DMT of Parallel-Path and Layered Networks Under the Half-Duplex Constraint

K. Sreeram, S. Birenjith, and P. Vijay Kumar, *Fellow, IEEE*

***Abstract*—In this paper, we study the diversity-multiplexing-gain tradeoff (DMT) of wireless relay networks under the half-duplex constraint. It is often unclear what penalty if any, is imposed by the half-duplex constraint on the DMT of such networks. We study two classes of networks; the first class, called KPP(I) networks, is the class of networks with the relays organized in K parallel paths between the source and the destination. While we assume that there is no direct source-destination path, the K relaying paths can interfere with each other. The second class, termed as layered networks, is comprised of relays organized in layers, where links exist only between adjacent layers.**

We present a communication scheme based on static schedules and amplify-and-forward relaying for these networks. We also show that for KPP(I) networks with $K \geq 3$, the proposed schemes can achieve full-duplex DMT performance, thus demonstrating that there is no performance hit on the DMT due to the half-duplex constraint. We also show that, for layered networks, a linear DMT of $d_{\max}(1-r)^+$ between the maximum diversity $d_{\max}$ and the maximum MG, $r_{\max} = 1$ is achievable. We adapt existing DMT optimal coding schemes to these networks, thus specifying the end-to-end communication strategy explicitly.

***Index Terms*—Amplify-and-forward protocols, cooperative diversity, diversity-multiplexing gain tradeoff, explicit codes, KPP networks, layered networks, multi-antenna networks, multi-hop networks, relay networks.**

I. Introduction

WIRELESS relay networks are a class of multi-terminal networks comprised of a source with messages to be communicated to a sink and other nodes, termed relays, whose purpose is to aid in the source-sink communication. While much is known about general full-duplex relay networks, the additional complexity arising from the half-duplex constraint, has not been fully understood.

Manuscript received July 22, 2008; revised April 24, 2012; accepted May 15, 2012. Date of publication March 19, 2013; date of current version May 15, 2013. This work was supported in part by NSF-ITR Grant CCR-0326628, in part by the DRDO-IISc Program on Advanced Mathematical Engineering, and in part by Motorola's University Research Partnership Program with IISc. This paper was presented in part at the 10th International Symposium on Wireless Personal Multimedia Communications, December 2007, in part at the Information Theory and Applications Workshop, San Diego, January 2008, and in part at the 2008 IEEE International Symposium on Information Theory.

K. Sreeram is with the University of California, Berkeley, CA 94704 USA (e-mail: ksreeram@berkeley.edu).

S. Birenjith is with the Department of Electrical Communication Engineering, Indian Institute of Science, Bangalore, India (e-mail: biren@ece.iisc.ernet.in).

P. V. Kumar is with the Department of Electrical Communication Engineering, Indian Institute of Science, Bangalore, India (e-mail: vijay@ece.iisc.ernet.in).

Communicated by M. Gastpar, Associate Editor for Shannon Theory.

Digital Object Identifier 10.1109/TIT.2013.2253162

In a half-duplex network, a node cannot both transmit and receive simultaneously. When there is more than one relay node, the transmission and reception of the various nodes have to be carefully orchestrated in order to meet the half-duplex constraint for all the nodes, a process termed as scheduling. In the networking literature, while protocols for centralized and decentralized scheduling are extensively studied, physical layer effects are not fully considered. In the case of relay networks, an information theoretic understanding of even centralized scheduling seems lacking.

In this paper, we initiate a fundamental study of the DMT of multi-hop half-duplex networks. We study the DMT of two classes of half-duplex networks: the first class, called KPP(I) networks, is the class of networks with the relays organized in K parallel paths between the source and the destination. While we assume that there is no direct source-destination path, the K relaying paths can interfere with each other. The second class, called layered networks, is comprised of relays organized in layers, where links exist only between adjacent layers. We show that, for layered networks, a linear DMT (of the form $d_{\max}(1-r)^+$) between the maximum diversity of $d_{\max}$ and maximum MG of 1 can be achieved. We also show that for KPP(I) networks, the proposed schemes can achieve full-duplex DMT performance, thus demonstrating that there is no performance hit arising from the half-duplex constraint. While the paper is mainly concerned with nodes with single antennas, we also consider extensions to multiple antenna nodes. We also show that existing code constructions for the point-to-point channel can be adapted to the relay network. The key technical idea in the paper is to construct schemes for a simple network model called KPP network, which are then used as a building block to construct schemes for other complicated network models including KPP(D) networks, which do not have direct link, KPP(I) networks and layered networks.

A. Prior Work

There are three main lines of work that are related to the current paper; the first one involving the study of DMT of half-duplex two-hop relay networks, the second relating to the DMT of full-duplex multi-hop relay networks and the third studying the capacity of arbitrary full-duplex relay networks. We summarize related prior work in these areas in the following.

1) Two-Hop Networks: The Diversity-Multiplexing gain Tradeoff (DMT) as a means of evaluating point-to-point, multiple-antenna schemes in the context of slow-fading channels was proposed by Zheng and Tse [5]. Beginning with the viewpoint of relaying as a mechanism for cooperatively increasing diversity [3], [4], several papers in the literature have studied

two-hop half-duplex relay networks with the DMT as a performance criterion [6]–[11], [16]. However, the DMT of even simple half-duplex relay networks remain unknown, except for the single-relay network solved in [24].

In the present paper, the DMT of two-hop, half-duplex relay networks in the absence of a direct link between the source and destination is fully characterized and shown to equal that under full-duplex operation. In a parallel and independent work [26], the same result is obtained using a completely different method.

2) Multi-Hop Networks: In the case of multi-hop networks (with potentially multiple antennas), much less is known. There is some existing work that characterizes the DMT of *full-duplex* multi-hop networks, even in the presence of multiple antennas. This line of work was initiated in [15], where the DMT of the product of rayleigh-fading matrices was computed. This is then applied to show that a natural AF protocol where nodes simply forward their received symbols will not even achieve full diversity. To remedy this, a new AF protocol named flip-and-forward protocol is proposed, which achieves full diversity. Building on this, new AF protocols are constructed in [36], [37] and the DMT of full-duplex multi-hop networks is fully characterized. The performance of practical coding schemes like relay selection are analyzed in [28], [29]. While most of these papers are focussed on full-duplex networks, the attention here, is on half-duplex networks. A naive strategy of applying a full-duplex protocol for a half-duplex layered network is to activate alternate layers of relays in a given time. However, this strategy can result in a rate loss factor of two, which is equivalent to a multiplexing-gain (MG) loss factor of two. In the current paper, we study multi-hop networks under the half-duplex constraint and show that the multiplexing-gain penalty can be avoided through improved scheduling.

3) Capacity: In a pioneering paper [27], the compound-channel capacity of wireless full-duplex relay networks is characterized to within a constant number of bits. As a corollary of this result, the DMT of arbitrary full-duplex relay networks is shown to equal the cut-set bound. For half-duplex relay networks, this scheme is shown to achieve the best-possible DMT for *any given static schedule*. However, it is unclear what the best static schedules are, and whether they can attain the performance of the best non-static scheduling schemes (see, for an example of non-static scheme, the dynamic-decode-and-forward scheme proposed in [7]). In the current paper, we show that for certain networks, static schedules are indeed optimal and even simple amplify-and-forward protocols are sufficient to achieve the DMT, as opposed to the infinite block-length random-coding schemes in [27].

B. Network Model

1) Setting and Channel Model: We use uppercase letters to denote matrices and lowercase letters to denote vectors/scalars. Vectors and scalars are differentiated by the context. Boldface letters are used to denote a random entity, irrespective of whether it is a random scalar, vector or a matrix. Let $G = (V, E)$ be an undirected graph used to represent the wireless network, where V is the set of nodes and the edge set E has an edge (i, j) if nodes i and j are neighbors. We adopt the following channel model:

$$\mathbf{y}_i(t) = \sum_{j:(i,j)\in E} \mathbf{H}_{ij}(t)\mathbf{x}_j(t) + \mathbf{w}_i(t), \tag{1}$$

where t denotes time, $\mathbf{y}_i(t)$ corresponds to the received signal at node i, $\mathbf{w}(t)$ is the noise vector, $\mathbf{H}_{ij}(t)$ is the channel matrix and $\mathbf{x}_j(t)$, the vector transmitted by the node j, all at time t. This model captures both the broadcast and interference nature of the wireless medium.

2) Assumptions: Our description is in terms of the equivalent complex-baseband, discrete-time channel. We follow the literature [7] in making the following standard assumptions concerning the channel model:

1) All channels are assumed to be quasi-static and to experience Rayleigh fading and hence all fade coefficients are i.i.d., circularly-symmetric, complex Gaussian $\mathbb{C}\mathcal{N}(0, 1)$ random variables.
2) The additive noise at each receiver is also modeled as possessing an i.i.d., circularly-symmetric, complex Gaussian $\mathbb{C}\mathcal{N}(0, 1)$ distribution.
3) Each receiver (but none of the transmitters) is assumed to have perfect channel-state information of all the upstream channels in the network.[1]
4) There is a long-term power constraint ρ at each node, i.e., for T large enough,

$$\frac{1}{T}\sum_{t=1}^{T} |x_j(t)|^2 \leq \rho. \tag{2}$$

We will further assume that each node has a single antenna throughout the paper, except in Section VI, where we will study the scenario where each node potentially has multiple antennas.

C. Network Taxonomy

In this section, we will describe the two classes of networks that we study in detail in this paper.

1) K-Parallel-Path Networks: We begin with a definition of a class of networks called K-parallel path networks.

Definition 1: The *K-parallel path (KPP) network* is defined as a single-source, single-sink (ss-ss) network that can be expressed as the union of K node-disjoint paths called relaying paths, each comprised of at least one relay, connecting the source to the sink. Relaying path i is labeled P_i and along path P_i, there are $n_i - 1$ relay nodes labeled $\{R_{ij}\}_{j=1}^{n_i-1}$ (see Fig. 1(a)).

The definition 1 of KPP networks precludes the possibility of either having a direct link between the source and the sink, or of having links connecting nodes lying on different node-disjoint paths. We now extend the definition of KPP networks to include these possibilities.

Definition 2: If a given network is the union of a KPP network and a direct link between the source and sink, then the network is called a KPP-network-with-direct-link and is abbreviated as *KPP(D) network*. If a given network is the union of a

[1]However, for the protocols proposed in this paper, the CSIR is utilized only at the sink. Please see Section I-D.2 for details.

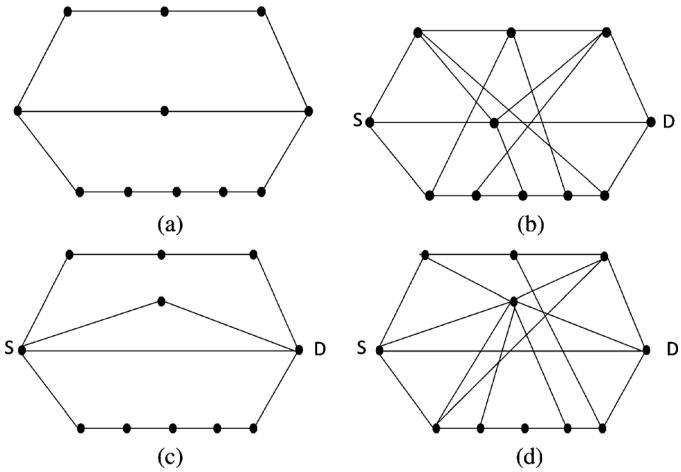

Fig. 1. Examples of KPP networks (a) KPP, $K = 3$. (b) KPP(I), $K = 3$. (c) KPP(D), $K = 3$. (d) KPP(I, D), $K = 3$.

KPP network as well as links interconnecting relays in various paths, then the network is called a KPP-network-with-interference and abbreviated by *KPP(I) network*. If a given network is the union of a KPP(I) network and a direct link, then the network is called a KPP-network-with-interference-and-direct-path and is abbreviated by *KPP(I, D) network*. Fig. 1 provides examples of all four variants of KPP networks.

Note that in KPP(I) networks, there are no interfering links emanating from the source or arriving at the destination; the interfering links are present *only between relays*.

For a KPP(D), KPP(I) or KPP(I, D) network, we can find a set of K node-disjoint paths from the source to the sink, in which every node appears once. We call the union of these K node-disjoint paths, called backbone paths $P_i, i = 1, 2, \ldots, K$, as the *backbone* KPP network. When there are many choices for the K node-disjoint paths, we will arbitrarily fix one such choice. A *start node* and *end node* of a backbone path are the first and the last relays respectively along the path. Let P_i have n_i edges. The j-th edge on the i-th path P_i will be denoted by e_{ij} and the associated fading coefficient by g_{ij}.

2) Layered Networks: We next define *layered networks*. Layered networks have been considered previously in [15], [25] and [28], albeit under different names and slightly varying definitions.

Definition 3: A network is said to be a *layered network* with L layers, if there exists a partition of the vertex set V into subsets $V_0, V_1, \ldots, V_L, V_{L+1}$ such that

1) V_0, V_{L+1} denote the singleton sets corresponding to the source and sink respectively.
2) $|V_i| > 1, i = 1, 2, .., L$.
3) There are edges only within a layer or between adjacent layers. In short, if there is an edge between a node in vertex set V_i and a node in V_j, then $|i - j| \leq 1$.
4) For any two nodes that lie in adjacent layers, there exists an edge between them.

We note that a layered network may or may not have links within a layer, however, it does have links between every pair of nodes that lie in adjacent layers. An example of a layered network is shown in Fig. 2.

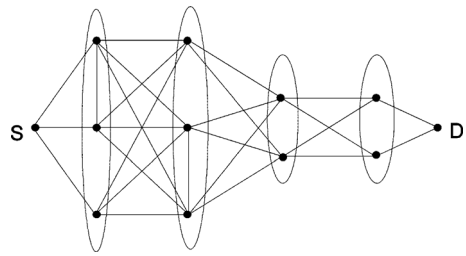

Fig. 2. An example layered network.

Remark 1: KPP(I) Vs. Layered Networks: We note that KPP(I) and layered networks are have non-trivial overlap. For example, the layered network in Fig. 2 has 2 node-disjoint paths which can be chosen in many ways. However, none of these choices give rise to the property that these paths contain all nodes in the network. Hence, this network is not a KPP(I) network. Conversely, the simple KPP network shown in Fig. 1(a) cannot be viewed as a special case of layered networks since in a layered network, there are L well defined layers such that edges are only inside a layer or between adjacent layers and such a definition is impossible in this case.

Remark 2: Two-Hop Networks: With respect to the above network taxonomy, the well-studied two-hop relay network in [7], [24] is a KPP(I,D) network with $K = M, M$ being the number of relays. In the absence of a direct link, the two-hop relay network is a special case of KPP(I) network. On the other hand, if we have a two-hop relay network with direct link but make the additional assumption of relay isolation (as in [16]), then it is a KPP(D) network with $K = M$.

D. Background

We refer the reader to Section II-A.1 of [1] for a background on the diversity-MG tradeoff (DMT).

1) Cut-Set Bound on DMT: For each of the networks described in this paper, we can obtain an upper bound on the DMT, based on the cut-set upper bound [2], [33] on mutual information, which applies to both full and half-duplex networks. This was formalized in [9] in the context of DMT as follows:

Lemma 1.1: Let r be the rate of MG at which communication between the source and the sink is taking place. Given a cut ω, there is a channel matrix $\mathbf{H}_\omega$ connecting the input terminals of the cut to the output terminals. Let us term the DMT corresponding to this $\mathbf{H}_\omega$ matrix as the DMT of the cut, $d_\omega(r)$. Then the DMT between the source and the sink is upper bounded by

$$d(r) \leq \min_{\omega \in \Lambda} \{d_\omega(r)\},$$

where Λ is the set of all cuts between source and destination.

2) Amplify and Forward Protocols: An AF protocol $\wp$ is a protocol $\wp$ in which each node in the network operates in an amplify-and-forward fashion. Let the protocol operate for T time instants. Such protocols induce a linear channel model between the vector of symbols transmitted by the source and the vector of symbols received by the sink of the form:

$$\mathbf{y} = H(\wp)x + w, \tag{3}$$

TABLE I
PRINCIPAL RESULTS SUMMARY

Network	Upper bound on DMT	Achievable DMT	Is upper bound achieved?	Reference
KPP(K $\geq$ 3)	$K(1-r)^+$	$K(1-r)^+$	✓	Theorem 2.1
KPP(D)(K $\geq$ 4)	$(K+1)(1-r)^+$	$(K+1)(1-r)^+$	✓	Theorem 3.1
KPP(I)(K $\geq$ 3)	$K(1-r)^+$	$K(1-r)^+$	✓	Theorem 4.1
Layered	Concave	$d_{\max}(1-r)^+$	✓ for $L \leq 3$ Generally, linear DMT between $d_{\max}$ and r_{max} is achieved.	Theorem 5.1

where $\mathbf{y} \in \mathbb{C}^T$ denotes the signal received at the sink, $\mathbf{w}$ is the noise vector, $\mathbf{H}(\wp)$ is the $(T \times T)$ induced channel matrix and $\mathbf{x} \in \mathbb{C}^T$ is the vector transmitted by the source.

Definition 4: Given a random matrix $\mathbf{H}$ of size $m \times n$, we define the *DMT of the matrix* $\mathbf{H}$ as the DMT of the associated channel $\mathbf{y} = \mathbf{Hx} + \mathbf{w}$ where $\mathbf{y}$ is an m-length received column vector, $\mathbf{x}$ is an n-length transmitted column vector with total power constraint ρ and $\mathbf{w}$ is a $\mathbb{CN}(0, I)$ column vector. We denote the DMT of matrix $\mathbf{H}$ by $d_H(.)$.

A schedule is necessary to determine when a given node will transmit or receive. Once a schedule is specified, the nodes while listening, store the data received in their buffers, and transmit the oldest symbol in the buffer (after scaling by a constant value to meet the power constraint) when they get a transmit opportunity[2]. Therefore this scheme does not require any channel state information (CSI) at the relays. It turns out, as pointed out in [7], that the value of the scaling constant does not affect the DMT of the network and will therefore be assumed to be equal to one[3].

Remark 3: Coding to Achieve DMT: As pointed out above, the scheme is fully specified by specifying the protocol. It should be noted, however, that in the induced channel model $\mathbf{y} = \mathbf{Hx} + \mathbf{w}$ the transmitted symbols need to be coded in order to achieve the proposed DMT. This coding is performed only at the source and DMT-optimal codes for point-to-point channels can be used here.

E. Some Results From a Related Prior Paper

In a related previous paper [1], we developed basic results that will be instrumental in deriving the DMT of certain classes of networks in the present paper. Two results from [1] that we will draw upon throughout the paper, are reproduced here for convenience.

The first result formalized in the theorem below, is that the correlated noise encountered at the sink of many AF protocols can be treated as white in the scale of interest.

Theorem 1.2: [1]: Consider a channel of the form $\mathbf{y} = \mathbf{Hx} + \mathbf{z}$, with $\mathbf{z} = \mathbf{z}_0 + \sum_{i=1}^{M} \mathbf{G}_i \mathbf{z}_i$, where the $\{\mathbf{z}_i\}_{i=0}^{M}$ are i.i.d. $\mathbb{CN}(\underline{0}, I)$ random vectors, and where each entry of the $N \times N$ random matrix $\mathbf{G}_i$ is a polynomial function of some underlying i.i.d. complex Gaussian random variables $\mathbf{h}_1, \mathbf{h}_2, \ldots, \mathbf{h}_L$. Then the DMT of the channel $\mathbf{y} = \mathbf{Hx} + \mathbf{z}$ is the same as the DMT of the channel $\mathbf{y} = \mathbf{Hx} + \mathbf{w}$ where $\mathbf{w}$ is now a $\mathbb{CN}(\underline{0}, I)$ random vector.

We also proved a result pertaining to the DMT of block-lower-triangular (blt) matrices. The ℓ-th sub-diagonal matrix, $A^{(\ell)}$ of a blt matrix A whose (i,j)th entry is the matrix A_{ij}, is defined as the matrix comprising only of the block-entries $A_{\ell 1}, A_{(\ell+1)2}, \ldots, A_{N(N-\ell+1)}$ with zeros everywhere else. The last sub-diagonal is the sub-diagonal corresponding to the largest ℓ, such that $A^{(\ell)}$ is non-zero. We now state the result on blt matrices:

Theorem 1.3: [1]: Consider a random blt matrix $\mathbf{H}$ having component matrices $\mathbf{H}_{ij}$ of size $N_i \times N_j$. Let $M := \sum_{i=1}^{N} N_i$ be the size of the square matrix $\mathbf{H}$. Let $\mathbf{H}^{(0)}$ be the diagonal part of the matrix $\mathbf{H}$ and $\mathbf{H}^{(\ell)}$ denote the last sub-diagonal matrix of $\mathbf{H}$. Then,

1) $d_H(r) \geq d_{H^{(0)}}(r)$.
2) $d_H(r) \geq d_{H^{(\ell)}}(r)$.
3) In addition, if the entries of $H^{(\ell)}$ are independent of the entries in $H^{(0)}$, then $d_H(r) \geq d_{H^{(0)}}(r) + d_{H^{(\ell)}}(r)$

F. Results

In this paper, we present simple communication schemes for KPP and layered networks and analyze the DMT of these schemes. The principal results of the paper are tabulated in Table I. In many cases, the DMT achieved equals the cut-set bound and the scheme is thereby optimal. In all these cases, it turns out that the half-duplex constraint does not entail any penalty in the DMT performance of these networks. Also, the network DMT can be achieved using existing, explicit coding schemes. Furthermore, we analyze networks with multiple antenna nodes for which we provide an achievable scheme and compute the associated DMT. Some of these results were presented in conference versions of this paper [18]–[21] (see also [22], [23]).

G. Outline

In Section II, we focus on half-duplex KPP networks and present protocols achieving the network DMT for $K \geq 3$. We extend this result to KPP(D) networks in Section III. In Section IV, KPP(I) networks with half-duplex relays are considered, and schemes achieving network DMT are presented for KPP(I) networks allowing certain types of interference. In

[2]We assume that the network is in operation for sufficient amount of time, so that if an edge is active, the node at beginning of the edge always has a symbol to transmit.

[3]The value of this scaling constant does, however, affect the finite SNR performance, a detailed study of which is beyond the scope of the present paper.

Section V, we use the schemes for KPP networks as a building block to construct schemes for layered networks and show that a linear DMT $d_{\max}(1-r)^+$ between maximum diversity $d_{\max}$ and maximum MG of 1 is achievable. In Section VI, we consider multi-antenna layered and KPP networks and give an achievable DMT. Finally, in Section VII, we show that explicit CDA based codes can achieve the DMT of all the proposed protocols.

II. Half-Duplex Networks With Isolated Paths—KPP Networks

In this section, we consider KPP networks with single-antenna nodes operating under the half-duplex constraint. We begin by stating the main result:

Theorem 2.1: For a KPP network, there exists a scheme achieving the cut-set bound of $d(r) = K(1-r)^+$ if $K \geq 3$ or $K = 2$ and $n_1 = n_2 \pmod 2$, where n_i is the length of path i.

The remainder of the section is dedicated to proving this result. First, observe that the cut-set upper bound on DMT (Lemma 1.1) for the class of KPP networks is given by: $d(r) \leq K(1-r)^+$, and therefore it is sufficient to design protocols that attain this bound.

A. Protocols for KPP Networks

As noted in the introduction, we restrict our attention in this paper to the class of AF protocols under which node operations are restricted to scaling and forwarding. To completely specify the protocol, it remains only to identify a schedule. We will consider periodic schedules, i.e., schedules in which edge activations are periodic with period N. Let $C = \{c_1, \ldots, c_N\}$ denote the set of N time instants (which we will refer to as "colors"). The protocol can be described as a coloring scheme, where $A_{ij} \subseteq C$ denotes the time instants during which edge e_{ij} is active, i.e., node u transmits and node v listens (where $e_{ij} = (u, v)$). A *simple protocol* for a KPP network is defined as a half-duplex protocol, where at any node, at any given time instant, only one of the incoming or outgoing edges is active. The main idea is that a simple protocol avoids broadcasting at the source and interference at the destination. While the reason for avoiding interference at the destination is obvious, we also avoid broadcasting at the source to ensure that each path forwards distinct symbols. Thus any edge coloring satisfying the following constraints is a simple protocol:

$$A_{i1} \cap A_{j1} = \phi, i \neq j. \tag{4}$$

$$A_{in_i} \cap A_{jn_j} = \phi, i \neq j. \tag{5}$$

$$A_{ij} \cap A_{i(j+1)} = \phi, j = 1, 2, \ldots, n_i - 1. \tag{6}$$

$$|A_{ij}| = m_i, j = 1, 2, \ldots, n_i. \tag{7}$$

An example of a simple protocol with $K = 4, N = 4$ and $m_i = 1$ for all i, is shown in Fig. 3. Thus in this example, there are four paths, edge activations are periodic with period 4 and each edge is activated once in each cycle. The label alongside each edge indicates its color, i.e., the time instant during which the edge is activated.

While a simple protocol avoids interference at the receiver, there could be interference at some of the intermediate nodes,

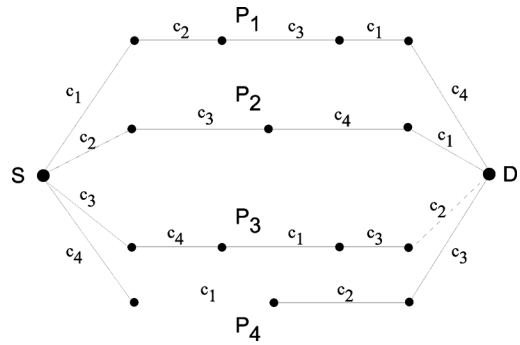

Fig. 3. KPP network simple protocol.

Fig. 4. Back-flow on a path.

which we term as back-flow. For example, consider a KPP network and let v_1, v_2, v_3, v_4 be four consecutive vertices lying on one of the K parallel paths. In a simple protocol, at a given time, let v_1 and v_3 transmit, thereby causing the edges (v_1, v_2) and (v_3, v_4) to be active. Due to the broadcast and interference constraints, transmission from v_3 interferes with the reception at v_2. This is termed as *back-flow*, and is illustrated in Fig. 4. Back-flow can be avoided by ensuring that there are at least two inactive edges between any two active edges along a backbone path:

$$A_{ij} \cap A_{i(j+2)} = \phi, j = 1, 2, \ldots, n_i - 2.$$

Lemma 2.2: If a simple protocol, operating on a KPP network, satisfies the constraints listed below, then it achieves the optimal DMT of $d(r) = K(1-r)^+$:

1) A single symbol is transmitted from the source in every time slot, i.e., $\cup_i A_{i1} = C$.
2) An equal number of symbols are transmitted through all the K backbone paths, i.e., $m_i = m \; \forall \; i$.
3) The protocol avoids back-flow.

Proof: The listed conditions ensure that the induced channel between the source transmission and reception, after permutation of symbols and accounting for delays, can be described as:

$$\mathbf{y} = \begin{bmatrix} \mathbf{g}_1 & & & \\ & \mathbf{g}_2 & & \\ & & \ddots & \\ & & & \mathbf{g}_K \end{bmatrix} \mathbf{x} + \mathbf{w}, \tag{8}$$

where $\mathbf{g}_i$ denotes the product fading coefficient of the path P_i, and $\mathbf{w}$ is the noise vector at the sink. The DMT of this channel is equal to the DMT of the above matrix $\mathbf{H}$ since the noise is white in the scale of interest (see Theorem 1.2). The DMT of $\mathbf{H}$ can be easily computed to be $d(r) = K(1-r)^+$. ■

Remark 4: It should be noted that while each scalar symbol x_i goes through only one of the K paths, it should be noted that the x_i are the output of the coding at the source, which ensures that

each *uncoded* symbol passes through all K-paths thus providing full diversity.

Remark 5: Back-Flow: The condition (3) in the theorem can be dropped for KPP networks, due to Lemma 2.3, proved later. However, we keep the condition here because of its application to KPP(D) networks later. Furthermore, while back-flow does not degrade the DMT, it can hurt finite SNR performance.

B. The Case of $K \geq 4$

For the case of $K \geq 4$ it turns out that we can always design a simple protocol which avoids back-flow. We design a generic protocol with K colors, $\{c_1, \ldots, c_K\}$ as follows: three colors (G_{i0}, G_{i1}, G_{i2}) are cyclically repeated on path i, except for the last edge of the path where we use color F_i. We use a potentially different color on the last edge to ensure that the relays transmit to the sink at distinct time instants. Thus

$$A_{ij} = \begin{cases} G_{i((j-1) \bmod 3)}, & j \neq n_i \\ F_i, & j = n_i \end{cases}, \tag{9}$$

and hence we use $G_i = (G_{i0}, G_{i1}, G_{i2}), F_i$ as descriptors of the simple protocol. For the case of $K \geq 4$, setting $(G_{i0}, G_{i1}, G_{i2}) = (c_i, c_{(i+1)}, c_{(i+2)})$ and $F_i = c_{(i+3)}$ yields a simple protocol. Here the color index should be taken modulo K whenever it strictly exceeds K. It can be easily checked that the protocol satisfies all conditions of Lemma 2.2 and hence achieves the DMT $d(r) = K(1-r)^+$.

C. The Case of $K = 3$

We established that when $K \geq 4$, there are simple protocols avoiding back-flow which achieve the network DMT. This is not the case in general when $K = 3$. In particular, avoiding back-flow is not possible in all KPP networks with $K = 3$. Therefore, we analyze the effect of back-flow and show that back-flow does not cause any loss in DMT performance.

Lemma 2.3: Consider a KPP network operating under a simple protocol, which on neglecting the effect of back-flow, induces a block-diagonal channel matrix. Then, the DMT of the protocol taking into account the effect of back-flow, is lower bounded by the DMT when neglecting the effect of back-flow.

Proof: The presence of back-flow creates entries in the strictly lower-triangular portion of the induced channel matrix. Since the DMT of a lower triangular matrix is lower bounded by the DMT of the corresponding diagonal matrix (by Theorem 1.3), we have that the system with back-flow will yield a DMT no worse than the one without back-flow. ■

Remark 6: Effect of Back-Flow at Finite SNR: While the presence of back-flow does not alter the DMT of the protocol (which is a high SNR characterization), it does impose several practical issues including the low-to-moderate SNR performance (which is affected by the power spent in re-forwarding of past information) and the complexity of decoding incurred. While we show in Section VII that to some extent the complexity issue can be addressed, a detailed study is beyond the scope of this work and is deferred for future work.

We will now exploit Lemma 2.3 to construct a protocol achieving the cut-set bound for the KPP network with $K = 3$. Without loss of generality we assume that the paths are ordered such that for the first ℓ paths, $n_i = 1 \bmod 3$ followed by the other paths. We design our protocol depending on ℓ.

1) Case 1: ($\ell = 0, 1,$ *or* 3) We will specify the protocol by specifying the activation sets $G_i = (G_{i0}, G_{i1}, G_{i2})$ and F_i for all i. We begin by setting F_i to be the natural choice: $F_i = G_{i(n_i - 1 \bmod 3)}$. The set of colors used is $C = \{c_0, c_1, c_2\}$. For $\ell = 0$,

$$G_i = \begin{cases} (\{c_i\}, \{c_{i+2}\}, \{c_{i+1}\}), & n_i = 0 \bmod 3 \\ (\{c_i\}, \{c_{i+1}\}, \{c_{i+2}\}), & n_i = 2 \bmod 3 \end{cases}.$$

For $\ell = 1$,

$$G_1 = (\{c_0\}, \{c_1\}, \{c_2\})],$$
$$G_2 = \begin{cases} (\{c_1\}, \{c_0\}, \{c_2\}), & n_2 = 0 \bmod 3 \\ (\{c_1\}, \{c_2\}, \{c_0\}), & n_2 = 2 \bmod 3 \end{cases},$$
$$G_3 = \begin{cases} (\{c_2\}, \{c_0\}, \{c_1\}), & n_3 = 0 \bmod 3 \\ (\{c_2\}, \{c_1\}, \{c_0\}), & n_3 = 2 \bmod 3 \end{cases}.$$

For $\ell = 3$,

$$G_i = (\{c_i\}, \{c_{i+1}\}, \{c_{i+2}\}).$$

This protocol avoids back-flow, uses all paths equally, and transmits fresh symbols in every time-slot. By Lemma 2.2, this protocol achieves the cut-set bound.

2) Case 2: ($l = 2$)
For $l = 2$, it turns out that back-flow cannot be avoided and hence we resort to a protocol that allows back-flow. Consider the protocol having the following descriptor: $G_i = (\{c_i\}, \{c_{i+1}\}, \{c_{i+2}\})$ and $F_i = G_{i(n_i - 1 \bmod 3)}$. After this assignment, we make the following modifications to A_{ij},

$$A_{3(n_3)} = \{c_2\}$$
$$A_{3(n_3 - 1)} = \{c_0\}, \text{if } n_3 = 2 \pmod 3.$$

It can be checked that the protocol satisfies the first two conditions of Lemma 2.2. While condition 3 is not satisfied because back-flow is present, by Lemma 2.3, back-flow does not worsen the DMT and therefore, the protocol achieves the cut-set bound on DMT.

D. The Case of $K = 2$

We now handle the case of $K = 2$. As in the previous case, it turns out that it is impossible to construct simple protocols that avoid back-flow. Furthermore, it is not even possible to construct a simple protocol that transmits a distinct symbol at each time slot. We present a construction that maximizes the rate of transmission of distinct symbols (i.e., MG).

Theorem 2.4: For a KPP network with $K = 2$, if the two path lengths are equal modulo 2, then a DMT of $d(r) = 2(1-r)^+$ is achievable. In the case of unequal path lengths, a DMT of $d(r) = 2(1 - \frac{2n_2\, r}{2n_2 - 1})^+$ is achievable, where $n_1 \leq n_2$.

Proof: Let n_1, n_2 be the number of edges in each path, with $n_1 \leq n_2$. It is convenient to relabel the $n_1 + n_2$ edges in the network so as to form a cycle $l_1, l_2, \ldots, l_{n_1+n_2}$ of length $n_1 + n_2$. The specific relabeling is given by

$$l_j = \begin{cases} e_{1j}, & j \leq n_1 \\ e_{2(n_2+n_1+1-j)}, & n_1 < j \leq n_1 + n_2 \end{cases}.$$

We associate edge $l_1, l_2, \ldots, l_{n_1+n_2}$ with coloring subsets $D_1, D_2, \ldots, D_{n_1+n_2}$ respectively with a single constraint,

$$D_j \cap D_{(j+1) \pmod{(n_1+n_2)}} = \phi, \ \forall\, j = 1, 2, \ldots, n_1 + n_2, \tag{10}$$

that simultaneously satisfies orthogonality at source, sink and the half-duplex constraint.

Case 1: $(n_1+n_2) = 0 \pmod 2$*:* For this case we choose 2 as the cycle length of the protocol in our construction. Accordingly let the set of colors be $C = \{c_0, c_1\}$. Set

$$D_j = \begin{cases} \{c_0\}, & j = 1, 3, \ldots, n_1 + n_2 - 1 \\ \{c_1\}, & j = 2, 4, \ldots, n_1 + n_2 \end{cases},$$

which essentially corresponds to coloring the cycle formed by the network alternately with the two colors c_0 and c_1.

Case 2: $(n_1 + n_2) = 1 \pmod 2$*:* The coloring prescribed in Case 1 does not work here, since the cycle is of odd length. Therefore, we resort to a different coloring in this case. This construction transmits $2n_2 - 1$ symbols in $2n_2$ time slots and therefore has a rate of $\frac{2n_2-1}{2n_2}$. We use the set of colors $C = \{c_1, c_2, \ldots, c_N\}$, where $N = 2n_2$. We will create the sets D_j using the following algorithm. In the algorithm, whenever we refer to D_j, with $j \notin \{0, 1, 2, \ldots, n_1 + n_2\}$, we mean $D_j = D_{j \pmod{n_1+n_2}}$ and with $j = 0$, we mean $D_j = D_{n_1+n_2}$.

Construction 2.5:

Step 1: $D_j \leftarrow \phi\ \forall\, j \in \{1, 2, \ldots, n_1 + n_2\}$.

Step 2:

$t \leftarrow 1$;

for $k = 1$ to n_2 in steps of 1:

for $i = 1$ to $n_1 + n_2 - 1$ in steps of 1:

if i is odd, $D_{i-k+1} \leftarrow D_{i-k+1} \cup \{c_t\}$;

if i is even, $D_{i-k+1} \leftarrow D_{i-k+1} \cup \{c_{t+1}\}$;

end for.

$t \leftarrow t + 2$;

end for.

Consider *Case 1* in the construction above. It is clear that a symbol is transmitted from the source in every time slot. Also the sink receives a symbol in every time slot alternately from one of the paths. Thus first two conditions of Lemma 2.2 are satisfied, whereas back-flow is not avoided by this protocol. By invoking Lemma 2.3, the proof follows. A similar analysis of *Case 2* in Construction 2.5 reveals that a DMT of $d(r) = 2(1 - \frac{2n_2 r}{2n_2-1})^+$ can be achieved in the unequal path length case, since the rate (maximum MG) of the protocol is $\frac{2n_2-1}{2n_2}$. ■

III. Half-Duplex KPP Networks With Direct Link

In this section, we consider KPP networks with a direct link between source and the sink. All other assumptions in previous section, such as single-antenna nodes, and half-duplex operation hold good here as well.

Theorem 3.1: For KPP(D) networks, the cut-set bound on DMT is achievable whenever $K \geq 4$.

Proof: When $K \geq 4$, we have established that there exists a simple protocol for the KPP network that avoids back-flow. When applied to the backbone KPP network, this protocol yields an induced channel matrix that is a diagonal matrix with the K path gains appearing cyclically along the diagonal. In the presence of a direct link, as is the case here, clearly it is the path gain $\mathbf{g}_d$ of the direct link that will appear along the diagonal of the induced channel matrix. The path gains of the backbone paths will appear in general below the main diagonal. However it is not hard to see that by suitably delaying symbols along each of the K paths, the path gains $\mathbf{g}_i, i = 1, 2, \ldots, K$ can be made to appear cyclically along a single sub-diagonal, say the D-th sub-diagonal, as shown below:

$$\mathbf{H} = \begin{bmatrix} \mathbf{g}_d & & & \\ \mathbf{g}_1 & \ddots & & \\ & \ddots & \ddots & \\ & & \mathbf{g}_k & \mathbf{g}_d \end{bmatrix}.$$

Next, consider the situation when the network is operated for a duration of $M = mK + D$ time-slots, for some positive integer m. We now invoke Theorem 1.3 to the induced channel matrix $\mathbf{H}$ to arrive at a lower bound on the DMT as,

$$\begin{aligned} d(r) = d_H(Mr) &\geq (1 - \frac{r}{M})^+ + K(1 - \frac{Mr}{M - D})^+ \\ &= (1 - r)^+ + K(1 - \frac{M}{M - D} r)^+, \end{aligned}$$

which, as M tends to infinity, becomes

$$d(r) \geq (K + 1)(1 - r)^+. \tag{11}$$

For KPP(D) networks, the cut-set upper bound on DMT (Lemma 1.1) yields $d(r) \leq (K + 1)(1 - r)^+$. Combining this with the DMT lower bound (11), we get $d(r) = (K+1)(1-r)^+$. ■

Remark 7: Since a two-hop relay network possessing N relays with direct link and with relays isolated is a KPP(D) network with $K := N$, the DMT optimal strategy for these family of networks is given by Theorem 3.1. This turns out to be the same strategy as the slotted-amplify-and-forward (SAF) protocol given in [16] for these networks.

IV. Half-Duplex KPP(I) Networks

We consider KPP(I) networks under the half-duplex constraint. KPP(I) networks have the additional complexity that the K backbone paths are not necessarily node disjoint and hence may have cross-links between relays that are a source of interference, see Fig. 1(b). As pointed out earlier, the definition of this class of network precludes the possibility of cross-links either between the source and relays or else between the destination and relays. Cross-links are permitted only between relays. Therefore, the cut-set bound (Lemma 1.1) that separates the source and the relays gives the same DMT upper-bound $d(r) \leq K(1 - r)^+$, as in the case of KPP networks. In this section, we will show that this DMT is in fact, achievable:

Theorem 4.1: Consider any KPP(I) network with $K \geq 3$. The cut-set bound on the DMT $d(r) = K(1 - r)^+$ is achievable.

The rest of this section is devoted to providing a proof to this statement. In the case of KPP networks, simple protocols en-

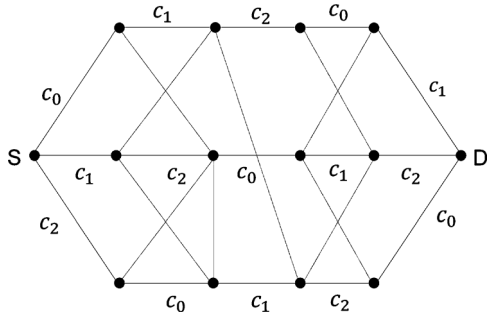

Fig. 5. Regular network and the associated protocol.

sured that there is no interference between the scheduled links, however this will not be possible in KPP(I) networks. We will develop alternate mechanisms to deal with interference.

A. Special Case: Regular Networks

We first study a special class of KPP(I) networks called regular networks, for which the protocol and analysis contains the key intuition required for solving general KPP(I) networks.

Definition 5: If in a KPP(I) network, each relaying path has the same number of relays (say L) so that the relays at distance i from the source can be clubbed together into layer i, and further, interference links exist only between adjacent layers or else, within a layer, then the network is called a (K, L) *regular network*.

An example of regular network is shown in Fig. 5.

We will now show that for regular networks, it is possible to achieve the cut-set bound on DMT. We first describe a simple protocol for the backbone KPP network (see Section II). The cycle length of the protocol is K, and hence we use the set of colors $C = \{c_0, c_1, \ldots, c_{K-1}\}$. The color for edge e_{ij} is given by,

$$A_{ij} = \{c_{i+(j-1)}\},\ 1 \le i \le K,\ 1 \le j \le L+1, \qquad (12)$$

where the subscript $(i+j-1)$ is computed modulo K. The coloring scheme for an example regular network is demonstrated in Fig. 5.

It can be verified that the protocol satisfies the conditions of Lemma 2.2 and thus achieves the cut-set bound on the backbone KPP network. The channel matrix induced by this scheme on the backbone network is a diagonal matrix thereby making it easy to compute the DMT. However, in the presence of interference links, which is the case in the KPP(I) network considered here, the matrix is no longer diagonal but has non-diagonal terms as well due to the presence of interference links. In the case of a regular network, the interference does have a special structure that can be exploited. Any symbol transmitted by the source is first received at the destination from the backbone path, and then received from other paths. This implies that the interference terms are present only on the lower-triangular entries of the matrix. Now, we can use Theorem 1.3, which shows that the DMT of this lower triangular matrix is lower bounded by the DMT of the diagonal matrix, in which interference terms are absent. Thus we get, the DMT of the proposed protocol on the regular network is

$$d(r) \ge d_{\mathrm{BB}}(r) = K(1-r)^+, \qquad (13)$$

where $d_{\mathrm{BB}}(r)$ denotes the DMT on the backbone KPP network. This lower bound also equals the cut-set bound and therefore $d(r) = K(1-r)^+$ and the protocol achieves the network DMT for any regular network. Thus we have the following theorem.

Theorem 4.2: The optimal DMT $d(r) = K(1-r)^+$ of (K, L)-regular networks is achievable using a simple protocol.

Corollary 4.3: For a two-hop relay network without direct link, the optimal DMT is achieved by a simple protocol.

Proof: The two-hop relay network without the direct link is a $(K, 1)$-regular network, where K denotes the number of relays in the network. This holds irrespective of the presence of links between relays, since they only contribute to intra-layer links. Thus the result follows from Theorem 4.2. ■

Remark 8: The result in Corollary 4.3 was also proven in a parallel and independent work [26]. The protocols used in this paper and in [26] are essentially the same as the SAF protocol [16], except that the protocol is applied to a network that does not have a direct link. However, the proof techniques used here and in [26] are very different.

B. General KPP(I) Networks

We now proceed to generalize the ideas from the regular network case to an arbitrary KPP(I) network. The basic idea idea for the regular network was that the simple protocol for the backbone KPP network ensured that the interference pattern falls only on the lower-triangular portion. This will remain the basic idea even for general KPP(I) networks.

Given a KPP(I) network, we can choose a backbone KPP network such that the network is a union of a backbone KPP network with additional edges connecting the K paths. We assume that the network is operated by a simple protocol designed for the backbone KPP network, i.e., the protocol is designed to work in the absence of interference. Our aim now is to study the effect of the same protocol being used in the presence of interference. Every symbol transmitted by the source has an intended backbone path through which it is expected to reach the destination. However, on account of the interfering links, the symbol will typically have more than one path through which to propagate to the destination.

1) Causal Interference: In general KPP(I) networks, protocols that create "causal interference" enforce a certain favorable structure on the interference.

Definition 6: Given a KPP(I) network, a simple protocol designed for its backbone KPP network is said to create *causal interference* in the KPP(I) network if the unique shortest delay experienced by every transmitted symbol is through a backbone path. Such a protocol will be termed as a causal protocol.

In the absence of interference, a simple protocol for a KPP network will induce a diagonal channel matrix. It follows that under the assumption of causal interference, the channel matrix induced by the protocol will be of lower-triangular form, since all the interfering paths incur a larger delay in comparison with the backbone path.

Lemma 4.4: If a protocol for a KPP(I) network achieves DMT $d_1(r)$ in the absence of interference (i.e., when operated on the backbone KPP network), then the protocol will achieve a DMT $d(r) \ge d_1(r)$, in the presence of causal interference.

Proof: Let $\mathbf{H}_{\mathrm{bb}}$ be the channel matrix induced by the protocol in the backbone KPP network. Since there is no interfer-

ence, this channel matrix is diagonal. When the same protocol is employed on the network with causal interference, the induced channel matrix $\mathbf{H}$ will be lower-triangular. By applying Theorem 1.3, we get

$$d_H(r) \geq d_{\mathbf{H}^{(0)}}(r) = d_{H_{\text{bb}}}(r) = d_1(r). \tag{14}$$

■

2) $K = 3$ Case: We now turn to constructing a protocol for a general KPP(I) network. We begin with the case $K = 3$. We will then go on to generalize this for the case $K \geq 3$. Even the $K = 3$ case turns out to be significantly more involved than was the case with regular networks. We state the main result here and defer the detailed proof to the appendix.

Theorem 4.5: For any KPP(I) networks with $K = 3$, there exists a causal protocol that achieves the cut-set bound $d(r) = K(1-r)^+$.

Proof: The idea of the proof is to construct a protocol that incorporates delays at the relaying nodes so that the resulting interference pattern when viewed from the destination appears causal, i.e., the end-to-end channel matrix is lower triangular. The detailed proof appears in Appendix A. ■

3) General $K \geq 3$: We now consider general KPP(I) networks with $K \geq 3$. For $K > 3$, the KPP network is first pre-processed using Lemma A.1 in Appendix A. We then time share between all $\binom{K}{3}$ possible 3-PP subnetworks, for each of which, we use the causal protocol proposed in Theorem 4.5. The induced channel matrix is a lower triangular matrix with all the K product coefficient g_i repeated $\binom{K-1}{2}$ times along the diagonal, with strictly lower-triangular terms involving causal interference. By Theorem 1.3, the DMT of this matrix is no worse than that of the diagonal matrix alone which is $d(r) = K(1-r)^+$.

V. Half-Duplex Layered Networks

In this section, we consider half-duplex layered networks. Full-duplex protocols may be modified to deactivate alternate layers in order to satisfy the half-duplex constraint; but that will lead to a MG loss of a factor of two. If back-flow is taken into account, then the MG loss factor may be as large as three. We will demonstrate in this section that there exists static schedules for which this MG loss is not incurred.

Theorem 5.1: For a layered network with at least 2 relays in each layer, a linear DMT of $d_{\max}(1-r)^+$ between maximum diversity $d_{\max}$ and maximum MG of 1 is achievable using a static schedule.

Proof: The basic idea is to time-share between various $(2, L)$-regular subnetworks of the network. Consider an fc layered network with L layers. Let there be R_i antennas in the i-th layer for $i = 0, 1, \ldots, L+1$. We consider the source as layer 0, and sink as the $(L+1)$-th layer so that $R_0 = R_{L+1} = 1$. Let $M_i := R_{i-1}R_i, i = 1, 2, \ldots, L+1$ be the number of fading coefficients between the $(i-1)$-th and ith layer of relays.

A path from source to sink in a layered network is said to be *forward-directed* if all the edges in the path are directed from i-th layer to $(i+1)$-th layer for some i. Note that if we choose two node-disjoint forward-directed paths, we get a subnetwork which is a $(2, L)$-regular network. Let $\mathcal{P}$ denote the set of all these forward-directed paths. Then $|\mathcal{P}| = N = \prod_{i=1}^{L} R_i$. Let $\mathbf{g}_i$ be the product of fading coefficients on path $P_i \in \mathcal{P}$. We construct the bipartite graph corresponding to $\mathcal{P}$ defined as follows:

1) The vertices on both sides of the bipartite graph correspond to paths in $\mathcal{P}$.
2) Connect a vertex associated with path P_i on the left to a vertex associated with P_j on the right if the two paths are node-disjoint.

Lemma 5.2: The bipartite graph corresponding to $\mathcal{P}$ has a complete matching.

Proof: We will prove this by producing an explicit complete matching on the bipartite graph. Let the relays in the j-th layer be indexed from $0, 1, \ldots, R_j - 1$. A forward-directed path can be uniquely represented by a tuple $(b_1, \ldots, b_L)$, where b_j denotes the index of the relay in the j-th layer visited by that path. For any given path P_i associated to the L tuple $(b_1, \ldots, b_L)$, consider a path $P_{\pi(i)}$ associated to the tuple $(c_1, \ldots, c_L)$ where $c_j = b_j + 1 \pmod{R_j}, j = 1, 2, \ldots, L$. Clearly these two paths are node-disjoint, because $R_j \geq 2, \forall\, j$ by definition of fc layered network. This mapping π creates a complete matching on the set of all forward-directed paths. ■

A path P_i along with its partner path $P_{\pi(i)}$ forms $(2, L)$-regular network, and is activated using the protocol specified in the proof of Theorem 4.2. Cycling through all combinations of $(2, L)$-regular networks, we get an induced lower-triangular matrix, in which $\mathbf{g}_1, \ldots, \mathbf{g}_N$ appears in the diagonal after N time instants. Hence, by Theorem 1.3, the DMT of the protocol $d(r) \geq d_{H_d}(Nr)$ where $\mathbf{H}_d$ is the diagonal matrix with $\mathbf{g}_1, \ldots, \mathbf{g}_N$ on the diagonal. We compute the DMT of $\mathbf{H}_d$ in the following lemma:

Lemma 5.3: Let $\mathbf{H}_d$ be the diagonal matrix with entries $\mathbf{g}_1, \ldots, \mathbf{g}_N$, which are the product coefficients on all possible forward-directed paths in the network. Then

$$d_{H_d}(r) = M_{\min}\left(1 - \frac{r}{N}\right)^+, \tag{15}$$

where $M_{\min} = \min_{i=1}^{L+1} M_i$.

Proof: See Appendix B. ■

Thus the proposed protocol achieves a DMT of

$$d(r) \geq d_{H_d}(Nr) = M_{\min}(1-r)^+. \tag{16}$$

We have that the maximum diversity is equal to the min-cut $d_{\max} = M_{\min}$ in any network [1]. Therefore a DMT of

$$d(r) \geq d_{\max}(1-r)^+ \tag{17}$$

is achievable. ■

VI. Networks With Multiple Antenna Nodes

In previous sections, we considered networks with all nodes having single antenna. In this section, we consider the general case of networks with multiple antenna nodes. Specifically, we consider KPP(I) and layered networks under half-duplex constraints. As it is difficult to characterize the DMT completely, we present lower bounds, i.e., an achievable DMT region.

A. Half-Duplex Layered Networks

In this section, we focus on layered networks and we will show how the static scheduling schemes developed for single antenna networks can be utilized in the case of multiple antenna

networks. Note that replacing a multi-antenna relay node with an equivalent number of single-antenna-nodes, the DMT of the resultant network cannot increase, and in this way, lower bounds on the DMT of single-antenna-node networks can be used to derive lower bounds on the DMT for the multiple-antenna case.

In dealing with multiple-antenna-node networks, a key idea is the following: At high values of MG, one should strive to create paths that comprise a sufficient number of antennas to support as many degrees of freedom as possible. For low MGs, we should create paths which are comprised of single antennas so that the effective channel has maximum diversity. Thus in general, we should choose the number of antennas in any given path to be a function of the MG.[4]

Consider a ss-ss layered multi-antenna network with n_0 antennas at the source, n_{L+1} antennas at the sink, and n_i antennas at i-th layer of relays, with $i = 1, 2, \ldots, L$. In order to vary the number of antennas in a path as a function of MG[5], we introduce the notion of partitioning. A *partitioning* $\mathfrak{p}$ of a layered network is a procedure in which several nodes in a layer are clubbed together to form a super-node. We refer to the number of nodes within a super-node as the *size* of the super-node. Partitioning could potentially include partitioning of the source (sink) by which we mean dividing the transmit antennas of the source (sink) into super-nodes. We term the resultant network comprised of super-nodes connected through super-edges as the super-network. For a given partitioning $\mathfrak{p}$, let $S_i^{(\mathfrak{p})}$ be the number of super-nodes in layer i. The number of super-edge-disjoint paths in the super-network is equal to the min-cut of the super-network given by,

$$N^{(\mathfrak{p})} := \min_{\{i=0,1,2,\ldots,L\}} S_i^{(\mathfrak{p})} S_{i+1}^{(\mathfrak{p})}.$$

We will consider only those partitionings under which the resultant super-network is regular. Let $\mathfrak{P}$ be the set of partitioning that have the following properties:

1) Each layer contains at least two super-nodes (this condition is similar to the condition that in a layered network each layer should contain at least one node).
2) The source and the sink are un-partitioned.
3) All layers have the same number of super-nodes, S. Each super-node, except for possibly one, is formed to have a size of $n_i^S := \left\lfloor \frac{n_i}{S} \right\rfloor$.

For the resultant regular super-network, using the protocol given in Theorem 4.2, a lower triangular channel matrix is induced. This has better DMT than the corresponding block-diagonal matrix by Theorem 1.3, yielding the following lower bound on DMT:

$$d(r) \geq \max\{d_{\max}(1-r)^+, \sup_{S \in \{2,3,\ldots,n_{\min}\}} S\, d_{(n_0, n_1^S, \ldots, n_L^S, n_{L+1})}(r)\}, \quad (18)$$

where $n_{\min} = \min n_i$ and $d_{(n_0,n_1,\ldots,n_{L+1})}(r)$ denotes the DMT of a product of independent Rayleigh matrices of size $n_0 \times n_1, n_1 \times n_2, \ldots, n_L \times n_{L+1}$, which can be computed using the techniques given in [15]. The justification for retaining the term $d_{\max}(1-r)^+$ as part of the maximization is because by adopting the matching-forward-directed-paths strategy in the presence of multiple antennas at the source and sink (see Theorem 5.1) we can achieve this lower bound.

B. KPP(I) Networks

In this section, we consider KPP(I) networks with all nodes, including the source and the sink, having multiple antennas. We can use the same optimal protocol for single-atennna KPP(I) networks with $K \geq 3$, notwithstanding the fact that the relays contain multiple antennas. Though this scheme works well when all nodes have the same number of antennas, this is no longer optimal for case with different number of antennas at each node. Unlike the scheme of choosing different paths depending upon MG, as in Section VI-A, we wish to consider a generic scheme where path i is activated for a fraction f_i of the duration. For $K = 3$, we will assume that all paths are equally activated, thus the only achievable fractions of activation are $f_i = \frac{1}{3}$. For $K > 3$, we will do time-sharing among all the 3PP sub-networks with different fractions of activation. Since we use the 3PP network as a basic unit, any path within a given 3PP network is activated one-third fraction of the duration for which that 3PP is activated. Thus any fraction of time of activation, f_i is limited to be less than $\frac{1}{3}$. In fact, we can show that all time fractions f_i, $1 \leq i \leq K$ are feasible as long as $(f_1, f_2, \ldots, f_K) \in \mathcal{F}$ where

$$\mathcal{F} := \{(f_1, f_2, \ldots, f_K) : \sum_{i=1}^{K} f_i = 1,\ 0 \leq f_i \leq \frac{1}{3}\}. \quad (19)$$

This is shown in Appendix C.

Let $\mathbf{H}_{ij}$ be the fading matrix on edge e_{ij}. Let the product of fading matrices along backbone path P_i be $\mathbf{G}_i$. Then $\mathbf{G}_i = \prod_{j=1}^{n_i} \mathbf{H}_{ij}$. Let the DMT of this path be defined as the DMT of this product matrix $\mathbf{G}_i$, $d_i(r)$, which can be computed according to formulae for the Rayleigh product matrix given in [15]. The channel matrix induced by the scheme will be block-lower-triangular, and by invoking Theorem 1.3, and result for the the DMT of a MIMO parallel channel with repeated coefficients in [1], we obtain a lower bound on the DMT of the protocol as,

$$d(r) \geq \sup_{(f_1,f_2,\cdots,f_K)\in\mathcal{F}} \inf_{\left\{ \substack{(r_1,r_2,\cdots,r_K):\\ \sum_{i=1}^K f_i r_i = r} \right\}} \sum_{i=1}^{K} d_i(r_i). \quad (20)$$

VII. Code Design

In this section, we demonstrate codes that the source needs to utilize in order to achieve the DMT promised in the earlier sec-

[4]We assume that the MG is fixed prior to operation and therefore that we can indeed choose the partitioning scheme based on the current value of MG. This makes practical sense because the decoder needs to be aware of the rate in order to decode the information. When this information is passed on to the decoder, we will assume that all the intermediate relays are also made aware of the rate.

[5]The idea of varying the protocol parameters depending on the MG r was used in [10] for the NSDF protocol.

tions. While a random Gaussian code will do the job, the focus here is on codes with shorter block length and lower decoding complexity.

A. DMT-Achieving Codes

Consider any network and AF protocol described above, and let us say the network is operated for N slots using such a protocol to obtain an induced channel matrix $\mathbf{Y} = \mathbf{H}\mathbf{X} + \mathbf{W}$ where X is a $(M \times 1)$ vector and $\mathbf{Y}, \mathbf{W}$ are $P \times 1$ vectors and $\mathbf{H}$ is a $P \times M$ matrix. However, to achieve the DMT of this induced channel one needs to code over both space and time, i.e, transmit a matrix $\mathbf{X}$ drawn from a space-time (ST) code $\mathcal{X}$ as opposed to just sending a vector. In order to obtain an induced channel with $\mathbf{X}$ being a $M \times T$ matrix, we do the following: Instead of transmitting a single symbol, each node transmits a row vector comprising of T symbols during each activation. Then the induced channel matrix takes the form: $\mathbf{Y} = \mathbf{H}\mathbf{X} + \mathbf{W}$, where $\mathbf{X}$ is now an $M \times T$ matrix, $\mathbf{Y}, \mathbf{W}$ are $M \times T$ matrices with channel matrix $\mathbf{H}$ remaining as before. We will regard the product MT as representing the block length of the ST code $\mathcal{X}$ since the transmission of code matrix $\mathbf{X}$ takes place over MT channel uses.

Now from [12], we know that if the code matrix is drawn from an approximately universal code $\mathcal{X}$, then the code $\mathcal{X}$ will achieve the optimal DMT of the channel matrix H irrespective of the statistics of the channel. Explicit minimal delay approximately universal codes for the case when $T = M$ are given in [13], constructed based on appropriate cyclic division algebras (CDA) [17]. These codes can be used here to achieve the optimal DMT of the induced channel matrix, thus providing an explicit communication scheme from end-to-end. Recent work [32] on decoding algorithms for these approximately universal codes has shown that these codes can be decoded efficiently using a *regularized lattice decoder*, whose computational complexity is linear in the rate.

B. Short DMT-Optimal Code Design

If the channel $\mathbf{H}$ is block diagonal, then we can use shorter block-length codes designed for the MIMO parallel channel in [30], [31]. This construction is shown to be approximately universal in [14]. If the channel is block-lower-triangular, as is the case for all the protocols considered in this paper, then we can use the parallel-channel coding scheme along with a successive decoding scheme, where each block that achieves the full DMT is decoded separately and its effect subtracted from the next block. This scheme does suffer from error propagation, since if we decode T blocks, the error probability gets multiplies at most by a factor of T. However, this does not affect the SNR scaling and therefore does not affect the DMT.

APPENDIX A
PROOF OF THEOREM 4.5

We begin with some definitions. All these definitions are with respect to the graph of a KPP(I) network, and their scope is limited to this proof.

Definition 7: A *partition* $\mathcal{P}$ is defined as a set of K nodes obtained by selecting precisely one node from each of the K parallel paths.

We use the term partition here in the sense of a boundary separating one part of the graph from the other, although we do permit edges to cross the partition. The nodes on each backbone path can be assumed to be ordered from left to right. Therefore it is meaningful to speak of nodes that are to the left of the partition and nodes that are to the right of the partition. We use this natural partial order on the set of nodes to define a partial order on partitions.

Definition 8: The set of nodes to the left of a partition $\mathcal{P}$ is denoted as $V_l(\mathcal{P})$, and the set of nodes to the right of $\mathcal{P}$ is denoted as $V_r(\mathcal{P})$. A partition $\mathcal{P}_1$ is said to be on the left of another partition $\mathcal{P}_2$ if $V_l(\mathcal{P}_1) \subseteq V_l(\mathcal{P}_2)$. The set of nodes in between $\mathcal{P}_1$ and $\mathcal{P}_2$ with $\mathcal{P}_1$ being to the left of $\mathcal{P}_2$ are precisely $V_r(\mathcal{P}_1) \cap V_l(\mathcal{P}_2)$.

Definition 9: Two partitions $\mathcal{P}_1, \mathcal{P}_2$ with $\mathcal{P}_1$ to the left of $\mathcal{P}_2$ are said to form a stage, s if there are no links between $V_l(\mathcal{P}_1)$ and $V_r(\mathcal{P}_1)$ and there are no links between $V_l(\mathcal{P}_2)$ and $V_r(\mathcal{P}_2)$, i.e., no links cross either partition. Then $\mathcal{P}_1$ is called the left partition of the stage, $\mathcal{P}_l(s)$ and $\mathcal{P}_2$ is called the right partition of the stage, $\mathcal{P}_r(s)$. Nodes in between $\mathcal{P}_l(s)$ and $\mathcal{P}_2$ are called the internal nodes of the stage.

Definition 10: Fix an order among the K parallel paths of the network. Given a subset $S \subseteq \{1, 2, .., K\}$, a KPP(I) network restricted to S is defined as the induced subgraph consisting of nodes present in the $|S|$ parallel paths specified by the set S. We may also say that KPP(I) network is restricted to the parallel paths specified by S.

Definition 11: (k-Switch): Consider a KPP(I) network restricted to $k > 1$ parallel paths. Let $\mathcal{P}_1$ and $\mathcal{P}_2$ be two partitions on the restricted network, such that $\mathcal{P}_1$ is to the left of $\mathcal{P}_2$. Construct a bipartite graph between set of nodes in $\mathcal{P}_1$ and that in $\mathcal{P}_2$ as follows. We draw an edge in the bipartite graph, between a node in $\mathcal{P}_1$ and a node in $\mathcal{P}_2$ if the corresponding nodes are connected by an edge which does not lie on any backbone path. However, if partitions share a node on a particular backbone path, we do not draw an edge between them. If the bipartite graph has a complete matching, then the two partitions are said to form a k-switch, w^k in the restricted network. $\mathcal{P}_1, \mathcal{P}_2$ are called the left and the right partitions $\mathcal{P}_l(w^k), \mathcal{P}_r(w^k)$ of the k-switch respectively.

Definition 12: A k-switch in a KPP(I) network restricted to k parallel paths is said to be *contiguous* if there are no nodes in between the two partitions on any of the k backbone paths in the restricted network. Otherwise, the k-switch is said to be *non-contiguous*.

Examples of a 3-switch in a 3PP network is shown in the Fig. 6.

Having the necessary definitions in place, we now proceed to specify a protocol for a 3PP network in terms of delays introduced at nodes as well as a schedule of edge activations. We recall that a 3PP(I) network can be viewed as a union of a backbone 3PP network along with links that connect between various nodes in the network, which we will term as interference links. The particular schedule is obtained for a given 3PP(I) network in three steps, as described below:

1) Step 1—Preprocessing: We begin by deactivating nodes in backbone paths that are part of non-contiguous switches, and redefining backbone paths to include shorter

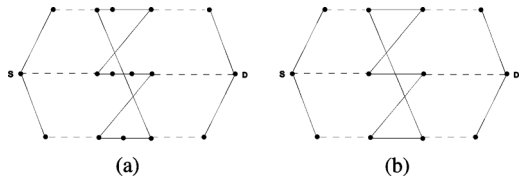

Fig. 6. Examples of 3-switches in KPP(I) networks (a) Non-contiguous 3-switch (b) Contiguous 3-switch.

paths. By deactivating a node, we mean that we will request the node to never transmit. Since there are no transmissions from that node, it can be effectively deleted from the graph. This will lead to a graph in which no two nodes in a given backbone path are connected by a shorter path independent of the backbone path. The below lemma, though we need it only for $K = 3$, is stated in a general setting.

Lemma A.1: Any KPP(I) network can be converted to a second KPP(I) network which does not have any non-contiguous k-switches, for $2 \leq k \leq K$, by deactivating certain nodes from the network and appropriately redefining the backbone paths.

Proof: Let us consider the given KPP(I) network. We employ the following algorithm on the KPP(I) network to perform this conversion:

1) Identify any non-contiguous k-switch in the network for any $2 \leq k \leq K$. If there are no non-contiguous k-switches in the network, terminate the algorithm.
2) Given a non-contiguous k-switch, re-define the backbone path in such a way that the segment of the newly-defined backbone paths lying in between the left and right partition are precisely the edges corresponding to the complete matching. After redrawing the KPP network, it can be seen that there are nodes which are a part of none of the backbone paths. Deactivate these nodes. Since the k-switch was non-contiguous, there exists at least one such node. It can be readily verified that this reconfiguration does not affect the number of backbone paths.
3) Repeat Step 1).

This algorithm clearly terminates, since it deactivates at least one node during each iteration and there are only a finite number of nodes. It is for this reason, that this procedure excludes contiguous switches where there will be no node to deactivate. Note that throughout the iterative process, the number of backbone paths has always remained fixed at K. ■

The above lemma establishes that it is indeed possible to remove all non-contiguous switches for any given KPP(I) network.

2) Step 2—Decomposition Into Stages: After preprocessing the network, we decompose the network into stages. As we shall see later, this decomposition of network into stages will be very useful in identifying a DMT optimal schedule for the network.

Definition 13: A network is said to be decomposed into a set of stages $s_1, s_2, \ldots, s_N$ if all the nodes of the network graph can be split into into N stages $s_1, s_2, \ldots, s_N$ such that $\mathcal{P}_r(s_i) = \mathcal{P}_l(s_{i+1})$ for every i.

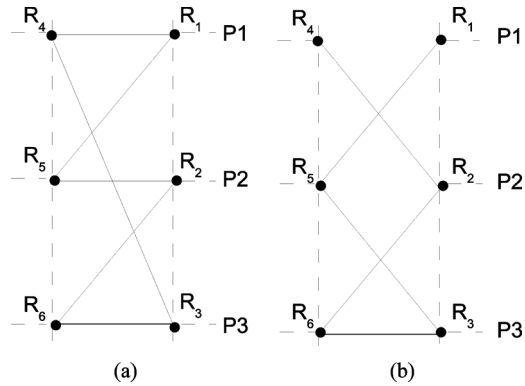

Fig. 7. T-3 stages in KPP(I) network with $K = 3$ (a) Case 1 (b) Case 2.

Now, we attempt to decompose a given 3PP(I) network into a set of stages with each stage having certain properties that will be useful for us to obtain an efficient schedule for the network. We note that, the set of nodes that are adjacent to the destination and the destination itself form a stage, called sink stage. All the nodes in the sink stage other than the sink form the left partition, and the right partition contains the sink and any other two nodes. An analogous definition yields the source stage.

Remark 9: The definition of a stage does not preclude the possibility of having interference links connecting nodes within a partition. We will prove later that such interference links will not present non-causal interference. With this foresight, we will neglect the links present on partitions for now and later return to demonstrate that these do not change the causal nature of the interference.

The decomposition of 3PP(I) network into stages will be carried out in a hierarchial manner, i.e., we identify stages belonging to three different categories, with a category being characterized by some common features. Only after we have identified all stages belonging to the first category, we will start looking for next category of stages from the remaining nodes.

Definition 14: A stage is called a *T-3 stage* (short for Type-3) if its left and right partitions either form a contiguous 3-switch or else contains two contiguous 2-switches when the 3PP(I) network is restricted to two different sets of parallel paths. See Fig. 7 for an example. Apart from stages of this type, we will also regard the source and sink stages as T-3 stages.

It must be noted that any pair of partitions comprising a contiguous 3-switch or two contiguous 2-switches is indeed a stage. This can be proved by contradiction, by showing that if the pair of partitions is not a stage, then there must exist a non-contiguous 3-switch or a non-contiguous 2-switch in the network, which have been assumed to be removed in the previous preprocessing step. Thus it is sufficient to identify all non-contiguous 3-switches and 2-switches to identify T-3 stages.

Having identified all T-3 stages in the network, we focus on the induced subgraph consisting of only nodes that are not internal to any T-3 stage. This subgraph will comprise of multiple components separated by T-3 stages. The subgraph is used to identify the remaining stages of the network. We proceed to identify another category of stages in these components.

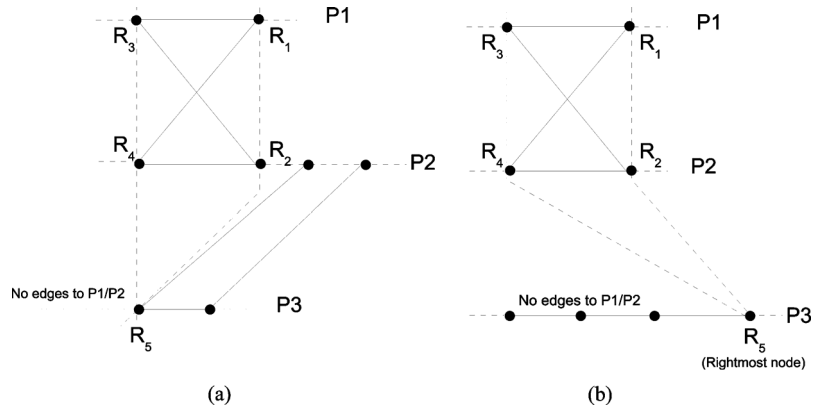

Fig. 8. Type T-2 stages in a KPP(I) network with $K = 3$ (a) Case 1 (b) Case 2.

The following definition is with respect to a given network component.

Definition 15: Consider a contiguous 2-switch w^2 in the network restricted to two paths (say paths P_1 and P_2). Consider the leftmost node v in P_3 connected to $V_r(\mathcal{P}_r(w^2))$. if there is no such node connected $V_r(\mathcal{P}_r(w^2))$, choose the rightmost node in P_3 in the network component as v. Choose $\mathcal{P}_l(w^2) \cup \{v\}$ as the left partition and $\mathcal{P}_r(w^2) \cup \{v\}$ as the right partition to form a stage. We call a stage of this type as a *T-2 stage*, see Fig. 8 for two examples of T-2 stages.

The definition assumes that the given choice of partition indeed forms a stage. This can be proved using the fact that the component does not contain any T-3 stage. Continuing the hierarchial decomposition into stages, we now focus on the induced subgraph consisting of only nodes that are not internal to any T-3 or T-2 stage. Once again, we are left with remaining network components. Any such component does not contain a stage of type T-2 or T-3 and is thus guaranteed not to have any switches. We proceed to decompose further as follows:

If there are no interference links in the given component, then the whole component is treated as a stage. An example of such a stage is given in Fig. 9(a). If there are interference links, we pick a backbone path to which this link is connected and label it as P_1. Find the rightmost node in P_1, say R_1, that is connected to an interference link. Let the other end of this interference link be connected to a node R_2 in a second backbone path, which we label as P_2. Conceivably, node R_1 could be connected via other interference links to nodes in path P_2 that lie to the right of node R_2. If this is the case, then we choose the rightmost such node and relabel this node on path P_2 as node R_2. Then we do the following to identify a stage from the component:

1) If there are no interference links connecting to a node on P_2 to the right of R_2, then R_1, R_2 and the rightmost node of the remaining backbone path P_3 in the component, form a left partition; this left partition along with the rightmost three nodes of the component acting as the right partition, can be verified to be a stage. An example of such a stage is given in Fig. 9(b)
2) If there is an interference link connected to a node on path P_2 that is to the right of R_2, this interference link must originate from the remaining backbone path P_3 since the network component does not contain any switches; choose the leftmost node R_3 in P_3 connected to the right of R_2; this node could potentially be connected to multiple nodes in P_2. Let R_4 be the leftmost node in P_2 connected to R_3. Now set R_1, R_2 and R_3 as the three nodes for the left partition and R_1, R_4 and R_3 as the three nodes for the right partition. An example of such a stage is given in Fig. 9(c).

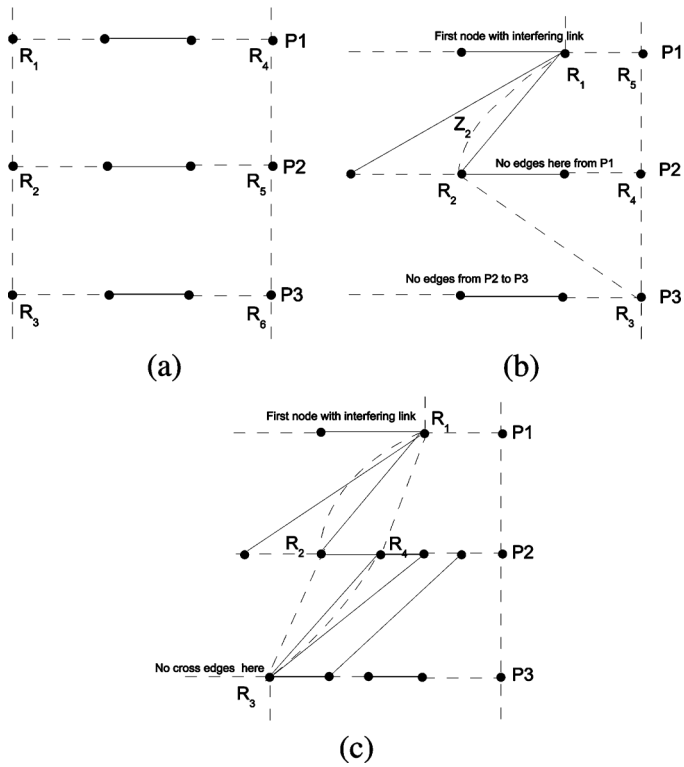

Fig. 9. T-1 stages in KPP(I) network with $K = 3$ (a) Case 1 (b) Case 2 (c) Case 3.

A feature of the stages produced by this last step is that no switches are contained within the stage and we will label such stages as Type-1 stages (T-1). Also, the procedure of identifying a T-1 stage starts with the identification of an interference link, and the T-1 stage so obtained will encompass the interference

link. Since there are only finite number of interference links, the hierarchial procedure will result in the decomposition of the entire network into stages of type T-3, T-2 or T-1.

Once the network is decomposed into stages, we assign colors to edges, indicating their activation pattern, as well as delays to nodes.

3) Step 3—Assigning Colors and Delays: In this final step, we construct the protocol for the given network by suitably assigning to each edge, a color that represents the activation time slot. We also introduce delays at node in order to make the protocol causal. Note that the given network does not possess any non-contiguous switches and that the network has been decomposed into N stages, $s_1, s_2, \ldots, s_N$, with each stage being of type T-1, T-2 or T-3.

A protocol will yield only causal interference if at every node where there is an interference link branching out from the backbone path, the shortest delay from the node to the destination through the branch-out link is strictly greater than the delay on the backbone path. It is easily seen that this condition does not depend on the network to the "left" of this node. For this reason, in designing the protocol, we can begin at the rightmost node and make sure that this condition is satisfied for all nodes.

Hence, to construct the protocol, we start from the rightmost stage s_N and proceed toward left stages, assigning colors to edges and delays to nodes in each stage. At the end of this process, we will obtain a protocol for communication, where every node will be given a delay that must be added to every symbol that it receives and every edge will be given a color (or equivalently, a time slot) to amplify and forward the last symbol that it received. As in 3PP network, the protocol will have a cycle length of 3, and the colors are denoted as A, B and C in the order of time slots. It must be noted that the delays assigned to a node and colors assigned to edges on either side of the node are not independent to each other. We adopt the convention that a delay of zero at a node corresponds to edges on either side of the node in the backbone path transmitting consecutively(for instance, if they have been assigned colors A and B respectively, then that corresponds to zero delay). If edges on either side are assigned colors A and C respectively, that means that the delay at the node is 1. Under this convention, a node can be assigned any delay other than $2 \pmod 3$, without violating the half-duplex constraint at the node.

Before explaining the details, we first provide an outline of how delays are added to nodes in each of the stage, as we move from right to left. Assume that we reach a stage s_i.

1) From any node in the stage, we will make sure that the delay to the right partition, along the particular backbone path on which the node is situated, is strictly less than the delay to the right partition on any other path that might lead from the same node (In making this assessment, we will ignore any links connecting nodes within a partition as noted in Remark 9). We will ensure this by adding delays to the three nodes in the right partition of the stage, which we will denote by D_i^r, where D_i^r is a three-component vector comprising of the three delays to be added. We refer to this as right-compensation of a stage.
2) We will further add delays to the nodes in the left partition D_i^l so that the delays incurred in traveling from left partition to right partition along any of the backbone paths is the same. We refer to this as left-compensation of the stage. It is easy to see that adding delays on the left partition does not change the right compensation of the stage, because the delays added to nodes in left partition will not affect the schedule of activations of edges within the stage.

We will take a quick look into how delay compensation works in different categories of stages. For T-1 stages, it can be easily proved that there exists delays that can lead to right-compensation and left-compensation. Further, if D_i is a right compensation vector for the T-1 stage, there exists another right compensation vector D_i' such that D_i and D_i' differ in only one component and even in that component the difference in the values is equal to one modulo three. This is because the T-1 stage can contain at most one interference link, and hence there will always be a node that is not affected by this link. We will utilize this degree of freedom in choosing the right compensation vector for T-1 stages later. However, as it turns out, just adding delays is not sufficient for stages containing switches, i.e., for T-3 and T-2 stages. So for these stages, we resort to "neutralizing" all interfering links by carefully designing the protocol. An interference link is considered neutralized under a protocol, if the receiving node of the interference link is scheduled to receive at a different time than the time during which the transmitting node is active. We will explain this process in detail in what follows by detailing how the delays, and colors are assigned within each stage.

We start with the rightmost stage s_N, the sink stage, that is comprised of three single-edge paths connecting from the backbone paths to the sink. We assign three colors A, B and C to the three edges, so that the transmissions to the destination are all orthogonal. We set D_N^l as the zero vector since no left-compensation is required for this stage. Whenever a stage, s_{i+1} is compensated and colors are assigned to the edges inside it, the following information is passed on to the stage s_i that is immediately to its left: its left delays, D_{i+1}^l and the colors on the edges in backbone paths immediately to the right of $\mathcal{P}_l(s_{i+1})$, which we call as right colors and which can be regarded as a vector comprising of the three colors on the three respective, ordered, backbone paths. Thus, the right colors and D_N^l are passed to the stage immediately to the right of sink stage.

In this procedure, we first consider the case when the i-th stage s_i is a T-1 stage. First, the delays D_i^r for right-compensation are computed. Then the delays $D_i^r + D_{i+1}^l$ are added to the right partition. However, if this gives delays that violate half-duplex constraints (i.e., one of the delays turns out to be equal to 2 modulo 3), then the degree of freedom in the $T1$ stage is utilized to add more delay to a path so that the delays do not violate the half-duplex constraint in any of the paths. Inside the stage, the edges are colored consecutively so that there is no delay at any node inside the stage. The left-compensation delays D_i^l are computed so that the delays on all the backbone paths become equal. We make an additional modification in the special case that the stage s_{i-1} to the left is of T-3 type and all the right colors output to stage s_{i-1} are the same. In this case, the degree of freedom in choosing the right compensation delays for the stage s_i is utilized so that the right colors of s_{i-1} are not the

same. The reason for this modification will become clear later when we consider T-3 stages.

We now consider the case when s_i is a T-2 stage. We first color the two edges of the contiguous 2-switch on the backbone path with distinct colors. It can be verified easily that this can always be done without violating the half-duplex constraint irrespective of the right colors given to stage s_i. This coloring neutralizes the interference links in the stage and therefore, there is no need for right compensation of the stage, i.e., we set $D_i^r = 0$. However delays still need to be added to the right partition of the stage since this is also the left partition of stage s_{i+1}. Delays are added to the right partition such that the delay on each node on the right partition is greater than or equal to the delays D_{i+1}^l. We compute the left compensation vector D_i^l to ensure that all the backbone paths have equal delays.

As the last case, let s_i be of type T-3. Then all the interference links can be effectively neutralized if the three edges on the backbone paths can be assigned distinct colors. It can be easily shown that this can be done without violating the half-duplex constraint as long as the right colors for this stage are not all the same. As an example, if the right colors are (A, B, B), then we can choose the colors in stage s_i as (B, A, C). However if the right colors are (A, A, A), we cannot get distinct colors on the stage s_i without violating the half-duplex constraint, since no edge in s_i is allowed to have color A. Therefore, we must ensure that the right colors are all not the same. The right colors will not all be the same if the stage s_{i+1} is a T-3 or a T-2. If s_{i+1} is of type T-1, then it has already been modified in such a way that the right colors of s_i are not all same. Thus all the interference links in the T-3 stage can be neutralized by choosing distinct colors on the three paths. Thus, just as in the T-2 case, there is no need for right compensation of the stage, i.e., we set $D_i^r = 0$. Delays are added to the right partition such that the delay on each node on the right partition is greater than or equal to the delays D_{i+1}^l. We compute the left compensation vector D_i^l to ensure all the backbone paths have equal delays.

Now the delay from any node in stage s_i to its right partition of is strictly greater than that along the corresponding backbone path to the right partition. Once the right partition has been reached, both paths incur equal delay (since this would have been ensured within the stage s_{i-1} itself). In this way, we have ensured that in the simple protocol obtained by the above procedure, the unique shortest delay experienced by every transmitted symbol is through a backbone path. Hence the interference is causal under the designed protocol. An example of a 3PP(I) network, after decomposition into stages, coloring and delaying the network is depicted in Fig. 10. In this figure, the numbers denote the amount of delay to be added at that particular node. For example, the delay of 3,indicates that the node waits for 3 time slots (which amounts to one protocol cycle).

However, as noted in Remark 9, we have not accounted for the interfering links between nodes within a partition while constructing the protocol. In the lemma given below, we justify that interference links connecting nodes inside a partition do not alter the causal nature of the protocol.

Lemma A.2: Assume that a protocol is designed for a 3PP(I) network by the procedure detailed above. Then the edges that are present within any partition of the network do not alter the causal nature of the protocol.

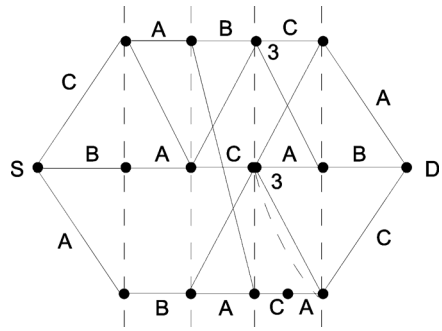

Fig. 10. Adding colors and delay in KPP(I) network.

Proof: It suffices to show that for any link inside a left partition, the delay along the backbone path leading from the tail of the interference link to the right partition is lesser than that along any other path to the right partition from the same node.

Let the network be decomposed into N stages, $s_1, s_2, \ldots, s_N$. Without loss of generality, consider an interference link in the left partition $\mathcal{P}_i$ of an arbitrary stage $s_j, 1 \leq j \leq N$ connecting nodes R_1 and R_2 on the first and second backbone paths, say P_1 and P_2. Then focussing on the path from R_1, we will prove that the link under consideration does not create any non-causal interference. By symmetry the same will hold even if we choose R_2. Assume that delay encountered by a symbol forwarded via the backbone paths P_1 and P_2 within the stage s_j be d_1 and d_2 respectively. As part of left compensation, let d_1' and d_2' be added to nodes R_1 and R_2 respectively.

Any symbol which is received at R_1 is delayed for d_1', and then forwarded. Then the symbol simultaneously starts following both the backbone path P_1 as well as the path via the interference link. Therefore the delay d_1' does not come into picture for this relative delay comparison and we have that the delay through the shortcut path that includes the interference link is equal to $1 + d_2' + d_2$ and the delay through the back-bone path from R_1 is d_1. Since the algorithm detailed above ensures that $d_2' + d_2 \geq d_1$, the interference remains causal considering the path beginning from the relay R_1. Thus the protocol remains causal even in the presence of interference link within a partition. ■

This completes the proof of the theorem.

APPENDIX B
PROOF OF LEMMA 5.3

Proof: We need to compute $d_{H_d}(r)$. Let $H^{(k)}$ be the set of M_k fading coefficients of links connecting nodes in the $(k-1)$-st layer to those in the k-th layer, $k = 1, 2, \ldots, L+1$. We associate with every path P_i an $(L+1)$-tuple of fading coefficients $\Theta_i = (\theta_{i1}, \theta_{i2}, \ldots, \theta_{i(L+1)})$, where $\theta_{ik} \in H^{(k)}$ is the fading coefficient of the kth link in path P_i. Now $\mathbf{g}_i$ is related to Θ_i as $\mathbf{g}_i = \prod_{k=1}^{L+1} \theta_{ik}$.

Consider a variable transformation where $\boldsymbol{\alpha}_{jk}$ is defined such that $\rho^{-\boldsymbol{\alpha}_{jk}} = |\boldsymbol{\theta}_{jk}|^2$. Now the DMT $d(r)$ of the parallel channel is characterized as,

$$\begin{aligned}
\rho^{-d(r)} &\doteq \Pr\{\log\det(I+\rho\mathbf{H}_d\mathbf{H}_d^\dagger) \le r\log\rho\} \\
&= \Pr\{\det(I+\rho\mathbf{H}_d\mathbf{H}_d^\dagger) \le \rho^r\} \\
&= \Pr\{\prod_{i=1}^{N}(1+\rho|\mathbf{g}_i|^2) \le \rho^r\} \\
&= \Pr\{\prod_{i=1}^{N}(1+\rho\prod_{k=1}^{L+1}|\boldsymbol{\theta}_{ik}|^2) \le \rho^r\} \\
&= \Pr\{\prod_{i=1}^{N}(1+\rho\prod_{k=1}^{L+1}\rho^{-\boldsymbol{\alpha}_{ik}}) \le \rho^r\} \\
&= \Pr\{\prod_{i=1}^{N}(1+\rho^{1-\sum_{k=1}^{L+1}\boldsymbol{\alpha}_{ik}}) \le \rho^r\} \\
&\doteq \Pr\{\prod_{i=1}^{N}\rho^{(1-\sum_{k=1}^{L+1}\boldsymbol{\alpha}_{ik})^+} \le \rho^r\} \\
&= \Pr\{\sum_{i=1}^{N}(1-\sum_{k=1}^{L+1}\boldsymbol{\alpha}_{ik})^+ \le r\} \qquad (21) \\
&\le \Pr\{\sum_{i=1}^{N}(1-\sum_{k=1}^{L+1}\boldsymbol{\alpha}_{ik}) \le r\}. \qquad (22)
\end{aligned}$$

Since θ_{ik} refers to the fading coefficient at the k-th layer on the i-th path, several of the θ_{ik} correspond to the same fading coefficient $\mathbf{h}_j^{(k)}$, which is the j-th fading coefficient in the k-th layer with $1 \le j \le M_k$. Therefore, in the sum above, several of the $\boldsymbol{\alpha}_{ik}$ actually correspond to the same fading coefficient. So we define the scaling coefficient corresponding to a fading coefficient $\mathbf{h}_j^{(k)}$ as $\alpha_j^{(k)}$. Each $\alpha_j^{(k)}$ appears in the above sum $N_k := \frac{N}{M_k}$ times. Thus we get

$$\rho^{-d(r)} \dot{\le} \Pr\{N - \sum_{k=1}^{L+1} N_k \sum_{j=1}^{M_k} \boldsymbol{\alpha}_j^{(k)} \le r\} \doteq: \rho^{-d_1(r)}. \qquad (23)$$

Using Varadhan's lemma [35], $d(r)$ and $d_1(r)$ can be written as,

$$d(r) = \inf_{(\alpha_{ik})\in S} \sum_{i,k} \alpha_{ik}, \qquad (24)$$

$$d_1(r) = \inf_{(\alpha_{ik})\in T} \sum_{i,k} \alpha_{ik}, \qquad (25)$$

where the sets S and T be defined as

$$S := \{(\alpha_{ik}) : \sum_{i=1}^{N}(1-\sum_{k=1}^{L+1}\alpha_{ik})^+ \le r\}, \qquad (26)$$

$$T := \{(\alpha_{ik}) : \sum_{i=1}^{N}(1-\sum_{k=1}^{L+1}\alpha_{ik}) \le r\}. \qquad (27)$$

Simplifying $d_1(r)$ as per (23), we get,

$$d_1(r) = \inf_{\left\{\substack{N-\sum_{k=1}^{L+1}N_k\sum_{j=1}^{M_k}\alpha_j^{(k)}\le r\ ,\\ \alpha_{(j)}^k\ge 0}\right\}} \sum_{k=1}^{L+1}\sum_{j=1}^{M_k}\alpha_j^{(k)}. \qquad (28)$$

Define $\alpha^{(k)} := \sum_{j=1}^{M_k}\alpha_j^{(k)}$ to obtain,

$$d_1(r) = \inf_{\{N-\sum_{k=1}^{L+1}N_k\alpha^{(k)}\le r\ ,\ \alpha^{(k)}\ge 0\}} \sum_{k=1}^{L+1}\alpha^{(k)}. \qquad (29)$$

Let ℓ be the index such that $N_\ell = \max N_k$. The infimum is attained in the above minimization by $\alpha^{(\ell)} = \frac{N-r}{N_\ell}, \alpha^{(i)} = 0, \forall\ i \ne \ell$ and the value of the infimum is $\frac{N-r}{N_\ell}$. Thus $d(r) \ge d_1(r) = \frac{N-r}{N_\ell}$.

We can further show that $d(r) = d_1(r)$ by showing that the optimizing α_{ik} in T for $d_1(r)$ also lies in S. An optimizing assignment of α_{ik} for obtaining $d_1(r)$ is given by, $(\alpha_{i\ell})^* = \alpha_j^{(\ell)}$, for some j, and we know that $\alpha_j^\ell = \frac{N-r}{N_\ell M_\ell} = \frac{N-r}{N}, \forall\ j$. Thus

$$(\alpha_{ik})^* = \frac{N-r}{N}, \text{if } k = l, \text{ and 0 otherwise.} \qquad (30)$$

Clearly $(\alpha_{ik})^* \in S \cap T$. This implies that this is the optimizing α_{ik} for (20) too. Therefore, we have

$$d(r) = d_1(r) = \frac{N-r}{N_\ell}. \qquad (31)$$

■

Appendix C
Different Fractions of Activation in Multi-Antenna KPP(I) Networks

Lemma C.1: Let $K \ge 4$, and let $M = \binom{K}{3}$ be the possible combinations of 3PP networks in the given KPP network. The extreme points of the convex region $\mathcal{F}$ in (19) are given by vectors which have only three non-zero entries equal to $\frac{1}{3}$.

Proof: The region $\mathcal{F}$ is convex, and thus every point in $\mathcal{F}$ can be expressed as a linear combination of its extreme points. Suppose it is not the case. Then there exists an extreme point $x = [x_1\ x_2\ \dots\ x_K]$, such that at least one entry, say x_1 without loss of generality, is less than $\frac{1}{3}$. But, due to constraints of the region, this forces one more entry, say x_2 without loss of generality, to be greater than $\frac{1}{3}$. Then clearly, $\exists\ \delta > 0$ such that

$$x_1 + \delta \le \frac{1}{3}, \quad x_2 - \delta \ge 0$$
$$x_1 - \delta \ge 0, \quad x_2 + \delta \le \frac{1}{3},$$

so that $x' = [x_1+\delta\ x_2-\delta\ \dots\ x_K]$ and $x'' = [x_1-\delta\ x_2+\delta\ \dots\ x_K]$ belong to the region $\mathcal{F}$. Now,

$$x = \frac{1}{2}x' + \frac{1}{2}x'',$$

which contradicts our hypothesis that x is an extreme point. This completes the proof. ■

ACKNOWLEDGMENT

Thanks are due to K. Vinodh and M. Anand for useful discussions.

REFERENCES

[1] K. Sreeram, S. Birenjith, and P. V. Kumar, "DMT of multihop networks: End points and computational tools," *IEEE Trans. Inf. Theory*, vol. 58, no. 2, pp. 804–819, Feb. 2012.

[2] T. M. Cover and A. A. E. Gamal, "Capacity theorems for the relay channel," *IEEE Trans. Inf. Theory*, vol. 25, no. 5, pp. 572–584, Sep. 1979.

[3] A. Sendonaris, E. Erkip, and B. Aazhang, "User cooperation diversity-Part I: System description," *IEEE Trans. Commun.*, vol. 51, no. 11, pp. 1927–1938, Nov. 2003.

[4] J. N. Laneman and G. W. Wornell, "Distributed space-time-coded protocols for exploiting cooperative diversity in wireless networks," *IEEE Trans. Inf. Theory*, vol. 49, no. 10, pp. 2415–2425, Oct. 2003.

[5] L. Zheng and D. Tse, "Diversity and multiplexing: A fundamental tradeoff in multiple-antenna channels," *IEEE Trans. Inf. Theory*, vol. 49, no. 5, pp. 1073–1096, May 2003.

[6] J. N. Laneman, D. Tse, and G. W. Wornell, "Cooperative diversity in wireless networks: Efficient protocols and outage behavior," *IEEE Trans. Inf. Theory*, vol. 50, no. 12, pp. 3062–3080, Dec. 2004.

[7] K. Azarian, H. E. Gamal, and P. Schniter, "On the achievable diversity-multiplexing tradeoff in half-duplex cooperative channels," *IEEE Trans. Inf. Theory*, vol. 51, no. 12, pp. 4152–4172, Dec. 2005.

[8] R. U. Nabar, H. Bolcskei, and F. W. Kneubuhler, "Fading relay channels: Performance limits and space-time signal design," *IEEE J. Select. Areas Commun.*, vol. 22, no. 6, pp. 1099–1109, Aug. 2004.

[9] M. Yuksel and E. Erkip, "Multiple-antenna cooperative wireless systems: A diversity-multiplexing tradeoff perspective," *IEEE Trans. Inf. Theory*, vol. 53, no. 10, pp. 3371–3393, Oct. 2007.

[10] P. Elia, K. Vinodh, M. Anand, and P. V. Kumar, "D-MG tradeoff and optimal codes for a class of AF and DF cooperative communication protocols," *IEEE Trans. Inf. Theory*, vol. 55, no. 7, pp. 3161–3185, Jul. 2009.

[11] N. Prasad and M. K. Varanasi, "High performance static and dynamic cooperative communication protocols for the half-duplex fading relay channel," *IEEE Trans. Wireless Commun.*, vol. 9, no. 1, pp. 328–337, Jan. 2010.

[12] S. Tavildar and P. Viswanath, "Approximately universal codes over slow-fading channels," *IEEE Trans. Inf. Theory*, vol. 52, no. 7, pp. 3233–3258, Jul. 2006.

[13] P. Elia, K. R. Kumar, S. A. Pawar, P. V. Kumar, and H.-F. Lu, "Explicit, minimum-delay space-time codes achieving the diversity-MG tradeoff," *IEEE Trans. Inf. Theory*, vol. 52, no. 9, pp. 3869–3884, Sep. 2006.

[14] P. Elia and P. V. Kumar, "Space-time codes that are approximately universal for the parallel, multi-block and cooperative DDF channels," in *Proc. IEEE Int. Symp. Inf. Theory*, Seoul, Jun. 3, 2009.

[15] S. Yang and J.-C. Belfiore, "Diversity of MIMO multihop relay channels," submitted to *IEEE Trans. Inf. Theory*, Available Online.

[16] S. Yang and J.-C. Belfiore, "Towards the optimal amplify-and-forward cooperative diversity scheme," *IEEE Trans. Inf. Theory*, vol. 53, no. 9, pp. 3114–3126, Sep. 2007.

[17] B. A. Sethuraman, B. S. Rajan, and V. Shashidhar, "Full-diversity, high-rate, space-time block codes from division algebras," *IEEE Trans. Inf. Theory*, vol. 49, no. 10, pp. 2596–2616, Oct. 2003.

[18] K. Sreeram, S. Birenjith, and P. V. Kumar, "Multi-hop cooperative wireless networks: Diversity multiplexing tradeoff and optimal code design," in *Proc. Inf. Theory Appl. Worksh., UCSD*, Feb. 2008.

[19] K. Sreeram, S. Birenjith, K. Vinod, M. Anand, and P. V. Kumar, "On the throughput, DMT and optimal code construction of the K-parallel-path cooperative wireless fading network," in *Proc. 10th Int. Symp. Wireless Pers. Multimedia Commun.*, Dec. 2007.

[20] K. Sreeram, S. Birenjith, and P. V. Kumar, "DMT of multi-hop cooperative networks-Part I: K-parallel-path-networks," in *Proc. IEEE Int. Symp. Inf. Theory*, Toronto, Jul. 2008.

[21] K. Sreeram, S. Birenjith, and P. V. Kumar, "DMT of multi-hop cooperative networks-Part II: Layered and multi-antenna networks," in *Proc. IEEE Int. Symp. Inf. Theory*, Toronto, Jul. 6–11, 2008.

[22] K. Sreeram, S. Birenjith, and P. V. Kumar, Multi-Hop Cooperative Wireless Networks: Diversity Multiplexing Tradeoff and Optimal Code Design [Online]. Available: http://arxiv.org/pdf/0802.1888

[23] K. Sreeram, S. Birenjith, and P. V. Kumar, On the Throughput, DMT and Optimal Code construction of the K-Parallel-Path Cooperative Wireless Fading Network 2007, USC CSI Tech. Rep., CSI-2007-06-07.

[24] S. Pawar, S. Avestimehr, and D. Tse, "Diversity multiplexing tradeoff of the half-duplex relay channel," in *Proc. 45th Allerton Conf. Commun., Control, Comput.*, Sep. 2007.

[25] S. Borade, L. Zheng, and R. Gallager, "Amplify and forward in wireless relay networks: Rate, diversity and network size," *IEEE Trans. Inf. Theory*, vol. 53, no. 10, pp. 3302–3318, Oct. 2007.

[26] S. O. Gharan, A. Bayesteh, and A. K. Khandani, "On the diversity-multiplexing tradeoff in multiple-relay network," *IEEE Trans. Inf. Theory*, vol. 55, no. 12, pp. 5423–5444, Nov. 2009.

[27] A. S. Avestimehr, S. N. Diggavi, and D. N. C. Tse, "Wireless network information flow: A deterministic approach," *IEEE Trans. Info. Theory*, vol. 57, no. 4, pp. 1872–1905, Mar. 2011.

[28] R. Vaze and R. W. Heath Jr., "Maximizing reliability in multi-hop wireless networks with cascaded space-time codes," in *Proc. Inf. Theory Appl. Worksh., UCSD*, Feb. 2008.

[29] R. Vaze and R. W. Heath Jr., "To code in space and time or not in multi-hop relay channels," *IEEE Trans. Signal Process.*, vol. 57, no. 7, pp. 2736–2747, Jul. 2009.

[30] H.-F. Lu, "Explicit construction of multi-block space-time codes that acheive the diversity-MG tradeoff," in *Proc. IEEE Int. Symp. Inf. Theory*, Seattle, USA, 2006.

[31] S. Yang, J.-C. Belfiore, and G. Rekaya, "Perfect space-time block codes for parallel MIMO channels," in *Proc. IEEE Int. Symp. Inf. Theory*, Seattle, USA, 2006.

[32] J. Jalden and P. Elia, "DMT optimality of LR-aided linear decoders for a general class of channels, lattice designs, and system models," *IEEE Trans. Inf. Theory*, vol. 56, no. 10, pp. 4765–4780, Oct. 2010.

[33] T. M. Cover and J. A. Thomas, *Elements of Information Theory*, 2nd ed. New York: Wiley, 2006.

[34] R. A. Horn and C. R. Johnson, *Matrix Analysis*. New York: Cambridge Univ. Press, 1985.

[35] A. Dembo and O. Zeitouni, *Large Deviations Techniques and Applications*, 2nd ed. New York: Springer-Verlag, 1998.

[36] Q. Li, K. H. Li, and K. C. Teh, "Achieving optimal diversity-multiplexing tradeoff for full-duplex MIMO multihop relay networks," *IEEE Trans. Inf. Theory*, vol. 57, no. 1, pp. 303–316, Jan. 2011.

[37] R. Pedarsani, O. Leveque, and S. Yang, "Maintaining diversity: A fundamental challenge in MIMO relay networks," in *Inf. Theory Appl. Worksh.*, San Diego, Jan. 2011.

K. Sreeram is a postdoctoral researcher at the University of California, Berkeley. He received his Ph.D. in Electrical Engineering and M.S. in mathematics from the University of Illinois Urbana- Champaign. He is a recipient of the Van Valkenburg research award from UIUC, 2013, a co-recipient of the Qualcomm Cognitive Radio Contest rst prize, 2010, a recipient of Qualcomm Roberto Padovani Scholarship, 2010, a recipient of the S.V.C. Aiya medal from the Indian Institute of Science, 2008, and a co-recipient of Intel India Student Research Contest rst prize, 2006. His research interests include information theory, wireless networks, approximation algorithms and computational biology.

S. Birenjith recieved his B.Tech. degree in Electronics and Communication Engineering from College of Engineering, Trivandrum and M.Sc.(Engg.) from Indian Institute of Sceince, Bangalore in 2008. During 2008–2011, he was on the faculty of Electronics & Communication Engineering at Govt Engineering Colleges in Kerala. He is currently undergoing his Ph.D. degree programme at the Department of Electrical Communication in Indian Institute of Science, Bangalore.

P. Vijay Kumar (S'80–M'82–SM'01–F'02) received the B.Tech. and M.Tech. degrees from the Indian Institutes of Technology (Kharagpur and Kanpur),and the Ph.D. Degree from the University of Southern California (USC) in 1983, all in Electrical Engineering. From 1983–2003 he was on the faculty of the EE-Systems Department at USC. Since 2003, he has been on the faculty of the Indian Institute of Science, Bangalore. He also holds the position of Adjunct Research Professor at USC. His current research interests include codes for distributed storage, distributed function computation, sensor networks and space-time codes for MIMO and cooperative communication networks. He is an ISI highly-cited author. He is also co-recipient of the 1995 IEEE Information Theory Society Prize Paper Award as well as of a best paper award at the DCOSS 2008 conference on sensor networks. He is currently on the Board of Governors of the IEEE Information Theory Society.