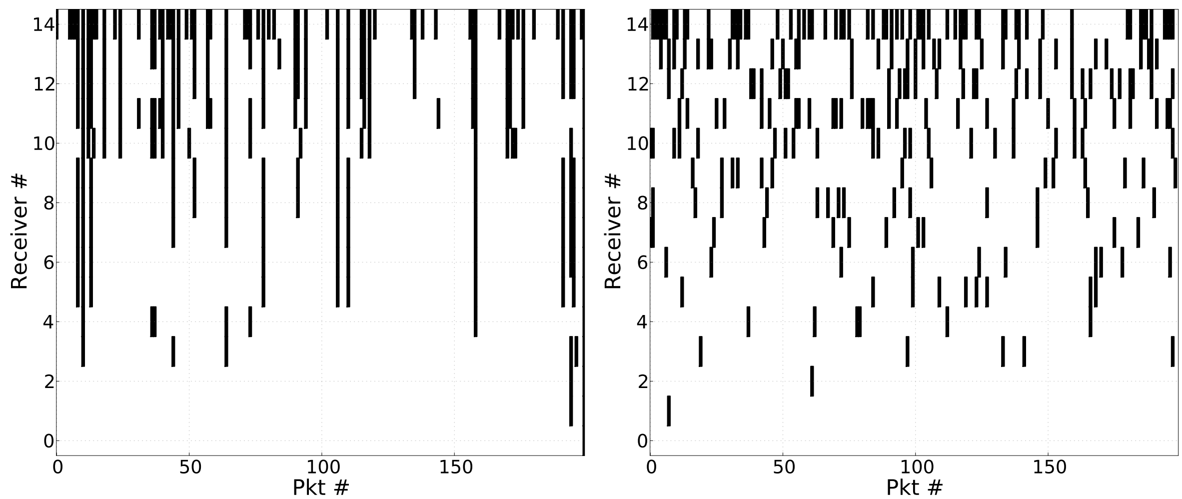
Protocol designs typically assume that reception on different links is independent [5, 7, 21, 12, 4, 10]. Network simulators [2, 13] and work on network analysis also typically ignore these correlations as it is commonly believed that these correlations are too hard to capture.

From a set of wireless measurements over the past 4 years, we have found that reception on different links is not always independent. This observation is not new; one of the earliest sensor network deployment studies, Great Duck Island, observed correlated reception [18]. However, the degree of correlation varies greatly across network setups and link layers.

This paper presents a new metric called κ that captures this degree of correlation. κ is a 3-tuple metric that measures packet reception correlation on two links that have a common transmitter. A κ of 1 means that the receptions on the two links are highly correlated, zero means they are independent, and -1 means that the losses on one are highly correlated with successes on the other. Unlike the existing metrics, the range of values that κ can take is not a function of PRRs of the links. This bias in other metrics makes them poor estimators of protocol performance. Section 3 discusses the κ metric in detail.

This paper shows that measuring correlation using κ can help us understand performance of protocols that exploit the broadcast nature of wireless to route or forward packets. Such protocols include opportunistic and network coding protocols. This paper shows that κ can predict opportunistic protocol performance. It shows how κ can be used to understand *when* a network coding protocol, such as Rateless Deluge [7], is beneficial over protocols that don't do net-

MobiCom'10, September 20–24, 2010, Chicago, Illinois, USA.



(a) Empirical Trace (b) Synthetic Independent Trace

Figure 1: Packet reception at multiple receivers when a single transmitter is broadcasting packets; (a) empirical trace from Mirage testbed on channel 16, and (b) independent synthetic trace with same PRR. Packet losses are marked by black overlines. The empirical trace has many more packet losses aligned across different receivers than the independent trace: empirical links (visually) show many correlated losses.

work coding. This allows researchers to make informed decisions about when to use network coding protocols for their testbeds and deployments. Section 6 describes this comparison in detail. Sections 4 and 5 systematically explore the usefulness of κ in understanding opportunistic routing protocol performance.

Measuring κ on IEEE 802.15.4 [20] and 802.11 (WiFi) [1] networks, we find that different networks see different degrees of correlated link pairs. In one network nearly 70% of the link pairs have a κ higher than 0.8, while in another network less than 20% of the link pairs fall in that range. This paper discusses possible causes of these correlations. In general, high power external interference causes packet losses at multiple receivers and increases correlation. Movement of an obstacle such that it blocks line-of-sight to only one receiver at a time reduces correlations. Section 7 investigates and presents such factors that affect correlation.

Overall, this paper makes four research contributions. First, it outlines desirable properties for an inter-link reception correlation metric that can predict protocol performance. It shows that a widely used metric does not satisfy these properties and is unsuitable for predicting protocol performance. Second, it presents a new metric called κ that addresses the shortcomings of existing metrics. Third, it shows that κ explains how well an opportunistic protocol like ExOR performs in a network. Fourth, it shows how κ can be used to choose between network coding and no-network-coding protocols for a network. Specifically, κ helps to choose the best between Deluge and Rateless Deluge for a wireless network.

This paper shows that reception on different links can be correlated and that measuring this correlation can help us understand when and why certain protocols perform the way they do. We believe that this insight is useful in designing efficient future protocols.

2. TESTBEDS

In an attempt to observe and understand various degrees of correlation present in wireless networks, this paper uses measurements and experiments from both 802.15.4 [20] and 802.11b [1] networks. 802.15.4 is an IEEE PHY-MAC low power, low data rate network standard with a 16 channel spectrum that overlaps the spectrum of 802.11b. It provides a data rate of 250 kbps and maximum transmit power of 0dBm, which are much lower than 802.11b's 11 Mbps capability and 23dBm maximum transmit power.

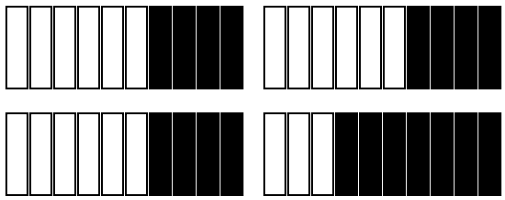
We run 802.15.4 experiments using TinyOS running on the Intel Mirage testbed [9], which consists of 100 Micaz [19] nodes placed along the ceiling. An Ethernet back-channel provides communication to all the nodes.

802.11b experiments run on a university testbed at data rates 1, 2, 5.5 and 11 Mbps. We refer to this testbed as University testbed. This testbed consists of 40 nodes located along the hallways of 2 adjacent department buildings in Stanford University. Packard Electrical Engineering Building houses 15 nodes specific to this network spread out across 4 floors. 25 nodes are located in Gates Computer Science Building and are evenly distributed across 6 floors. These nodes use the Madwifi driver and Click modular router [11] on a Linux kernel. Ethernet cables provide an IP based back-channel. In both the buildings heavy 802.11b/g traffic exists that are not under our control.

Apart from these two testbeds, we use several ad-hoc testbeds to explore answers to specific questions that arise. We also use publicly available Roofnet [14] datasets.

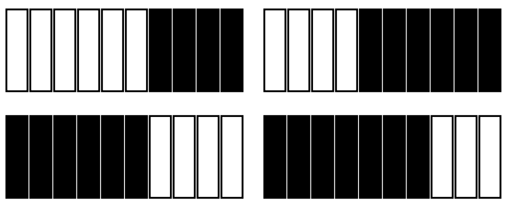
3. THE INTER-LINK CORRELATION METRIC

This section shows that reception at multiple links is correlated. It explores a metric that captures such correlations in a way useful to understand performance of protocols that allow a node to use multiple links to route packets. This section also outlines what the desired property is of such a metric. It investigates two existing metrics to measure this inter-link correlation: χ, a conditional probability metric that led earlier work to conclude that reception on links is independent of each other [16, 15] and ρ, the standard cross-correlation index used in statistics [6]. This section shows that χ and ρ do not have the desired property and defines a normalized form of ρ as the inter-link reception correlation metric, κ. It



(a) Simplest Scenario (same PRR) (b) General Scenario (different PRRs)

Figure 2: Reception success (white) and failure (black) patterns on a link pair with perfectly positive correlation.



(a) Simplest Scenario (PRRs sum to 1.0) (b) General Scenario (PRRs don't sum to 1.0)

Figure 3: Reception success (white) and failure (black) scenarios on a link pair with perfectly negative correlation.

presents κ measurements for several 802.15.4 and 802.11 testbeds, showing that κ varies across networks and link layers.

3.1 Correlated Losses

A small experiment shows inter-link reception correlations. A single node on the Mirage testbed transmits 200 broadcast packets on channel 16. All the nodes that hear the packets report to a centralized server.

Figure 1(a) shows packet reception at different receivers from empirical measurements. Packet losses are marked by black overlines. Long vertical overlines indicate that packets are lost at multiple receivers. For visual comparison, Figure 1(b) shows how reception across receivers looks like when losses are independent using synthetically generated traces. The comparison shows that packet losses on different links in Figure 1(a), from empirical measurements, are correlated. While Figure 1 shows that reception correlations exist, quantifying such correlations is desirable in understanding their implications to protocol performance.

3.2 Desired Property

One of the goals of this work is to explore a metric that captures inter-link reception correlation in a way meaningful to understand protocols that use multiple links simultaneously to route packets. Examples of such protocols include opportunistic routing and network coding protocols. In such protocols, every node chooses a set of links, from all the available links, to reduce number of transmissions to let a packet make progress towards destination.

After choosing the first link, a node has to carefully choose the next link in the set. The node does not improve the chances of a packet making progress if transmissions that succeed on the second link are a subset of transmissions that succeed on the first link. It's desirable for our metric to identify such link pairs as perfectly positively correlated. Figures 2(a) and 2(b) show perfectly positively correlated link pairs for two different cases; when both the links have same PRR and when they have different PRRs.

On the other hand, if the second link's transmissions that fail are a subset of transmissions that succeed on the first link, then a node maximizes the chances of every packet making progress. It's desirable for our metric to identify such link pairs as perfectly negatively correlated. Figures 3(a) and 3(b) show perfectly negatively correlated link pairs for two different cases; when the PRRs of the two links sum to 1 and when the PRRs don't sum to 1.

3.3 Exploring Correlation Metric: χ

Previous work has used cross-conditional probability, χ, as the inter-link correlation metric [16, 15]. χ is a 3-tuple quantity, defined on a transmitter, t, and two receivers, x and y, that can hear packets from t, as:

$$\chi_{t,x,y} = P^{(t)}_{x/y}(0/0) - P^{(t)}_{x}(0), \tag{1}$$

where, $P^{(t)}_{x/y}(0/0)$ is the probability, when t transmits, that a packet failed on link t$\rightarrow$x given that it failed on link t$\rightarrow$y and $P^{(t)}_{x}(0)$ is the probability that a packet failed on link t$\rightarrow$x. If the failures on the two links, t$\rightarrow$x and t$\rightarrow$y are independent then χ is 0.

A simple example demonstrates why χ is not a good inter-link correlation metric for our purposes. We generate a synthetic trace of packet receptions on two links with varying PRRs, for three cases. In the first case, the reception on the link pair are independent. In the second case, the reception on the link pair has perfect positive correlation with the two links having same PRR. This is the simplest perfectly positively correlated case shown in Figure 2(a). In the third case, the reception on the link pair has perfect negative correlation with the PRRs summing to 1. This is the simplest perfectly negatively correlated case shown in Figure 3(a).

Figure 4(a) shows calculated values of χ over a range of PRRs for two independent links. The metric properly reflects the independence of the uncorrelated links with near zero values. However, in Figure 4(b), for the simplest case of perfectly positively correlated links, χ fails to identify the correlation, but rather reflects the PRR of the receiver in question, causing confusion as to whether the links may be only partially correlated or even independent. As PRR approaches 0, it is clear why the two links can be misinterpreted as independent links when using the χ metric. Figure 4(c) also demonstrates this shortcoming for the simplest case of perfectly negatively correlated links. This shows that the cross conditional probability metric fails to identify correlation.

3.4 Exploring Correlation Metric: ρ

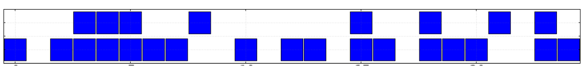
As an alternative, we consider a popular quantity in statistics that measures correlation between two quantities: the cross-correlation index, ρ. ρ, in our definition, is a 3-tuple of one transmitter, t and two random variables, x and y, corresponding to reception at two receivers. This paper assumes that x and y are random variables representing 1 for a successful reception and 0 for a failure (they are therefore Bernoulli distributions). The rest of this paper uses "x" and "y" to refer to both the receivers and their corresponding random variables. ρ is defined as:

$$\rho_{t,x,y} = \begin{cases} \frac{E[x.y]-E[x].E[y]}{\sigma_x.\sigma_y}, & \sigma_x.\sigma_y \neq 0 \\ 0, & \text{otherwise} \end{cases} \tag{2}$$

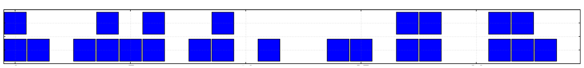
where $\sigma_x = \sqrt{E[(x-E[x])^2]}$ is the standard deviation of x, E[x.y] is the empirical mean of the product of x and y, E[x] is the mean of x, and E[y] is the mean of y. The normalization factor is the maximum difference possible between E[x.y] and E[x].E[y]. It comes from the Cauchy-Schwarz inequality:

(a) Independent (b) Positively Correlated (c) Negatively Correlated

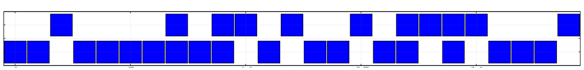
Figure 4: ρ, κ and χ for synthetic data traces with varying PRRs. The PRRs for the two links are equal for the independent and positive correlation case and $PRR_2 = 1 - PRR_1$ for the negative correlation case. For these cases, $\kappa = \rho$. The cross-conditional metric, χ does not always correctly identify receiver pairs as correlated or negatively correlated, while ρ and κ do.



(a) κ=0.0, ρ=0.0



(b) κ=1.0, ρ=ρ^{max}=0.53



(c) κ=-1.0, ρ=ρ^{min}=-0.49

Figure 5: Packet reception sample for synthetically generated traces with different κ's. Blue squares correspond to successful packet reception. The PRR's for the two links are 0.4 and 0.7 in each case. (a) independent receiver pairs, (b) perfectly correlated reception and (c) negatively correlated reception.

$$E[(x-E[x]).(y-E[y])] \leq \sqrt{(E[(x-E[x])^2]).(E[(y-E[y])^2])}$$

$$E[x.y] - E[x].E[y] \leq \sigma_x.\sigma_y$$

The product mean, E[x.y] is the probability that both x and y receive the same packet, $P^{(t)}_{x,y}(1,1)$. The empirical means, E[x] and E[y], are the packet reception ratios of the links t→x and t→y, respectively. Moreover, $\sigma_x = \sqrt{P_x.(1-P_x)}$ and $\sigma_y = \sqrt{P_y.(1-P_y)}$. Therefore, ρ can be rewritten as:

$$\rho_{t,x,y} = \begin{cases} \frac{P^{(t)}_{x,y}(1,1) - P_x.P_y}{\sqrt{P_x.(1-P_x).P_y.(1-P_y)}}, & \sigma_x.\sigma_y \neq 0 \\ 0, & \text{otherwise} \end{cases} \quad (3)$$

ρ compares the probability that both links actually receive a given packet to the probability that both would receive a given packet if their receptions were independent. If the difference between these two values is zero then the receptions at x and y are independent, positive means positively correlated and negative means negatively correlated.

The graphs in Figure 4 show that, unlike χ, ρ properly identifies perfectly positive and negative correlations between links, for the simplest cases.

3.4.1 PRR Bias in ρ

While ρ identifies perfect correlations for the simplest cases, it does not do so for the generic cases. In other words, when two links with perfect positive correlation have different PRRs, ρ may not identify them. Similarly, when two links with perfect negative correlation have PRRs that don't sum to 1, then ρ may not identify them. This is due to an inherent PRR bias in ρ.

LEMMA 1. *The range of ρ of [-1,1] is not tight. The true range of ρ depends on the packet reception ratio pairs, namely P_x and P_y. The maximum ρ is given by:*

$$\rho^{max} = \frac{min(P_x, P_y) - P_x.P_y}{\sigma_x.\sigma_y} \quad (4)$$

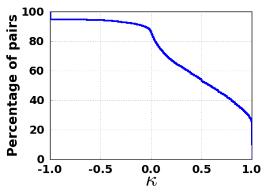
and the minimum ρ is given by:

$$\rho^{min} = \begin{cases} \frac{-P_x.P_y}{\sigma_x.\sigma_y}, & P_x + P_y \leq 1 \\ \frac{P_x+P_y-1-P_x.P_y}{\sigma_x.\sigma_y}, & \textit{otherwise} \end{cases} \quad (5)$$

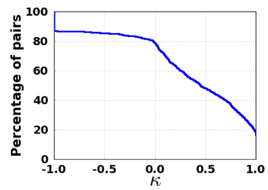
where $\sigma_x = \sqrt{P_x.(1-P_x)}$

Lemma 1 says that ρ cannot take an arbitrary value in [-1,1] for arbitrary packet reception ratio pairs. For example, if two links have different packet reception ratios, it is impossible for the two links to have received the same packets (a ρ of 1). Similarly, if the packet reception ratios of the two links do not sum up to 1, then it is impossible for every packet to be received at exactly one of the two receivers. Only when both the packet reception ratios are 0.5, the true range of ρ is [-1,1]. The proof of Lemma 1 is in the Appendix.

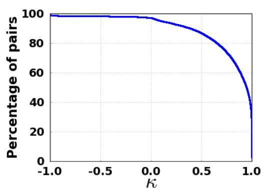
From a system's perspective, the limited range of ρ can limit ρ's usefulness. For example, consider a pair of links with PRR 0.9 and 0.1. The maximum ρ for this case is 1/9 ≈ 0.11. Getting ρ = 0.11 means that for every packet lost at the higher PRR link, the corresponding packet is also lost at the lower PRR link, and for every packet received at the lower PRR link, the corresponding packet is received at the higher PRR link. For any protocol that exploits spatial diversity of links, after adding the higher PRR link, there is no gain by using the lower PRR link. Although ρ is small, such a link pair should be classified as a highly correlated pair. For this reason, we further normalize ρ such that ρ^{max} is mapped to a 1 and ρ^{min} is mapped to -1. We define κ as this normalized ρ.



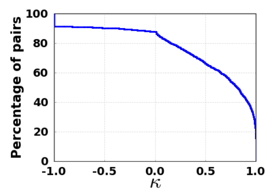
(a) Channel 26, Power 0dBm



(b) Channel 26, Power -25dBm



(c) Channel 16, Power 0dBm



(d) Channel 16, Power -25dBm

Figure 6: Complimentary CDF of κ for link pairs on Mirage. On channel 26, about 35% of the link pairs have a $\kappa > 0.8$. On channel 16, at the highest power level nearly 60% of the pairs have a $\kappa > 0.8$. At the lowest power level of -25dBm on channel 16, this percentage is close to 55%. Channel 16 shows more highly correlated link pairs than channel 26.

$$\kappa_{t,x,y} = \begin{cases} \frac{\rho_{t,x,y}}{\rho^{max}_{t,x,y}}, & \text{if } \rho_{t,x,y} > 0 \\ \frac{-\rho_{t,x,y}}{\rho^{min}_{t,x,y}}, & \text{if } \rho_{t,x,y} < 0 \\ 0, & \text{otherwise} \end{cases} \tag{6}$$

3.5 Understanding κ

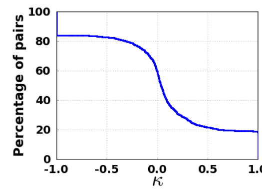
A $\kappa_{t,x,y}$ of zero means that reception at x and y are independent for packets from t. κ can be a maximum of 1 corresponding to perfect reception correlation. If κ = 1 and if $PRR_x > PRR_y$, then if x receives a packet then necessarily y receives the same packet and if y loses a packet, then x also loses the same packet. κ can be a minimum of -1 corresponding to perfect negative correlation. If κ = -1 and if $PRR_x + PRR_y < 1$ then x and y never receive the same packet, and if $PRR_x + PRR_y > 1$, then x and y never lose the same packet.

Figure 5 shows how reception from synthetically generated traces appear at receivers x and y, and the corresponding κ values. Figure 5(a) shows reception on links when the traces for the two links are generated independently. κ identifies these traces to be independent. Figure 5(b) shows traces for two links in which, when the lower PRR link receives a packet the other link also receives that packets and when the higher PRR link loses a packet the other link also loses the same packet. This link pair has a perfect correlation and κ is 1.0. Figure 5(c) shows traces for two links in which, when the higher PRR link loses a packet it is necessarily received on the other. This pair has a perfect negative correlation and κ is -1.0.

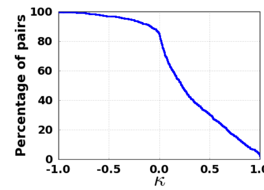
Note that the comparison of ρ against χ in Figure 4 still holds for κ, as for the cases discussed in that figure ρ=κ. In other words, κ can also identify the simplest cases of perfectly correlated, independent and perfectly negatively correlated link pairs, just as ρ.

3.6 κ on Testbeds

We run an experiment to measure κ on testbeds. Every node takes a turn to send a burst of 50,000 broadcast packets. Every receiver that receives a packet, sends the successfully received sequence numbers over the wired back-channel to a server. The server runs this experiment for different parameters such as channel, data rate (for 802.11) and transmission power level.



(a) Roofnet 11Mbps



(b) University 11Mbps

Figure 7: Complimentary CDF of κ for receiver pairs on the University and Roofnet 802.11 testbeds. Less than 20% of the link pairs have a $\kappa > 0.8$. Roofnet has many more negatively correlated links than University.

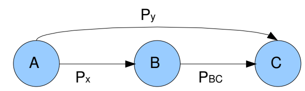


Figure 8: A 3-node 2-hop network. Node A is the source and Node C is the destination. The best shortest ETX path is A→B→C. However, Node C can sometimes hear A directly.

Figures 6 and 7 show the complimentary cumulative distribution function (CCDF) of κ for all communicating node pairs for all transmitters. These plots include, for two different testbeds, all possible pairs of links that heard at least one packet from a transmitter.

Figure 6 shows CCDFs for the Mirage testbed on channels 26 and 16 for two power levels: 0dBm and -25dBm. Channel 16 shows more link pairs to have correlated reception than channel 26. We found that channel 16 on Mirage overlaps with a cohabited 802.11 network. The 802.11 nodes being higher power systems than 802.15.4 cause losses on multiple 802.15.4 links just as Figure 1 showed.

Figure 7 shows the CCDFs for the University testbed and the Roofnet datatraces at the maximum transmit power level and 11Mbps transmit rate. These plots are representative of CCDFs at other power levels and rates and are not shown for brevity. Less than 20% of all the link pairs have a $\kappa > 0.8$.

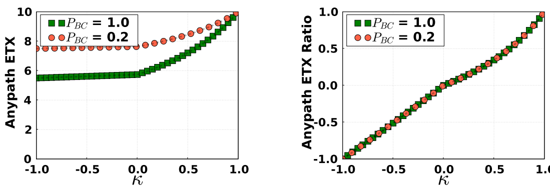
3.7 Summary

The κ metric is based on the cross-correlation index. It varies across networks. The same link pair can have different κ's depending on the channels, power levels and the data rate.

Overall, there are link pairs on all the testbeds that have highly correlated reception. The next three sections explore how relevant inter-link reception correlation is to the performance of three protocols: a simple opportunistic reception scheme, the ExOR protocol [3] and Rateless Deluge [7].

4. OPPORTUNISTIC ROUTING

This section examines how κ can help predict the performance and benefits of a simple opportunistic reception routing protocol in a simple network setup. We study κ's usefulness using mathematical analysis and experiments. In a simple 2-hop 3-node set up, κ is an excellent predictor of opportunistic routing. The next section presents this study extended for general network setups.



(a) Anypath ETX (b) Anypath ETX Ratio

Figure 9: Anypath ETX and Anypath ETX Ratio based on Equation 8 for P_{BC}=1.0 and 0.2 with P_x=0.1 and P_y=0.1. Anypath ETX Ratio is easier to visually compare than Anypath ETX.

4.1 A Simple Opportunistic Protocol

We consider an opportunistic routing protocol, similar in flavor to the ExOR [3] protocol in 802.11. Every node has a set of potential nexthop nodes. The nexthop list is prioritized such that if a node receives a packet, it will forward that packet only if none of the higher priority nodes receive the same packet. In reality, a receiver coordination scheme is needed to make sure that all the nexthops know which nodes received this packet. In this section, however, we assume perfect receiver coordination and analyze the performance. Throughout this paper, we refer to the average number of transmissions to get a packet from a source to destination using opportunistic routing as the anypath ETX [12].

We start our analysis with a simple network of 3 nodes, namely A, B and C. In this setup, A is the source and C is the destination. Figure 8 shows this setup along with the packet reception ratios of all the links. A→B→C is the shortest ETX path but we allow opportunistic routing i.e. if C hears the packets from A then B will not forward such packets. We use random variables x and y to indicate a successful reception on links A→B and A→C respectively. For this setup, the anypath ETX from A to B is:

$$E[A] = P_{x,y}^{(A)}(1,0)(1+1/P_{BC}) + P_{x,y}^{(A)}(0,1) + P_{x,y}^{(A)}(1,1) + P_{x,y}^{(A)}(0,0)(1+E[A])$$

$$= \frac{1+P_{x,y}^{(A)}(1,0)/P_{BC}}{\left(1-P_{x,y}^{(A)}(0,0)\right)} \tag{7}$$

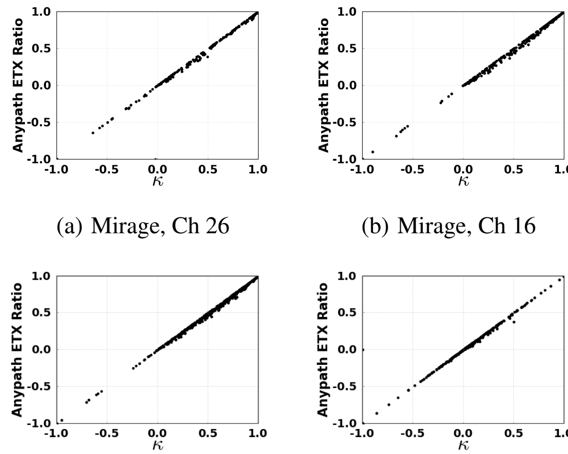
where, $P_{x,y}^{(t)}(b,c)$ is the probability that $x = b$ and $y = c$ when t transmits, $b, c \in \{0, 1\}$.

Equation 7 shows that the total number of transmissions from A to C is the sum of transmissions from A until B or C gets that packet and the transmissions from B to C of the packets from A that only B received. It also shows that the total number of transmissions is a function of the packet reception ratio of link B→C, and the joint probability statistics of links A→B and A→C. Note that κ that is relevant for this setup is that of the link pair A→B and A→C.

From Equations 3, 6 and 7, and using simple arithmetic, we can rewrite the anypath ETX (in Equation 7) as a function of κ, as:

$$E[A] = \begin{cases} \frac{1+[P_x.(1-P_y)-\kappa.\rho^{max}.\sigma_x.\sigma_y]/P_{BC}}{1-(1-P_x).(1-P_y)-\kappa.\rho^{max}.\sigma_x.\sigma_y}, & \kappa \geq 0 \\ \frac{1+[P_x.(1-P_y)-\kappa.\rho^{min}.\sigma_x.\sigma_y]/P_{BC}}{1-(1-P_x).(1-P_y)-\kappa.\rho^{min}.\sigma_x.\sigma_y}, & \kappa < 0 \end{cases} \tag{8}$$

where $\sigma_x = \sqrt{P_x.(1-P_x)}$, and ρ^{max}, ρ^{min} are given by Equations 4 and 5.



(a) Mirage, Ch 26 (b) Mirage, Ch 16

(c) University, 11Mbps (d) Roofnet, 11Mbps

Figure 10: Anypath ETX ratio vs κ for the 2-hop, 3-node network on testbeds. On all the testbeds, the average anypath ETX ratio increases as κ increases.

Equation 8 shows that the anypath ETX for the 2-hop setup can be directly inferred when the packet reception ratios of all the links involved is known and by knowing κ. It is not unreasonable to assume that the packet reception ratios are known; opportunistic protocols such as MORE use these packet reception ratios to compute the nexthop list for every node. The main purpose of this exercise is to motivate the need for measuring κ.

Lemma 2. *The anypath ETX for the 3-node network is monotonically non-decreasing in κ; the anypath ETX is maximum when κ=1 and is minimum when κ=-1.*

Lemma 2 says that the anypath ETX is minimum when receptions at B and C are negatively correlated, and that the anypath ETX increases with the increase in correlation. The anypath ETX when the receptions are independent is between the max and min anypath ETXs, when κ=0. The proof of this lemma is in the Appendix.

4.2 Anypath ETX and κ

Plotting anypath ETX, without any normalization, against κ does not give discernible trends. Figure 9(a) plots the anypath ETX using Equation 8 by varying κ for P_x=0.1, P_y=0.1 and two values of P_{BC}: 1.0 and 0.2. It shows that the anypath ETX can be quite different depending on P_{BC} and so it is not convenient to observe how it relates to κ. To plot the anypath ETX for any value of P_{BC} and still observe how it relates to κ, we normalize the anypath ETX as:

Anypath ETX Ratio =

$$\begin{cases} \frac{E[A]-E[A]_{inde}}{E[A]_{max}-E[A]_{inde}}, & E[A] \geq E[A]_{inde} \\ \frac{E[A]-E[A]_{inde}}{E[A]_{inde}-E[A]_{min}}, & \text{otherwise.} \end{cases} \tag{9}$$

where, the max, independent and min anypath ETX's for the 3-node setup are given by:

$$E[A]_{max} = \frac{1+(P_x - min(P_x,P_y))/P_{BC}}{max(P_x,P_y)} \tag{10}$$

$$E[A]_{inde} = \frac{1+P_x.(1-P_y)/P_{BC}}{P_x+P_y.(1-P_x)} \tag{11}$$

$$E[A]_{min} = \begin{cases} \frac{1+P_x/P_{BC}}{P_x+P_y}, & P_x + P_y \leq 1 \\ 1 + (1 - P_y)/P_{BC}, & \text{otherwise.} \end{cases} \quad (12)$$

The anypath ETX ratio is zero when the true anypath ETX matches the anypath ETX computed by assuming inter-link receptions to be independent. Note that the opportunistic routing protocols make this independent assumption and so an anypath ETX ratio of zero means that the estimate from the opportunistic routing protocols match the true anypath ETX. The ratio is positive when opportunistic routing protocol under-estimates the anypath ETX, with a maximum of 1 when the under-estimation is the maximum. The anypath ETX ratio is negative when the ETX is over-estimated, with a minimum of -1 when the over-estimation is the maximum.

Figure 9(b) plots the anypath ETX ratio against κ for the same setup as in Figure 9(a). The anypath ETX ratio allows for direct comparison of paths with different P_{BC}'s. The anypath ETX increases as κ increases. This is in agreement with Lemma 2.

To validate that the monotonicity argument carries over to empirical results, we analyze the anypath ETX for the same Mirage, University and Roofnet datatraces as in Section 3.6. We look at all the cases where a node, A can send packets to node C, either directly or through B. For all such cases, we compute the actual anypath ETX from A to C, using the datatraces. From this anypath ETX, we compute the anypath ETX ratio and compare it with κ of the link pair A→B and A→C.

Figure 10 plots the anypath ETX ratio against κ for both the 802.15.4 and 802.11 testbeds. In all the testbeds, the average anypath ETX ratio increases as κ increases, as Lemma 2 pointed out.

4.3 Summary

This section shows that κ directly correlates with the performance of opportunistic routing in a simple 2-hop, 3-node setup. In fact, it shows that as κ increases the anypath ETX for a given source-destination pair also monotonically increases.

5. EXTREMELY OPPORTUNISTIC ROUTING (EXOR)

The results presented in Section 4 are for the 3-node setup. This section explores if κ is useful in understanding how an opportunistic routing protocol will perform in a general multi-hop, multiple-receiver setting.

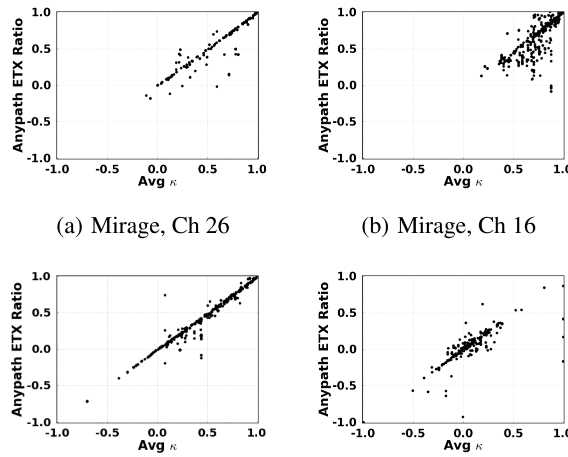
We extend the anypath ETX computation for the general case in which node t is the source, n is the destination and the anypath traverses through nodes 1,.., n-1.

$$\begin{aligned} E[t] = \frac{1}{1 - P^{(t)}_{1,2..n}(0,..,0)} \Big[1 + P^{(t)}_{1,2..n}(1,0,..,0).E[1] \\ + P^{(t)}_{2,3..n}(1,0,..,0).E[2] + .. + P^{(t)}_{n-1,n}(1,0).E[n-1] \\ + P^{(t)}_{n}(1).E[n] \Big] \end{aligned} \quad (13)$$

5.1 Experimental Methodology

We use the same Mirage, University and Roofnet data traces as in Section 3.6. For every source destination pair, the source computes the nexthop list from all the nodes that have a lower shortest path cost to the destination than the source. It includes a node in the nexthop list if that node can send at least 10% of the packets in a batch of size 100 packets. This methodology is same as the one for ExOR proposed by Biswas et al. [3].

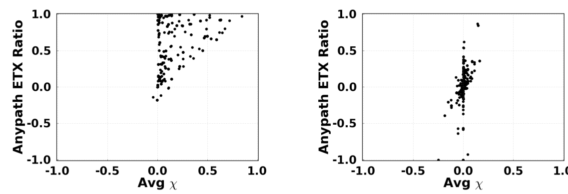
For every node in the nexthop list, we compute the anypath ETX ratio using Equation 13. For every such node, n, we compute the



(a) Mirage, Ch 26 (b) Mirage, Ch 16

(c) University, 11Mbps (d) Roofnet, 11Mbps

Figure 11: Anypath ETX ratio of a node vs average of all the κ's in the anypath for that node. On all the testbeds, the average anypath ETX ratio increases as the average κ of the anypath increases.



(a) Mirage, Ch 26 (b) Roofnet, 11Mbps

Figure 12: Anypath ETX ratio of a node vs average of all the χ's in the anypath for that node for the Mirage and Roofnet testbed traces. Many anypath ETX ratios map to a χ of $\approx$ 0.0: χ is not a good indicator of protocol performance.

average of all the κ's of all the link pairs in which n is the transmitter and the receiver is from n's nexthop list. For example, for a node 1 with nodes 2, 3 and 4 in its nexthop list, we compute the average of κ's of link pairs {1→2, 1→3}, {1→2, 1→4} and {1→3, 1→4}.

5.2 Results and Observations

Figure 11 plots the anypath ETX ratio against the average κ for every node for every source-destination pair for different testbeds. In general, the anypath ETX ratio increases as the average κ of the anypath increases; as more receiver pairs in the anypath are correlated, the error in the anypath ETX estimate from opportunistic routing increases and the estimate is usually an underestimate. Figure 12 plots the anypath ETX ratio against average anypath χ for the Mirage and Roofnet testbeds. χ is often close to 0 for different values of ETX ratio: using χ can lead to the incorrect conclusion that link pair receptions are uncorrelated. χ is a bad indicator of ExOR's performance.

Opportunistic routing protocol performance depends on the degree of inter-link reception correlations in a network. Although κ is a 2-link metric, the average of all the κ's in the anypath still gives information on how good opportunistic routing performs.

	Ch26 (κ=0.55, PRR=0.91)		Ch16 (κ=0.85, PRR=0.85)		Movement (κ=0.04, PRR=0.57)	
	Packets Sent	Latency (sec)	Packets Sent	Latency (sec)	Packets Sent	Latency (sec)
Rateless Deluge	339.2	29.8	424.6	36	473	42.8
Deluge	329.8	22.4	377.4	25	638.6	50.8
Improvement	-3%	-33%	-13%	-44%	26%	16%

Table 1: Comparison of the performance improvement obtained by using Rateless Deluge instead of Deluge for dissemination in different network environments. Each reading in the table is calculated as the average of readings from five different experiments. The table shows that when κ and PRR are high, Deluge performs more efficiently than Rateless Deluge. But in the presence of uncorrelated links and lower PRR, Rateless Deluge clearly has better performance.

6. NETWORK CODING PROTOCOLS

Previous sections showed that κ is an excellent predictor of opportunistic protocol performance. This section explores if κ can give insights on network coding protocols. Specifically, this section looks at the performance of two dissemination protocols; Deluge (which does not use network coding) [8] and Rateless Deluge [7] which uses network coding to reduce the number of packet transmissions needed for dissemination. We find that κ can guide us as to *when* network coding protocols are beneficial. The network coding technique of Rateless Deluge is more efficient only when the network reception correlation is low.

6.1 Network Coding

There is a recent thrust in the wireless networking community towards using network coding for routing [7, 10, 4]. Such protocols exploit the broadcast nature of wireless by using multiple nexthops to forward packets to a single destination. As loss patterns vary across different nexthops, each nexthop can lose different packets. To avoid knowing exactly which packets are received at which receivers, network coding protocols code multiple original packets and send the coded packets instead. The attractive feature of network coding is that, to decode N original packets, a receiver needs N linearly independent coded packets. Moreover, the N coded packets need not be consecutive.

For example, say a source node is trying to disseminate a binary image of N packets to all its neighbors. The source uses network coding to code N original packets into N linearly independent coded packets and broadcasts them. As different receivers have different link qualities from the source, in this example, they each lose a different coded packet. One of the receivers requests for one more coded packet so that it can receive N linearly independent coded packets to decode the image. The source then broadcasts one additional coded packet. As all the receivers lost only one, although different, coded packet, they all hear this additional transmission and decode the image. In the absence of network coding, each lost packet at each receiver would have to be retransmitted. Thus, network coding allows for a more efficient use of the wireless spectrum.

However, this example assumes that different links lost different packets i.e all the link pairs have perfectly negatively correlated reception. If the reception were to be perfectly correlated i.e if all the links had lost the same packet then with or without network coding the total number of transmissions would be the same. However, if network coding is used then there could be significant computation and time overhead for coding and decoding. Therefore, the extent of correlation in reception at link pairs has implications to network coding protocol performance. We use Deluge, a binary image dissemination protocol and Rateless Deluge, its network coding counterpart, to study κ's implications to network coding protocol performance. We do not use 802.11 based network coding protocols such as MORE [4] for this study because our University testbed nodes could not handle the computation demand from MORE. Moreover, MORE and its no-network-coding counterpart, ExOR [3] have very different receiver coordination schemes. This difference in the coordination schemes manifest as different control overheads in the two protocols, making such a comparison study not directly useful in understanding inter-link correlation effects. The receiver coordination scheme in both Deluge and Rateless Deluge, however, are the same.

6.2 Performance Comparison

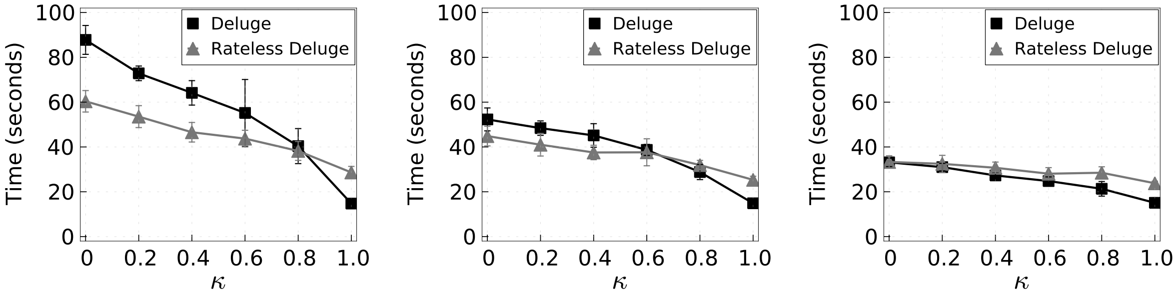
The general intuition is that when links have independent losses then network coding gives high gains and when they have correlated losses, then not to code may be better. To understand the realistic performance trade-offs of using network coding protocols, we run an experiment with Deluge and Rateless Deluge in a small testbed of 8 telosb motes. We evaluated the performance of the two protocols under different conditions and environments; different channels, power levels, etc. We present results for three specific scenarios that are representative of the rest:

- **Ch26 Scenario:** We place nodes randomly in a small part of a room measuring roughly 5'x3'x5'. The experiments ran during normal office time. This experiment uses 802.15.4's channel 26 with the power level for transmissions set to -32.5dBm.
- **Ch16 Scenario:** For this scenario, we use the same topology as in the scenario above, but use channel 16 instead. Unlike channel 26, channel 16 overlaps with 802.11 channels and is, therefore, prone to external 802.11 noise.
- **Movement Scenario:** In this scenario, nodes are spaced further apart on channel 26 with the transmission power level set to -27.5dBm. During this test, the transmitter is in motion and the people in the lab are also free to walk around.

Prior to running dissemination experiments for the three scenarios, for each scenario, the transmitter sends 50,000 broadcast packets. The receivers note down which packets they receive. We use this information to compute κ and PRR for every link pair and report the average of all such κ's and PRR's.

Table 1 shows the results from ten experiments for each of the three scenarios along with the average κ values. It shows both the number of packets sent by both Deluge and Rateless Deluge, and the total time it took by both the protocols to disseminate.

For the Ch26 scenario, Deluge outperforms Rateless Deluge in terms of dissemination time – Rateless Deluge takes 33% longer to finish. The average κ for this scenario is 0.55 and average PRR is 0.91. For this scenario, since most of the packets succeed, not to code is better, even for medium κ values.



(a) PRR = 0.4 (b) PRR = 0.6 (c) PRR = 0.8

Figure 13: Time to disseminate for Rateless Deluge and Deluge for varying values of κ and PRR. Rateless deluge is better for lower κ, but Deluge is better for higher κ. The performance switch point varies based on the PRR of the links.

For the Ch16 scenario also, Deluge outperforms Rateless Deluge. The average κ for this scenario is higher than on channel 26 and is 0.85, and the average link PRR dropped to 0.85. The number of packets sent by both Deluge and Rateless Deluge increased by 15-25% compared to the Ch26 scenario. This is due to additional losses. The higher κ value increases the gap in Deluge and Rateless Deluge performance compared to the Ch26 case, with Rateless Deluge taking 44% longer and 13% more packets than Deluge.

The results for the Movement scenario also match our intuition about the two protocols for the case of independent receptions; the average κ for this scenario is 0.04. Rateless Deluge is significantly more efficient since it effectively uses network coding to reduce the number of packets that need to be rebroadcast. Rateless Deluge sends 26% fewer packets and finishes 16% earlier than Deluge.

These results follow the general intuition and reinforce that κ is a good measure of correlation. Moreover, they show that the knowledge of κ combined with PRR can be used to choose between coding and no coding protocols. To understand exactly when it's better to use network coding protocol, we need to finely vary both the PRR and κ of all the links. Such a study is useful in getting a general rule for making a choice between the two protocols.

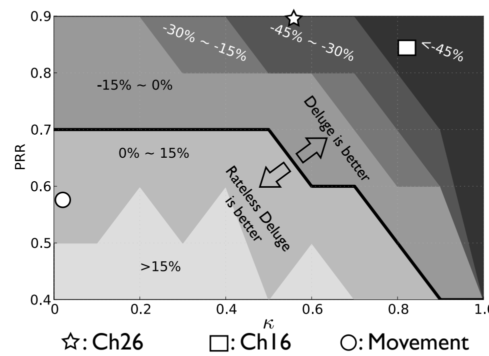


Figure 14: Performance improvements from using Rateless Deluge instead of Deluge for different network conditions. The contours correspond to specific percentage improvement values for the controlled experiment. Uncontrolled experimental results for the three cases, Ch26, Ch16, and Movement, match well with predicted results from the contour map.

6.3 Controlled Experiment

To understand the implications of inter-link correlation to Rateless Deluge, it's imperative to vary κ in a controlled way and look at Rateless Deluge's performance for each κ. A single 802.15.4 transmitter disseminates on channel 26 at maximum transmit power to 7 single-hop, near-by receivers. The maximum transmit power gives perfect links to all the receivers. The disseminating image has 9 pages with 20 packets per page.

We introduce two kinds of random losses: one at the transmitter and the other at the receivers. Losses at the transmitter are created by randomly dropping packets from the transmit queue of the node, while receiver losses are caused by randomly dropping packets from the receiver's receive queue. Transmit losses cause the same packets to be lost at all the receivers and receiver losses cause independent losses at the receivers. If P_t is the probability of packet loss at the transmitter and P_r is the probability of loss at every receiver, then varying P_t and P_r varies κ and the PRR of all the links. For example, if P_t=0.0 then the losses at the receivers are independent with a κ of 0.0 for any link pair. On the other hand, P_r=0 make all the losses to be correlated with a κ of 1.0 for any link pair. The PRR is given by $1-(P_t+(1-P_t)*P_r)$.

This experiment runs for PRR values between 0.4 and 0.9 and for each PRR value κ is varied between 0 and 1. For each combination of PRR and κ values, we run 10 experiments with Rateless Deluge and Deluge.

For all the PRRs, Rateless Deluge sends the same or fewer number of packets as Deluge. This is an expected behavior and is not shown for the same purpose. However, for the time to disseminate there is no clear winner. Figure 13 shows the total dissemination time for both Rateless Deluge and Deluge for different κ values for different PRRs (in different plots).

When κ is close to 1.0, Deluge finishes dissemination sooner than Rateless Deluge, for all the PRRs. When the links lose the same packets, both Deluge and Rateless Deluge send the same number of packets. Therefore, the additional time taken by Rateless Deluge is purely the coding overhead. The low-power cpu in Telosb motes causes the coding overhead time to be significant.

On the other hand, when κ is close to 0, Rateless Deluge finishes sooner than Deluge. For the κ values in between, the κ value where Deluge starts to out-perform Rateless Deluge depends on the PRR of the links. As links get poorer, this transition point shifts to the right.

Figure 14 shows the performance improvement with rateless deluge for different network conditions using the controlled experiments. The contours correspond to specific improvement values. This graph gives approximate decisions for when to use rateless deluge vs deluge. Moreover, this graph also provides the expected

(a) Signal Variation, Positive Correlation (b) Noise Variation, Positive Correlation (c) Signal Variation, Negative Correlation (d) Noise Variation, Negative Correlation

Figure 15: Illustration of correlated reception due to correlated variation of signal (a and c) and noise (b and d) at receivers.

gain or loss in performance by using rateless deluge for this network topology.

Figure 14 shows the accuracy of the predictions made using the data from controlled experiments. The markers on the figure show the results from the uncontrolled experiments. It is clear that the results from the uncontrolled experiments closely match the predicted results from the controlled experiments. This confirms that κ can help predict the performance tradeoff expected from using network coding.

This result suggests that the κ value of a network can be used to help decide which protocols should be used for the network.

6.4 Summary

These results show that a disseminating node can use the distribution of all its receiver pairs' κ values to decide whether to use Deluge or Rateless Deluge for data dissemination. If the distribution tends towards higher values of κ, or if the network has very high PRR links, Deluge is likely to be more efficient. This also validates our assertion that network protocols can benefit from a knowledge of the inter-receiver correlation to improve their performance and efficiency.

7. POSSIBLE CAUSES OF CORRELATION

This section explores what the possible causes that affect interlink reception correlation are. It shows that external noise from higher power systems cause losses on multiple links and increase correlated losses. Moreover, movement in a network usually blocks one link at a time and generally reduces correlations.

7.1 Positive and Negative Correlations

At the basic level, correlated losses occur due to correlated changes in the signal to interference plus noise ratios (SINR) on links operating near the limit of receiver sensitivity.

An obstacle, such as a person, may block line of sight from the transmitter to two receivers, preventing them from receiving packets. If the obstacle moves away, the signal strength at both receivers will increase, possibly high enough that they can hear the transmitter's packets. Figure 15(a) shows this scenario. This positive signal strength correlation can lead to positive packet reception correlation at the two receivers.

Another situation which can cause positive reception correlation is when external interference is positively correlated. Two nodes may both be very close to an interference source: in the case of 802.15.4, this can be an 802.11 base station as shown in Figure 15(b). A spike of external interference will cause the signal-to-noise ratio at both receivers to drop, causing correlated loss events.

Figure 15(c) shows an example of where a moving obstacle can create negative correlation. In this case, an obstacle moves such that it blocks one receiver as it moves to the left. As it clears the line of sight of that receiver, it starts blocking another. In this case, the signal strength variations at the two receivers are negatively correlated and the packet reception is also negatively correlated.

Figure 15(d) shows an example where external interference can

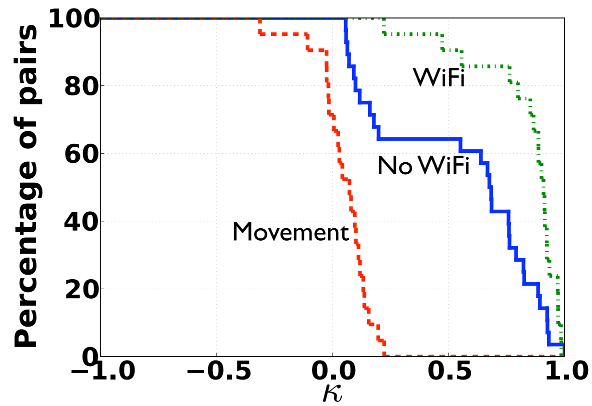


Figure 16: κ distributions of 802.15.4 link pairs when there is no WiFi interference, when there is WiFi and when there is movement. WiFi increases correlated losses and movement reduces correlated losses on link pairs.

create negative interference. High power interferers are in each other's communication range and use a CSMA/CA MAC layer. If one interferer is sending, the other keeps quiet and vice-versa. In this setup, the interferers affect different receivers. As only one of the high power interferers transmit at a time, only one of the receivers will have a high enough signal-to-noise ratio. This negative correlation in their noise leads to negative reception correlation.

Another hypothetical scenario for negative correlation is when interference sources at receivers are local and send packets periodically but unsynchronized with each other. This is possible on a testbed like Roofnet [14] in which the Roofnet links are long-haul and so different nodes can have different local access points sending interfering traffic. If these access points only send beacons, as they would in the early morning times, then they would periodically interfere with Roofnet receivers and cause losses. As the beacons from such access points are usually not synchronized, different receivers loose different packets, causing negatively correlated losses at such receivers. As this network is not under our control, we cannot confirm if this is possible in reality.

7.2 Experimental Results

To validate that some of the above discussed scenarios can occur and get a general sense of their prevalence, we dig deeper into the results for the uncontrolled experiments in Section 6 and try to explain the observed results. These experiments covered 3 scenarios. In the first scenario, we use a channel that is free of any external interference (from WiFi), channel 26. In the second scenario, we use a channel that overlaps a co-habiting access point's 802.11 channel, channel 16. In the third scenario, a person holds the transmitter and moves around until the experiment finishes on channel 26.

Figure 16 shows κ distributions of all the link pairs under all the three scenarios. The first scenario, without any WiFi interference,

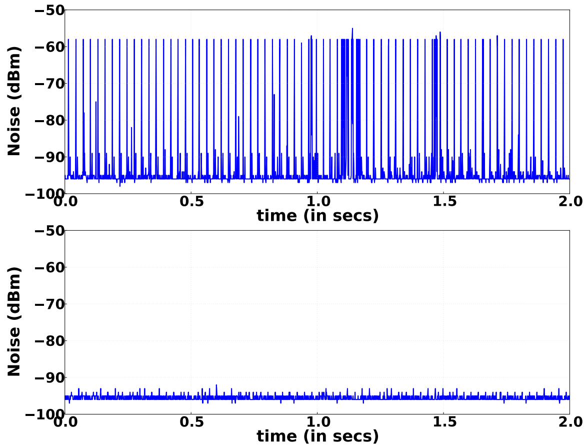


Figure 17: High frequency noise samples on channels 16 (top) and 26 (bottom). No 802.15.4 nodes were transmitting. Channel 16 shows large spikes from co-habiting 802.11 network while channel 26 is quiet.

is used as the baseline distribution. When there is WiFi interference, the distributions show an increase towards high κ values. Figure 17 shows noise floor samples from a single receiver, which is representative of the samples from other receivers, on channels 26 and 16. Channel 16 shows huge spikes from 802.11 transmissions while 26 is quiet. This shows that high power external noise sources cause losses on multiple links and increase correlated losses.

Figure 16 also shows that when there is movement, the obstacle blocks one link at a time causing loss on that link while the other links have successful reception. This introduces negative correlations and thus, reduce correlations or κ.

7.3 Summary

Variations in signal and/or noise correlated across receivers can lead to corresponding correlation in packet reception ratios. The results from experiments suggest that correlations in networks can arise from several factors. Some such factors include moving people, mobile nodes, other wireless systems, microwave ovens and so on.

8. CONCLUSION AND DISCUSSION

This paper shows that packet receptions over different links need not be independent of each other. It presents a metric called κ that captures this inter-link correlation. κ proves to be a better metric for measuring inter-link correlations than the cross-conditional metric commonly used in current literature. It shows that knowing how correlated inter-link receptions are in a network can guide us select between no network coding and networking coding protocols for that network. Specifically, it shows that when the receivers have correlated reception, using Deluge is preferable over its network-coding counterpart, Rateless Deluge. κ is also useful in understanding the performance of opportunistic routing protocols like ExOR in a network. In a network with highly correlated link pairs, the complexity in running ExOR may outweigh the performance gains from using it. For example, as Roofnet link pairs are mostly negatively correlated or independent ($\kappa <= 0$), as Figure 7 shows, running opportunistic and network coding protocols on that testbed is beneficial. However, as most of the links are positively correlated on Mirage, as Figure 6 shows, a no-network-coding protocol such as Deluge could be more beneficial.

These results also suggest that a protocol that takes inter-link correlations in to account can be more efficient. A recent work [22], uses this knowledge for a dissemination application and observes 30-50% reduction in both the number of packet transmissions and the total time for dissemination. As one more example, a modified version of Rateless Deluge that falls back to regular Deluge when the receivers have correlated reception can provide the best of both the worlds and achieve greater efficiency.

In our experience, the same network can have different inter-link correlations on different channels. Therefore, when researchers publish protocol comparison results, it will be useful to report κ distribution of their network for the corresponding channel. This will allow for reproducibility and a deeper understanding of the comparison.

Current simulators like NS-2 and TOSSIM do not take link reception correlations into account. Incorporating κ as an input to simulators can lead to more accurate, testbed-specific simulations. This will allow for fair protocol performance comparisons and more realistic results. Addressing this issue is an open problem.

Significant changes to the environment can alter κ of the link pairs in that network. An online way of measuring κ will therefore be useful in designing adaptive network protocols. This is also an open question.

9. ACKNOWLEDGEMENTS

This work was supported by generous gifts from DoCoMo Capital, the National Science Foundation under grants #0831163 and #0846014, the King Abdullah University of Science and Technology (KAUST), Microsoft Research, a scholarship from the Samsung Scholarship Foundation and a Stanford Terman Fellowship. Finally, we would like to thank our shepherd, Robert Morris, and the anonymous reviewers for their invaluable comments.

10. REFERENCES

[1] ANSI/IEEE Std 802.11 1999 Edition.

[2] The Network Simulator ns-2 (v2.1b8a). http://www.isi.edu/nsnam/ns/.

[3] S. Biswas and R. Morris. Exor: opportunistic multi-hop routing for wireless networks. In *SIGCOMM '05: Proceedings of the 2005 conference on Applications, technologies, architectures, and protocols for computer communications*, 2005.

[4] S. Chachulski, M. Jennings, S. Katti, and D. Katabi. Trading structure for randomness in wireless opportunistic routing. In *SIGCOMM '07: Proceedings of the 2007 conference on Applications, technologies, architectures, and protocols for computer communications*, 2007.

[5] R. Fonseca, O. Gnawali, K. Jamieson, and P. Levis. Four bit wireless link estimation. In *The Sixth Workshop on Hot Topics in Networks (HotNets-VI)*, Nov. 2007.

[6] R. Gray and L. Davisson. *Introduction to Statistical Signal Processing*. Cambridge University Press, New York, NY, USA, 2005.

[7] A. Hagedorn, D. Starobinski, and A. Trachtenberg. Rateless deluge: Over-the-air programming of wireless sensor networks using random linear codes. In *IPSN '08: Proceedings of the 7th international conference on Information processing in sensor networks*, pages 457–466, Washington, DC, USA, 2008. IEEE Computer Society.

[8] J. W. Hui and D. Culler. The dynamic behavior of a data dissemination protocol for network programming at scale. In

Proceedings of the Second International Conferences on Embedded Network Sensor Systems (SenSys), 2004.

[9] Intel Research Berkeley. Mirage testbed. `https://mirage.berkeley.intel-research.net/`.

[10] S. Katti, H. Rahul, W. Hu, D. Katabi, M. Médard, and J. Crowcroft. Xors in the air: practical wireless network coding. In *SIGCOMM '06: Proceedings of the Conference on Applications, technologies, architectures, and protocols for computer communications*, 2006.

[11] E. Kohler, R. Morris, B. Chen, J. Jannotti, and M. F. Kaashoek. The Click modular router. *ACM Transactions on Computer Systems*, 18(3):263–297, August 2000.

[12] R. Laufer, H. Dubois-FerriÃÍre, and L. Kleinrock. Multirate anypath routing in wireless mesh networks. In *Proceedings of IEEE Infocom 2009*, Rio de Janeiro, Brazil, April 2009.

[13] P. Levis, N. Lee, M. Welsh, and D. Culler. TOSSIM: Simulating large wireless sensor networks of tinyos motes. In *Proceedings of the First ACM Conference on Embedded Networked Sensor Systems (SenSys 2003)*, 2003.

[14] MIT Roofnet. 802.11 testbed. `http://pdos.csail.mit.edu/roofnet/`.

[15] A. Miu, G. Tan, H. Balakrishnan, and J. Apostolopoulos. Divert: Fine-grained path selection for wireless lans. In *Proceedings of the Second International Conference on Mobile Systems, Applications, and Services (MobiSys 2004)*, 2004.

[16] C. Reis, R. Mahajan, M. Rodrig, D. Wetherall, and J. Zahorjan. Measurement-based models of delivery and interference in static wireless networks. In *SIGCOMM*, pages 51–62, 2006.

[17] F. Stann, J. Heidemann, R. Shroff, and M. Z. Murtaza. RBP: Reliable broadcast propagation in wireless networks. Technical Report ISI-TR-2005-608, USC/Information Sciences Institute, November 2005.

[18] R. Szewczyk, J. Polastre, A. Mainwaring, and D. Culler. Lessons from a sensor network expedition. In *Proceedings of the First European Workshop on Sensor Networks (EWSN)*, Berlin, Germany, Jan. 2004.

[19] C. Technology. Micaz datasheet. `http://www.xbow.com/Products/Product_pdf_files/Wireless_pdf/MICAz_Kit_Datasheet.pdf`, 2006.

[20] The Institute of Electrical and Electronics Engineers, Inc. Part 15.4: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (LR-WPANs), Oct. 2003.

[21] TinyOS. MultiHopLQI. `http://www.tinyos.net/tinyos-1.x/tos/lib/MultiHopLQI`, 2004.

[22] T. Zhu, Z. Zhong, T. He, and Z.-L. Zhang. Exploring link correlation for efficient flooding in wireless sensor networks. In *Proceedings of the First USENIX/ACM Symposium on Network Systems Design and Implementation (NSDI)*, 2010.

APPENDIX

Proofs of the Lemmas

Proof of Lemma 1: We wish to determine the maximum and minimum values that ρ can take on. If P_x and P_y are kept constant, then $E[x] = P_x, E[y] = P_y, \sigma_x = \sqrt{P_x \cdot (1 - P_x)}, \sigma_y = \sqrt{P_y \cdot (1 - P_y)}$ are all constant as well. From the expression of ρ in Equation 2, we can see that the only variable term is $E[x \cdot y]$. Thus to maximize/minimize ρ we must maximize/minimize $E[x \cdot y]$.

We have that $E[x \cdot y] \leq E[x] = P_x$ and $E[x \cdot y] \leq E[y] = P_y$. These imply that $E[x \cdot y] \leq \min(P_x, P_y)$. Assuming without loss of generality that $P_x \leq P_y$. This inequality can be achieved as an equality by the distribution where:

$\{Px, y(1,1) = P_x, P_{x,y}(1,0) = 0, P_{x,y}(0,1) = P_y - Px, P_{x,y}(0,0) = 1 - P_y\}$.

Thus the maximum value of $E[x \cdot y] = \min(P_x, P_y)$, which yields the maximum value of ρ indicated in Equation 4.

Since $(x-1), (y-1)$ are both negative, $E[(x-1) \cdot (y-1)] = E[x \cdot y - y - x + 1] \geq 0$. Rearranging terms, and noting again that $E[x] = P_x, E[y] = P_y$, we get that $E[x \cdot y] \geq P_x + P_y - 1$. Further, since x and y are both non-negative random variables, we have that $E[x \cdot y] \geq \max(0, P_x + P_y - 1)$. This inequality is achieved with equality under the following two distributions. When $P_x + P_y \leq 1$, the following distribution minimizes $E[x \cdot y]$ to 0:

$\{P_{x,y}(1,1) = 0, P_{x,y}(1,0) = P_x, P_{x,y}(0,1) = P_y, P_{x,y}(0,0) = 1 - P_x - P_y\}$.

When $P_x + P_y \geq 1$, the following distribution minimizes $E[x \cdot y]$ to $P_x + P_y - 1$:

$\{P_{x,y}(1,1) = P_x + P_y - 1, P_{x,y}(1,0) = 1 - P_y, P_{x,y}(0,1) = 1 - P_x, P_{x,y}(0,0) = 0\}$.

These minimum values of $E[x \cdot y]$ correspond to the minimum values of ρ indicated in Equation 5. $\square$

Proof of Lemma 2: This lemma applies only if $P_{BC} > P_y$. This basically means that B is included as a possible next hop only if it has a better path to destination than A, which is a common assumption for opportunistic routing protocols. Under this constraint, the derivative $\frac{dE[A]}{d\rho}$ of the expression in Equation 8 can be shown to be:

$$\frac{dE[A]}{d\rho} = \frac{\sigma_x.\sigma_y.(P_{BC} - P_y)/P_{BC}}{(1 - (1 - P_x).(1 - P_y) - \rho.\sigma_x.\sigma_y)^2} \tag{14}$$

where $\sigma_x = \sqrt{P_x \cdot (1 - P_x)}$. Thus, for $P_{BC} > P_y$, $E[A]$ is monotonically non-decreasing in ρ. Further, for a given set of link PRRs, $\kappa \propto \rho$ for both $\rho > 0$ and $\rho < 0$. Also, $\kappa > 0$ for $\rho > 0$, $\kappa < 0$ for $\rho < 0$ and $\kappa = 0$ for $\rho = 0$. Then, ρ is monotonically non-decreasing with κ. By transitivity of monotonicity, $E[A]$ is monotonically non-decreasing with κ. $\square$